



Intelligence Community Technical Specification

XML Data Encoding Specification for Trusted Data Format

Version 2021-NOV

December 1, 2022

Distribution Notice:

This document has been approved for Public Release and is available for use without restriction.

Table of Contents

Chapter 1 - Introduction	1
1.1 - Purpose	1
1.2 - Scope	1
1.3 - Enterprise Need	1
1.4 - Conventions	2
1.4.1 - XML Namespaces	2
1.5 - Dependencies	2
1.5.1 - Specification Dependencies	2
1.5.2 - Inverse Dependencies	5
Chapter 2 - Development Guidance	7
2.1 - TDF Structure	7
2.1.1 - Version Declarations	10
2.2 - Assertions	10
2.2.1 - Assertion Scopes	10
2.2.1.1 - Assertion Scopes Within TDO	10
2.2.1.2 - Assertion Scopes Within TDC	10
2.2.1.3 - HandlingAssertion scopes within TDO	10
2.2.1.4 - HandlingAssertion scopes within TDC	11
2.2.2 - Mission-Specific Metadata Assertions	11
2.2.3 - Assertions and Data State	11
2.3 - Binding and BindingInfo	12
2.4 - Normalization Method	12
2.5 - Encryption and EncryptionInfo	12
2.6 - Linked or Embedded Data Objects	12
2.7 - MIME type	12
2.8 - BASE-TDF Schematron Rules	12
Chapter 3 - Constraints	13
3.1 - Data Validation Constraint Rules	13
3.1.1 - Purpose	13
3.1.2 - Inherited Constraints	13
3.1.3 - Value Enumeration Constraints	13
3.1.4 - Additional Constraints	13
3.1.4.1 - DES Constraints	13
3.1.5 - Constraint Rules	13
3.2 - Data Rendering Constraint Rules	13
3.2.1 - Purpose	13
3.2.2 - Rendering Constraint Rules	14
Chapter 4 - Conformance Validation	15
4.1 - Definitions	15
4.2 - Why a verbose validation strategy is required	15
4.3 - How to determine the ISM version within structured content	15
4.4 - Required Order of HandlingAssertions	15
4.5 - TDO Validation Steps	16
4.5.1 - Step 1 - TDO aware and cross Assertion constraints	16
4.5.2 - Step 2 - Extension point constraints	16
4.5.3 - Step 3 - TDO structure constraints	17

4.5.4 - Step 4 – ISM consistency constraints	17
4.5.4.1 - Step 4a – Consistency constraints for Assertions with resource level portion markings	17
4.5.4.2 - Step 4b – Consistency constraints for non EDH Handling Assertions with resource level portion markings	18
4.5.4.3 - Step 4c – Consistency constraints for Payloads with resource level portion marking	19
4.5.4.4 - Step 4d – Consistency constraints for Assertions and Payloads with non-resource level markings	20
4.6 - TDC Validation Steps	20
4.6.1 - Step 1 – TDC aware and cross Assertion constraints	20
4.6.2 - Step 2 – Extension point constraints	21
4.6.3 - Step 3 – TDC structure constraints	21
4.6.4 - Step 4 – ISM consistency constraints	21
4.6.4.1 - Step 4a – Consistency constraints for Assertions with resource level portion markings	21
4.6.4.2 - Step 4b – Consistency constraints for Assertions with non-resource level markings	22
4.6.5 - Step 5 - Recursive Validation	22
Appendix A - Feature Summary	23
A.1 - IC-TDF Feature Summary	23
A.1.1 - Features from V2014-DEC to V2021-NOV	23
A.1.1.1 - Features Partial and N/A from V2014-DEC to V2021-NOV	23
A.1.2 - Features from V1 to V2014-DEC	24
A.1.2.1 - Features Partial and N/A from V1 to V2014-DEC	24
Appendix B - Change History	25
B.1 - 2021-NOV Change Summary	25
B.2 - 2019-MAR Change Summary	28
B.3 - V2014-DEC-r2017-JUL Change Summary	30
B.4 - V2014-DEC Change Summary	35
B.5 - V3 Change Summary	36
B.6 - V2 Change Summary	37
Appendix C - List of Abbreviations	40
Appendix D - Bibliography	42
Appendix E - Points of Contact	46
Appendix F - IC CIO Approval Memo	47

List of Figures

Figure 1 - Related Specifications	5
Figure 2 - Inverse Dependency Specifications	6
Figure 3 - TDF Structure	9
Figure 4 - TDF Detailed Structure	9

List of Tables

Table 1 - XML Namepaces	2
Table 2 - Dependencies	3
Table 3 - Constraint Rules	14
Table 4 - Feature Summary Legend	23
Table 5 - IC-TDF Feature comparison V2014-DEC to V2021-NOV	23
Table 6 - IC-TDF Feature comparison V2014-DEC to V2021-NOV	23
Table 7 - IC-TDF Feature comparison V1 to V2014-DEC	24
Table 8 - IC-TDF Feature comparison V1 to V2014-DEC	24
Table 9 - DES Version Identifier History	25
Table 10 - Data Encoding Specification 2021-NOV Change Summary	25
Table 11 - Data Encoding Specification 2019-MAR Change Summary	29
Table 12 - Data Encoding Specification V2014-DEC-r2017-JUL Change Summary	30
Table 13 - Data Encoding Specification V2014-DEC Change Summary	35
Table 14 - Data Encoding Specification V3 Change Summary	36
Table 15 - Data Encoding Specification V2 Change Summary	37

Chapter 1 - Introduction

1.1 - Purpose

This *XML Data Encoding Specification for Trusted Data Format* (IC-TDF.XML) defines detailed implementation guidance for using Extensible Markup Language (XML) to encode IC-TDF data. This Data Encoding Specification (DES) defines the XML elements and attributes, associated structures and relationships, mandatory and cardinality requirements, and permissible values for representing trusted data format data concepts using XML.

1.2 - Scope

The *Intelligence Community Technical Specification Framework* (IC-SF.XML^[4]) defines the basic conceptual structure and outlines the core philosophy of Intelligence Community (IC) technical specifications. For convenience, a copy of this framework is included in every package.

This specification is applicable to the IC and information produced by, stored, or shared within the IC. This DES may have relevance outside the scope of intelligence; however, prior to applying outside of this defined scope, the DES should be closely scrutinized and differences separately documented and assessed for applicability.

1.3 - Enterprise Need

Information sharing within the national intelligence enterprise will increasingly rely on information assurance metadata (including enterprise data headers) to allow interagency access control, automated exchanges, and appropriate protection of shared intelligence. A structured, verifiable representation of security metadata bound to the intelligence data is required in order for the enterprise to become inherently "smarter" about the information flowing in and around it. Such a representation, when implemented with other data formats, improved user interfaces, and data processing utilities, can provide part of a larger, robust information assurance infrastructure capable of automating some of the management and exchange decisions today being performed by human beings.

The IC has standardized the various classification and control markings established for information sharing within the *XML Data Encoding Specification for Information Security Marking Metadata* (ISM.XML^[15]), *XML Data Encoding Specification for Information Resource Metadata* (IRM.XML^[14]) and *XML Data Encoding Specification for Enterprise Data Header* (IC-EDH.XML^[3]) specifications of the Intelligence Community Enterprise Architecture (IC EA) Data Standards. The IC-TDF.XML XML specification further expands on this body of work, adapting and extending it as necessary for Trusted Data Format (TDF) to function as the IC submission format for binding Assertion metadata with data resource(s). This TDF functionality supports the IC way ahead strategy of implementing secure cloud-based information exchange and discovery on the IC Enterprise.

Both enterprise needs and requirements for this specification can be found in the following policies and implementation guidance:

- 200 Series:
 - Intelligence Community Directive (ICD) 208, *Write for Maximum Utility*^[5]

- ICD 209, *Tearline Production and Dissemination* [\[6\]](#)
- Intelligence Community Policy Memorandum (ICPM) 2007-200-2, *Preparing Intelligence to Meet the Intelligence Community's Responsibility to Provide* [\[10\]](#)
- 500 Series:
 - ICD 500, *Director Of National Intelligence Chief Information Officer* [\[7\]](#)
 - ICD 501, *Discovery and Dissemination or Retrieval of Information within the IC* [\[8\]](#)
 - Intelligence Community Standard (ICS) 500-20, *IC Enterprise Standards Compliance* [\[12\]](#)
 - ICS 500-21, *Tagging of Intelligence and Intelligence-Related Information* [\[13\]](#)

1.4 - Conventions

Certain technical and presentation conventions are used in the creation of the IC technical specifications to ensure readability and understanding. For details, please see the “Specification Conventions” chapter in the IC-SF.XML [\[4\]](#).

1.4.1 - XML Namespaces

Namespaces referenced in this document and the prefixes used to represent them are listed in the following table. The namespace prefix of any XML Qualified Name used in any example in this document should be interpreted using the information below.

Table 1 - XML Namepaces

Prefix	URI
edh	urn:us:gov:ic:edh
ism	urn:us:gov:ic:ism
tdf	urn:us:gov:ic:tdf

1.5 - Dependencies

Specifications often rely on other specifications, components or artifacts, either directly or indirectly. For specific definitions of dependency terminology used throughout this section, please see the “Dependency Definitions” chapter in the IC-SF.XML [\[4\]](#).

IC-TDF.XML is dependent on many specifications; all MUST be consulted in conjunction with this document. For example IC-TDF.XML depends on *XML Data Encoding Specification for Trusted Data Format - Base* (BASE-TDF.XML [\[2\]](#)) for some Controlled Vocabulary Enumeration (CVE)s an several Schematron rules.

1.5.1 - Specification Dependencies

This technical specification directly depends on the technical specifications, documentation, and implementations listed in [Table 2](#). The dependencies listed below are directly referenced in this specification (e.g., Schema, Schematron), and are normative or informative as indicated.

The subsequent figure, [Figure 1](#), is an informative graphical representation of all of the Intelligence Community Chief Information Officer (IC CIO) specifications related to this specification. The graphic depicts dependencies. However, the representations may not match an exact schema

import tree or dependency diagram that an analysis of the Schema, Schematron or other documents would yield. For example, the graphic only shows a given specification once even though it may actually be imported by many specifications or be a direct dependency. All IC CIO specifications listed in [Table 2](#) will be shown in [Figure 1](#); however not all IC CIO specifications listed in [Figure 1](#) may appear in [Table 2](#). [Figure 1](#) is to aid users in gaining a general understanding of all dependencies whether direct or transitive.

Table 2 - Dependencies

Name	Dependency Description
<p><i>XML Data Encoding Specification for Information Security Marking Metadata</i> (ISM.XML.V2021-NOVr2022-NOV+^[15])</p>	<p>This specification depends on the LATEST technically sound, approved version of ISM.XML^[15]. The minimum version was based on compliance with the authoritative source, which is ICD-710^[9]. Per ICD-710, all security markings MUST be updated within 365 days of a release of the Register and Manual. As of this release, the latest version of ISM.XML is 2021-NOVr2022-NOV which is based on the Register and Manual released in August, 2019.</p>
<p><i>XML Data Encoding Specification for Enterprise Data Header</i> (IC-EDH.XML.V2019-MAR+^[3])</p>	<p>This specification does not depend on a specific version of IC-EDH.XML^[3]; versions later than version 2019-MAR MAY be used. The minimum version was based on a technical dependency; The merging of ARH into ISM.</p>
<p><i>XML Data Encoding Specification for Revision Recall</i> (REVRECALL.XML.V2021-NOV+^[18])</p>	<p>This specification depends on the LATEST technically sound, approved version of REVRECALL.XML^[18]. The minimum version was based on compliance with the authoritative source, which is ICPM 200-01^[11]. Per ICPM 200-01, there is one new value of FISA_COMPLIANCE_RECALL in CVEnum-RevRecallType, and there is a policy-driven redefinition of the meaning and usage of the RevRecallType ADMINISTRATIVE_RECALL that adds the new text "(but is not used for a FISA-compliance recall)."</p>
<p><i>XML Data Encoding Specification for Trusted Data Format - Base</i> (BASE-TDF.XML.V2021-NOV+^[2])</p>	<p>This specification does not depend on a specific version of BASE-TDF.XML^[2]; versions later than version 2021-NOV MAY be used. The minimum version was based on a technical dependency; Derivation from BASE-TDF.</p>

Name	Dependency Description
<p><i>Intelligence Community Specification Framework</i> (IC-SF.XML.V2021-NOV+[4])</p>	<p>This specification does not depend on a specific version of IC-SF.XML[4]; versions later than version 2021-NOV MAY be used, however, the newest version of IC-SF.XML SHOULD be used as IC-SF.XML is expected to always replace its preceding version. The minimum version was based on technical dependencies on IC-SF.XML; IC-SF.XML is the basic structure of and philosophy behind intelligence community technical specifications.</p>
<p>Schematron[19]</p>	<p>Schematron — International Organization for Standardization (ISO)/International Electrotechnical Commission (IEC) 19757-3:2006 — is a rule-based document schema definition language. In this specification Schematron is a formal language used to express normative business rules, so this reference is normative.</p> <p>The Schematron rules are normative in the sense that they convey criteria that a document MUST adhere to, exactly as English may be used to convey normative criteria. It is not necessary for implementers to use the specific Schematron encoding in this specification. Implementers MAY use any encodings, tools, or languages desired to implement validation schemes for conformance to this specification.</p> <p>Note: The Schematron rules in this specification use Transformations (XSLT) 2.0[21] query binding.</p>
<p>XSLT 2.0[21] implementation of Schematron[19] by Rick Jelliffe (2010-04-14)</p> <p>Note: The only available identifying descriptors for this implementation are the implementer's name and date of release. This implementation may be found at the following Uniform Resource Locator (URL): http://code.google.com/p/schematron/.</p>	<p>The International Organization for Standardization does not create nor endorse reference implementations of its standards. For the purposes of this specification the <i>behavior</i> of the implementation created by Mr. Jelliffe is normative.</p> <p>Implementers MAY use any encodings, tools, or languages desired to implement validation schemes for conformance to this specification. To conform to this specification, a validator MUST find a document valid <i>if and only if</i> the Schematron implementation by Mr. Jelliffe would find the document valid according to the Schematron rules in this specification.</p>

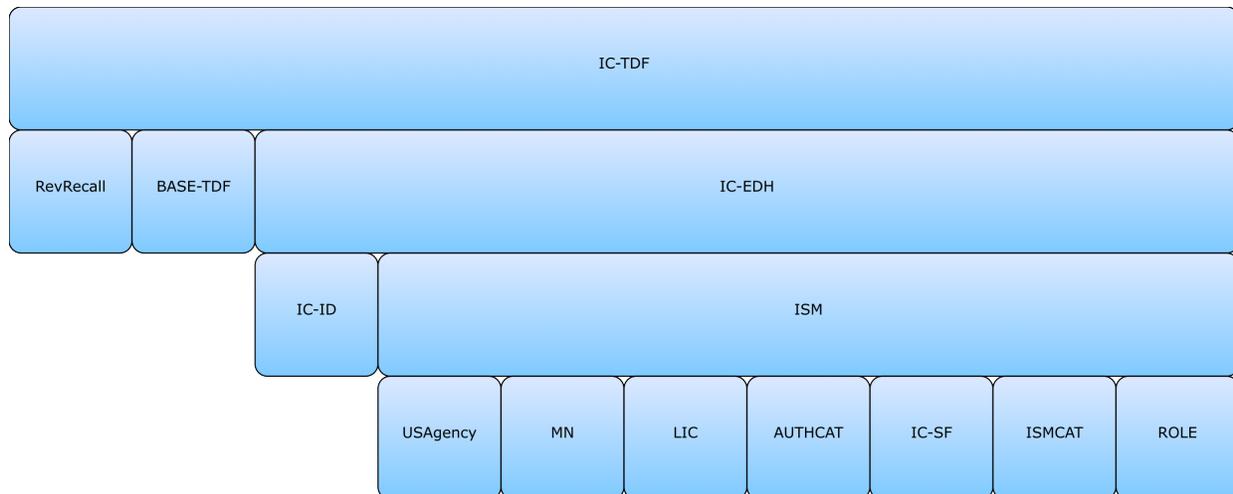


Figure 1 : Related Specifications

1.5.2 - Inverse Dependencies

Generally, it is only necessary to think of the *dependencies* in the dependency tree. However, with the specification versions being decoupled, it is also important to consider the *inverse dependencies*, for compatibility with newer versions of a given specification. The changes introduced to a given specification can sometimes make it incompatible with current versions of its inverse dependencies (specifications that uses the given specification).

Since this specification is one such specification that is used by other specifications released by the IC CIO, the [Figure 2](#) has been included to assist readers in understanding all of the inverse dependency relationships and how changes in this given specification may impact others specifications. This diagram is representative of direct and transitive inverse dependencies at the time of the release of this specification, but are subject to change over time and is presented in a list format that is different than [Figure 1](#).

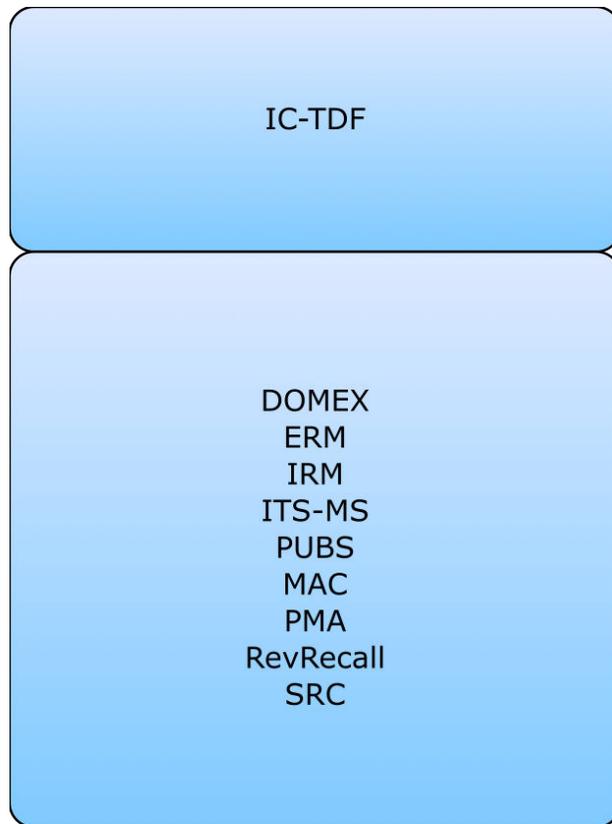


Figure 2 : Inverse Dependency Specifications

Chapter 2 - Development Guidance

For information on the structure and content of the specifications, please see the “Specification Overview” chapter in the IC-SF.XML^[4] framework document. This chapter is intended to expand upon the common information that the framework specifies providing specific development guidance that is specific to the implementation of this specification.

2.1 - TDF Structure

IC-TDF.XML is derived from *XML Data Encoding Specification for Trusted Data Format - Base* (BASE-TDF.XML^[2]) with the following differences:

- Requires Enterprise Data Header (EDH), Access Rights and Handling (ARH)
- Adds optional Revision Recall
- Requires HandlingAssertion

The IC-TDF.XML specification has a consistent and simple concept of Assertions and Payloads. For more information on the TDF Structure, please see the “TDF Structure” section of the “Development Guidance” chapter in BASE-TDF.XML^[2].

There are two options for root elements: Trusted Data Object (TDO) and Trusted Data Collection (TDC). A TDO contains some data (the Payload) and some statements about that data (the Assertions). In the context of TDF, an 'Assertion' is defined as a statement providing handling, discovery, or mission metadata describing a Payload, TDO, or TDC, depending on the scope of the Assertion. To facilitate handling and access control decisions, each TDO and TDC must contain at least two IC-EDH.XML^[3] in a **tdf:HandlingAssertion**. A **tdf:HandlingAssertion** is a special type of structured Assertion that cannot be encrypted. In general, the IC-EDH.XML^[3] **tdf:HandlingAssertion** contains the IC-EDH.XML^[3] for the TDO or Payload, providing the attributes needed for policy decisions regarding access control and how the data must be handled. The IC-EDH.XML^[3] **tdf:HandlingAssertion** defines security markings and handling controls using ISM.XML^[15]; these are the classification and other security markings for the TDO or Payload as a whole. The ISM.XML^[15] specification also defines ARH and Need-To-Know Metadata (NTK) elements with their own namespaces. ARH in a **tdf:HandlingAssertion** forms the *Access Rights and Handling* block, which contains required ISM.XML^[15] attributes and optional NTK elements. In addition to the IC-EDH.XML^[3] **tdf:HandlingAssertion**, there MAY also be an optional *XML Data Encoding Specification for Revision Recall* (RevRecall.XML^[18]) **tdf:HandlingAssertion** (see [Figure 4](#)). Additional mission Assertions (e.g., discovery metadata, source citations, records management, Document and Media Exploitation (DOMEX)) may also be provided as standard **tdf:Assertion** elements. These additional **tdf:Assertion** elements contain their own ARH blocks with required ISM.XML^[15] and optional NTK; in a mission **tdf:Assertion**, the ARH block applies to the **tdf:Assertion** itself. A TDC contains a list of TDOs (the Payload) and some statements about those TDOs (the Assertions). A TDC may also be a collection of collections, and contain other TDCs. Without impacting access control, on rare occasions, HandlingAssertion types of data MAY be inserted into a regular **tdf:Assertion**.

Each TDO consists of one or more Assertions and a Payload. TDO Assertions may optionally be cryptographically bound to the Payload to provide assurance over the integrity of the Assertion(s), the Payload, and the relationship between the Assertion(s) and Payload.

In a scenario where encryption is required, the TDO Assertion statements and/or TDO Payload may be optionally encrypted.

Each IC-TDF requires at least two IC-EDH.XML^[3] **tdf:HandlingAssertion**, optional RevRecall.XML^[18] **tdf:HandlingAssertion**, optional mission **tdf:Assertion** elements (which may be IC-CIO published Assertion specifications; e.g., *XML Data Encoding Specification for Intelligence Publications* (PUBS.XML^[17]), IRM.XML^[14], *XML Data Encoding Specification for Source Citations* (SRC.XML^[20]); or may be mission or domain-specific Assertions), and a Payload. The **tdf:HandlingAssertion** must consist of a structured IC-EDH.XML^[3] block. A common discovery Assertion might be a structured IRM.XML^[14] block. Mission specific metadata may consist of a structured block XML or unstructured data (binary). The Payload may be:

- Structured XML, implemented by the **tdf:StructuredPayload** element;
- String data, implemented by the **tdf:StringPayload** element;
- Base64Binary, implemented by the **tdf:Base64BinaryPayload** element; or
- A reference to an external Payload, implemented by the **tdf:ReferenceValuePayload** element.

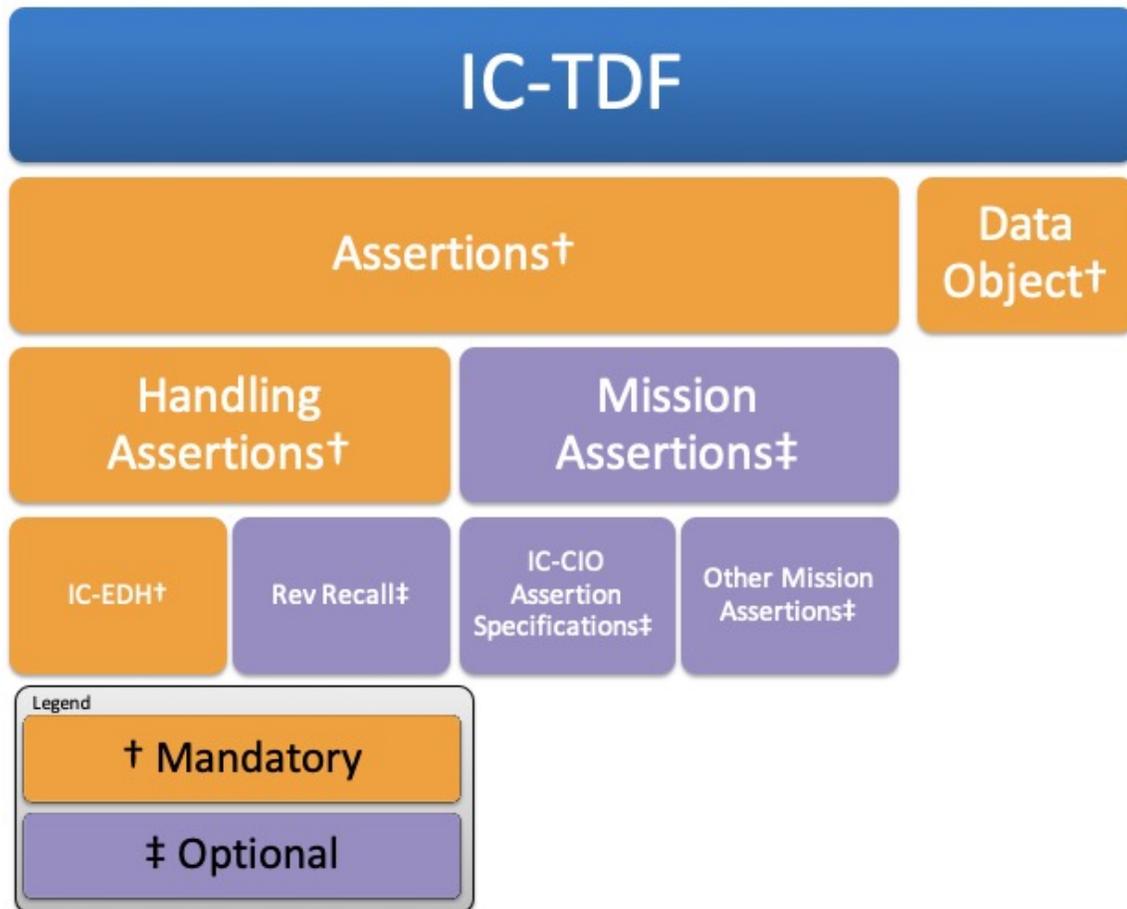


Figure 3 : TDF Structure

The diagram below shows where structural content IC specifications are used within the conceptual constructs of a TDO. The use of the IC-EDH.XML^[3] `tdf:HandlingAssertion` and Payload are required, whereas the RevRecall.XML^[18] `tdf:HandlingAssertion` and the mission-specific `tdf:Assertion` elements are optional.

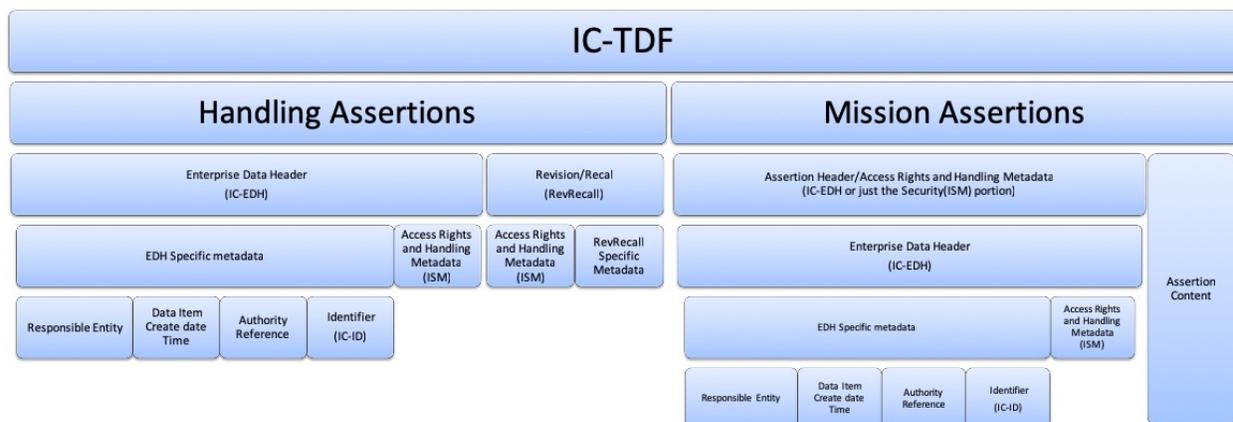


Figure 4 : TDF Detailed Structure

A TDC consists of a collection of TDOs or TDCs.

2.1.1 - Version Declarations

Specification versions are generally declared at the highest level of the XML structure that makes sense for its usage, generally either the root, or the first level of element that uses a specification. As such, many specifications used in a TDF are generally declared at the root (i.e. ISM.XML^[15], NTK, IC-EDH.XML^[3], etc.).

As extension points, Assertions and Payloads may have different versions of specifications specified for use inside itself. For example, the TDF may declare an ISM.XML^[15] `@ism:DESVersion` of "201412" while the Payload might be a legacy document that declares the ISM.XML^[15] `@ism:DESVersion` to be "9". In this case that Payload would be validated with ISM.XML.v9.

However, it is also possible that an Assertion or Payload does not contain declarations for versions of specifications. In this case they are considered to be the same versions that are declared in the TDF. That is, the extension points inherit specification versions from the TDF in which they reside and, if they are extracted from the TDF, those version declarations MUST be copied into that content during extraction to maintain validity as well as comprehensibility.

2.2 - Assertions

2.2.1 - Assertion Scopes

Assertions can be scoped to apply to different portions of an IC-TDF.XML instance. Several Assertion scopes imply certain meaning and processing instructions. TDF implements the concept of scope through the `@tdf:scope` attribute, which is used across all types of Assertions. The following sections explain the valid Assertion scopes for use within TDOs and TDCs and any additional processing requirements they imply.

2.2.1.1 - Assertion Scopes Within TDO

Assertions within a TDO can be scoped to apply to either the entire TDO, the Payload only, or both. For more information, please see the "Assertion Scopes within TDO" Section of the "Development Guidance" chapter in BASE-TDF.XML^[2].

2.2.1.2 - Assertion Scopes Within TDC

Assertions within a TDC can be scoped to apply to several different portions of a TDC instance. [Definition: The child TDOs and TDCs contained within a TDC are referred to as the *collection members*.] For more information, please see the "Assertion Scopes Within TDC" Section of the "Development Guidance" chapter in BASE-TDF.XML^[2].

2.2.1.3 - Handling Assertion scopes within TDO

A TDO has at a minimum two `tdf:HandlingAssertion` elements: a TDO `tdf:HandlingAssertion` with `@tdf:scope= "TDO"`, and a Payload

tdf:HandlingAssertion with **@tdf:scope= "PAYL"**. This allows for separate access control decisions to be made for the Payload versus the entire TDO (which includes the Payload metadata). There may be an additional **tdf:HandlingAssertion** with scope **"TDO"** that contains Revision/Recall information using the RevRecall.XML^[18] specification. A **tdf:HandlingAssertion** MUST NOT be encrypted.

2.2.1.4 - HandlingAssertion scopes within TDC

A TDC can only have a single **tdf:HandlingAssertion** containing an IC-EDH.XML^[3] specification and its **@tdf:scope** must be **"TDC"**. There may also be an optional second **tdf:HandlingAssertion** with **@tdf:scope= "TDC"** that contains Revision/Recall information for the TDC using the RevRecall.XML^[18] specification. A **tdf:HandlingAssertion** MUST NOT be encrypted.

2.2.2 - Mission-Specific Metadata Assertions

Although missions may create their own unique set of Assertions, no understanding by the enterprise beyond access control is assured. For more information, please see the "Mission-Specific Metadata Assertions" Section of the "Development Guidance" chapter in BASE-TDF.XML^[2].

2.2.3 - Assertions and Data State

If an Assertion statement or a Payload is encrypted, then there are in fact two (potentially different) markings needed for decision making, analysis, and querying: one for describing the handling required for the ciphertext and the other for the handling required for the unencrypted (and in effect external) state. For more information, please see the "Assertions and Data State" Section of the "Development Guidance" chapter in BASE-TDF.XML^[2].

In cases where statements and/or Payloads are encrypted, **tdf:HandlingAssertion** elements and **tdf:StatementMetadata** elements indicate whether their marks apply to the ciphertext vs. plaintext by using the attribute **@tdf:appliesToState**. This attribute may be leveraged in use cases such as:

- A user or system knows that they are not allowed to have/process data with NTK *systemXYZ*, and the user/system wants to query a large IC cloud repository and filter out results that require *systemXYZ* handling. For results with encrypted Payloads, if the **tdf:HandlingAssertion** only reflects the ciphertext handling (say Confidential) the user/system could get back thousands of encrypted results they cannot decrypt, should not see, and do not want to sort.
- Agency X publishes data to the IC cloud with encrypted Payloads. In a decrypted state, the Payload requires NTK markings that IC cloud cannot yet handle access-wise. In this case, when the markings in an Assertion have **@tdf:appliesToState="encrypted"**, they should be part of rollup and used for the handling of the TDO. When the markings in an Assertion have **@tdf:appliesToState="unencrypted"**, they should be excluded from rollup, and used for search filtering, or access and processing decisions in systems that are able to decrypt the Payload.

2.3 - Binding and BindingInfo

A key concept in the TDF specification is the ability to cryptographically assure the relationship among portions of the document. This assurance is represented by the optional `tdf:Binding` element available on each `tdf:Assertion` and `tdf:HandlingAssertion`. For more information, please see the “Binding and BindingInfo” Section of the “Development Guidance” chapter in BASE-TDF.XML^[2].



Note

Within IC-TDF.XML, `tdf:HandlingStatement` refers to an IC-EDH.XML^[3] instance (`edh:Edh` or `edh:ExternalEdh`).

2.4 - Normalization Method

The normalization method expressed in `tdf:Binding/tdf:SignatureValue/@tdf:normalizationMethod` and `tdf:Binding/tdf:BoundValueList/tdf:BoundValue/@tdf:normalizationMethod` is a Uniform Resource Identifier (URI) that provides guidance on how to format the included values such as whitespace, attributes, and child nodes in a universally consistent manner. For more information, please see the “Normalization Method” Section of the “Development Guidance” chapter in BASE-TDF.XML^[2].

2.5 - Encryption and EncryptionInfo

A key concept in the TDF specification is the ability to encrypt Payloads, Assertions, and keys. For more information, please see the “Encryption and EncryptionInfo” Section of the “Development Guidance” chapter in BASE-TDF.XML^[2].

2.6 - Linked or Embedded Data Objects

Linked objects classification does NOT impact the classification of the TDO. Embedded objects classification does impact the classification of the TDO.

2.7 - MIME type

For information on Media Type (MIME) type, please see the “MIME type” section of the “Development Guidance” chapter in BASE-TDF.XML^[2].

2.8 - BASE-TDF Schematron Rules

BASE-TDF schematron rules should be used for validation in derived child TDF instances (i.e. an IC-TDF.XML instance should be validated against BASE-TDF and IC-TDF.XML schematron rules).

Chapter 3 - Constraints

3.1 - Data Validation Constraint Rules

3.1.1 - Purpose

The IC-TDF.XML schema defines the data elements, attributes, cardinalities and parent-child relationships for which XML instances must comply. Validation of these syntax aspects is an important first step in the validation process. An additional level of validation is needed to ensure that the content complies with the constraints as specified in applicable IC policy guidance and codified in these constraint rules. Traditional schema languages are generally unable to effectively represent these additional constraints. For more information, please see the “Data Validation Constraint Rules” chapter in the IC-SF.XML^[4] framework document.

3.1.2 - Inherited Constraints

In an instance of IC-TDF.XML, the use of attributes and elements from supplementary data encoding specifications must be fully conformant with the constraint rules defined in those specifications. For a full list of supplementary specifications, see [Section 1.5 - Dependencies](#).

3.1.3 - Value Enumeration Constraints

IC-TDF.XML currently does not contain any CVEs.

3.1.4 - Additional Constraints

3.1.4.1 - DES Constraints

The DES version is specified through attributes on the root element. The schema constrains the values of these attributes. The `@DESVersion` attribute enables systems processing an instance document to be certain which set of constraint rules, schema, CVEs and business rules are intended by the author to be used.

3.1.5 - Constraint Rules

The detailed constraint rules for the IC-TDF.XML schema can be found in a separate document inside the Documents/IC-TDF directory, in the “IC-TDF_Rules.pdf” file. This document is generated from the individual Schematron files to provide a single searchable document for all of the constraint rules encoded in Schematron. Obsolete rule numbers are listed in the “IC-TDF_Rules.pdf” file.

3.2 - Data Rendering Constraint Rules

3.2.1 - Purpose

Rendering rules define constraints on the rendering and display of IC-TDF.XML documents. The intent is to inform the development of systems capable of rendering or displaying IC-TDF.XML

data for use by individuals not familiar with the details of the IC-TDF.XML markup. While expressed in a similar manner to the data validation constraint rules above, there is no expectation that evaluation of these rules can be automated; rather these rules should inform the evaluation of a system's capabilities and functionality.

3.2.2 - Rendering Constraint Rules

The following table contains the information for the IC-TDF.XML data rendering constraint rules.

Table 3 - Constraint Rules

Rule Number	Severity	Description	Human Readable Description
There are no Data Rendering Constraint rules at this time.			

Chapter 4 - Conformance Validation

An instance is considered conformant with the IC-TDF.XML specification if it passes all of the following normative validation steps. The following steps do not dictate how this validation strategy is implemented.

4.1 - Definitions

Terms are defined the first time they are used. Definitions are cumulative, meaning that a term used in any given step may be defined in a previous step. The definitions for IC-TDF.XML are defined in the “Definitions” section of the “Conformance Validation” chapter in BASE-TDF.XML^[2].

4.2 - Why a verbose validation strategy is required

The IC-TDF.XML specification is designed to be extremely flexible by allowing users to include several formats of in-line content in several extension points. For more information on why these *TDF extension points* require IC-TDF.XML instances to use a more verbose validation strategy, please see the “Why a verbose validation strategy is required” section of the “Conformance Validation” chapter in BASE-TDF.XML^[2].

4.3 - How to determine the ISM version within structured content

The IC-TDF.XML specification uses ISM.XML^[15] for conveying classification markings. The PUBS.XML^[17] specification also uses ISM.XML^[15]. Suppose the Payload contained an old PUBS.XML^[17] document, which used a different version of ISM.XML^[15] than defined in the IC-TDF.XML specification. Applying the version of ISM.XML^[15] business rules defined in IC-TDF.XML to this instance document could easily fail because the older version ISM.XML^[15] markings in the PUBS.XML^[17] document could contain different attributes, removed tokens, among other changes. The version of ISM.XML^[15] markings used within *structured content* is determined by the first occurrence of attribute `@ism:DESVersion` in document order contained in the structured content. If the structured content does not specify attribute `@ism:DESVersion`, then the ISM.XML^[15] version is defined to be the same as the ISM.XML^[15] markings used within the parent IC-TDF.XML structure (TDO or TDC).

4.4 - Required Order of Handling Assertions

Before any validation takes place on a TDO, a validation implementation MUST ensure that the TDO `tdf:HandlingAssertion` is the first `tdf:HandlingAssertion` in document order. The required order is detailed in the “Required Order of Handling Assertions” section of the “Conformance Validation” chapter in BASE-TDF.XML^[2].



Note

[Definition: The ISM.XML^[15] business rules define the first element in document order which specifies attribute `@ism:resourceElement="true"` to be the *resource element*.] The resource element contains the banner level ISM.XML^[15] markings for the entire instance (i.e., the “roll-up”).

4.5 - TDO Validation Steps

This section outlines the required steps to fully validate a TrustedDataObject (TDO).

4.5.1 - Step 1 - TDO aware and cross Assertion constraints

This step is intended to support validation which requires knowledge of the TDO structure.

IC-TDF.XML validation, to include schema and business rules, should be run during this step.

ISM.XML^[15] and NTK validation MUST NOT be run in this step because, as explained in the justification above, a *TDF extension point* could contain *structured content* which contains ISM.XML^[15] or NTK markings from a different version of ISM.XML^[15]/NTK than the TDO structure is using, which could fail validation. ISM.XML^[15] and NTK validation is performed in [Section 4.5.3 - Step 3 – TDO structure constraints](#). ARH and IC-EDH.XML^[3] validation SHOULD NOT be performed at this step as it may be problematic when dealing with extension points that utilize different versions of these specifications from those used in the TDO.

TDO aware validation MAY be performed during this step. For example, one might want to run business rules specific to a certain domain or system. Some examples of custom validation could include:

- If this TDO contains an Assertion with child element X, then it must also contain a peer Assertion with child element Y.
- Verify that this TDO instance contains a custom Assertion specific to a certain domain.
- Verify all bindings within this TDO.
- If the Payload is encrypted, attempt to decrypt it and run additional custom validation on the decrypted content.

4.5.2 - Step 2 – Extension point constraints

This step is intended to support validation for the content of all *TDF extension points* contained within the TDO.

The child content of any *TDF extension point* MAY be validated. Any content validated in this step MUST be validated independently and in isolation. Determining which *TDF extension points* are validated in this step is implementation specific. For example, an implementation might choose to only validate *structured content* while ignoring *binary content* and *string content* completely. Or, an implementation might define a configuration which only validates *structured content* whose root element is in a certain namespace or set of namespaces.

If the content being validated is *structured content*, then the ISM.XML^[15] business rules MUST NOT be applied unless the content is a *standalone ISM.XML^[15] document*. [Definition: A *standalone ISM document* is an XML document which specifies the ISM.XML^[15] attributes `@ism:resourceElement` and `@ism:DESVersion`.] Any NTK, ARH, or IC-EDH.XML^[3] validation SHOULD be performed during this step for the *structured content* if the appropriate `@DESVersion` attributes are specified.

Several examples of validation which could occur in this step include:

Schema and business rules for IC specifications from the 2012-Charlie release and earlier, including PUBS.XML^[17] and IRM.XML^[14].

Schema and business rules for mission specific Assertion statements.

Custom validation for an audio/video file contained within a binary Payload.

4.5.3 - Step 3 – TDO structure constraints

This step is intended to verify that ISM.XML^[15] markings within the *TDO structure* are consistent. By treating *structured content* within *TDF extension points* as black boxes, only the ISM.XML^[15] markings within the *TDO structure* will be validated. This includes ISM.XML^[15] markings within **tdf:HandlingAssertions** and **tdf:StatementMetadata**. It does not include ISM.XML^[15] markings within the Payload and Assertion extension points, which are considered 'black box' extensions in this step. This is also the time when any NTK, ARH, and IC-EDH.XML^[3] validation that is specific to the *TDO structure* itself SHOULD be performed.

If IC-TDF.XML rules were not run in Step 1:

[Definition: A *placeholder element* is an XML element whose localname is "PlaceholderContent", namespace is "urn:placeholder", and contains no text content or child elements.]

[Definition: A *TDF skeleton* is an IC-TDF.XML instance in which the structured content contained within all TDF extension points has been replaced by a placeholder element.] Whether *string content* and *binary content* is preserved when converting an IC-TDF.XML instance to a TDF skeleton is implementation specific. Replacing string content and binary content with default values may yield performance improvements during validation if that content is large in size and is not intended to be validated.

[Definition: A *TDF skeleton* whose root element is **tdf:TrustedDataObject** is referred to as a *TDO skeleton*.]

The **tdf:TrustedDataObject** element MUST be converted into a *TDO skeleton*, which MUST be validated in isolation against the normative portions of the ISM.XML^[15] specification version in use by the TDO. Additional validation MAY be performed during this step.

4.5.4 - Step 4 – ISM consistency constraints

This step is intended to verify that ISM.XML^[15] markings contained within *structured content* match the corresponding ISM.XML^[15] markings within the *TDO structure*. This step has several sub-steps because Assertions and Payloads require slightly different processing depending upon certain criteria.

4.5.4.1 - Step 4a – Consistency constraints for Assertions with resource level portion markings

[Definition: An *Assertion fragment* is a **tdf:Assertion** element containing at least one **tdf:StatementMetadata** element and a TDF extension point.] Whether an Assertion fragment contains any other child elements (**tdf:Binding**, **tdf:ReferenceList**, etc) is implementation specific.

[Definition: A *structured Assertion fragment* is an Assertion fragment whose TDF extension point is `tdf:StructuredStatement`.]

Structured Assertion fragments meeting the following criteria MUST be validated in isolation against the normative portions of the ISM.XML^[15] specification version in use by the TDO:

1. The *structured content* contains ISM.XML^[15] markings.
2. The ISM.XML^[15] markings contained in the *structured content* are from the same version of the ISM.XML^[15] specification as the Information Security Markings (ISM) markings within the *TDO structure*. See [Section 4.3 - How to determine the ISM version within structured content](#).
3. One of the `tdf:StatementMetadata` child elements specifies attribute `@ism:resourceElement="true"`.

Validation of a *structured Assertion fragment* verifies that the ISM.XML^[15] markings contained within the *structured content* and the ISM.XML^[15] markings contained within the `tdf:StatementMetadata` element are consistent. The ISM.XML^[15] business rules use the `tdf:StatementMetadata` ISM markings as the resource level ("banner level") markings and treat the ISM markings in the *structured content* as portion markings. Constraint #3 above ensures that a `tdf:StatementMetadata` element can provide the resource level markings required for the ISM.XML^[15] business rules.

For example, if the `tdf:StatementMetadata` contained `@ism:classification="U"` and the TDF extension point content contained `@ism:classification="TS"`, then the ISM.XML^[15] business rules would throw an error saying that unclassified documents must not contain TS portions.

4.5.4.2 - Step 4b – Consistency constraints for non EDH Handling Assertions with resource level portion markings

[Definition: A *Handling Assertion fragment* is a `tdf:HandlingAssertion` element containing at least one `arh:Security` element.] Whether an Assertion fragment contains any other child elements (`tdf:Binding`, `tdf:ReferenceList`, etc) is implementation specific.

[Definition: A *structured Handling Assertion fragment* is a Handling Assertion fragment whose contents contain ISM markings.]

Structured Handling Assertion fragments meeting the following criteria MUST be validated in isolation against the normative portions of the ISM.XML^[15] specification version in use by the TDO:

1. The `tdf:HandlingStatement` contains ISM.XML^[15] markings.
2. The ISM.XML^[15] markings contained in the `tdf:HandlingStatement` are from the same version of the ISM.XML^[15] specification as the ISM markings within the *TDO structure*. See [Section 4.3 - How to determine the ISM version within structured content](#).
3. One of the `tdf:HandlingStatement` child elements specifies attribute `@ism:resourceElement="true"`.

Validation of a *structured Handling Assertion fragment* verifies that the ISM.XML^[15] markings contained within the `tdf:HandlingStatement` are consistent. The ISM.XML^[15] business rules use the `arh:Security` ISM markings as the resource level (“banner level”) markings and treat the ISM markings in the *structured Handling Assertion fragment* as portion markings. Constraint #3 above ensures that a `tdf:HandlingStatement` element can provide the resource level markings required for the ISM.XML^[15] business rules.

For example, if the `tdf:HandlingStatement` contained `@ism:classification="U"` and the *structured Handling Assertion fragment* content contained `@ism:classification="TS"`, then the ISM.XML^[15] business rules would throw an error saying that unclassified documents must not contain TS portions.

4.5.4.3 - Step 4c – Consistency constraints for Payloads with resource level portion marking

[Definition: A *Payload fragment* is a `tdf:TrustedDataObject` element containing a single `tdf:HandlingAssertion` element which is the Payload `tdf:HandlingAssertion` and a child TDF extension point.] Whether a Payload fragment contains any other child elements (`tdf:Assertion`, etc) is implementation specific.

[Definition: A *structured Payload fragment* is a Payload fragment whose TDF extension point is `tdf:StructuredPayload`.]

Structured Payload fragments meeting the following criteria MUST be validated in isolation against the normative portions of the ISM.XML^[15] specification version in use by the TDO.

1. The *structured content* contains ISM.XML^[15] markings.
2. The ISM markings contained in the *structured content* are from the same version of the ISM.XML^[15] specification as the ISM markings within the *TDO structure*. See [Section 4.3 - How to determine the ISM version within structured content](#).
3. The *Payload tdf:HandlingAssertion* specifies attribute `@ism:resourceElement="true"`.

Validation of the *structured Payload fragment* verifies that the ISM.XML^[15] markings contained within the *structured content* are consistent with the ISM.XML^[15] markings in the *Payload tdf:HandlingAssertion*. The ISM.XML^[15] business rules use the *Payload tdf:HandlingAssertion* as the resource level (“banner level”) markings and treats the ISM.XML^[15] markings in the *structured content* as portion markings. Constraint #3 above ensures that the *Payload tdf:HandlingAssertion* can provide the resource level markings required for the ISM.XML^[15] business rules.

For example, if the *Payload tdf:HandlingAssertion* contained `@ism:classification="U"` and the *structured content* contained `@ism:classification="TS"`, then the ISM.XML^[15] business rules would throw an error saying that unclassified documents must not contain TS portions.

4.5.4.4 - Step 4d – Consistency constraints for Assertions and Payloads with non-resource level markings

This step is intended to check the consistency of ISM.XML^[15] markings within Assertions and Payloads which do not have corresponding resource level ISM.XML^[15] portion markings in the TDO structure (Assertions and Payloads not checked in step 4a, 4b, or 4c).

The `tdf:TrustedDataObject` element MUST be modified to replace *structured content* meeting the following criteria with a *placeholder element*:

1. The *structured content* contains ISM.XML^[15] markings.
2. The ISM markings contained within the *structured content* are from a *different version* of the ISM.XML^[15] specification as the ISM markings within the *TDO structure*. See [Section 4.3 - How to determine the ISM version within structured content](#).

The modified `tdf:TrustedDataObject` element MUST be validated in isolation against the normative portions of the ISM.XML^[15] specification version in use by the TDO.

Replacing all of the *structured content* containing ISM.XML^[15] markings from different versions allows the ISM.XML^[15] business rules for the version used within the TDO structure to run correctly. The ISM.XML^[15] business rules will use the `tdo tdf:HandlingAssertion` as the resource level (“banner level”) markings and treat the ISM.XML^[15] markings in the rest of the TDO as portion markings. This step is very similar to [Section 4.5.3 - Step 3 – TDO structure constraints](#), but step 3 replaces all *structured content* with a *placeholder element* whereas this step leaves *structured content* in-line if it uses the same ISM.XML^[15] version as the ISM markings within the TDO structure.

For example, if the *TDO tdf:HandlingAssertion* contained `@ism:classification="U"` and the *structured content* of an Assertion not checked in step 4a, 4b, or 4c (using the same ISM.XML^[15] version) contained `@ism:classification="TS"`, then the ISM.XML^[15] business rules would throw an error saying that unclassified documents must not contain TS portions.

4.6 - TDC Validation Steps

This section outlines the required steps to fully validate a TrustedDataCollection (TDC).

4.6.1 - Step 1 – TDC aware and cross Assertion constraints

This step is intended to support validation which requires knowledge of the TDC structure.

IC-TDF.XML validation to include schema and business rules should be run during this step.

ISM.XML^[15] validation MUST NOT be run in this step because, as explained in the justification above, a *TDF extension point* could contain *structured content* which contains ISM markings from a different version of ISM.XML^[15] than the TDC structure is using, which could fail validation. ISM.XML^[15] validation is performed in [Section 4.6.3 - Step 3 – TDC structure constraints](#). NTK, ARH, and IC-EDH.XML^[3] validation at this step may also be problematic when dealing with extension points that utilize versions of these specifications used in the TDO.

Additional validation may be performed during this step. For example, one might want to run business rules specific to a certain domain or system. Some examples of custom validation could include:

- Test if this TDC contains an Assertion with child element X, then it must also contain a peer Assertion with child element Y.
- Test if this TDC must contain a certain Assertion type, such as a Multi Audience Collection (MAC) Assertion.

4.6.2 - Step 2 – Extension point constraints

This step is intended to support validation for the TDF extension point content contained within child `tdf:Assertion` elements of the TDC. The rules outlined in [Section 4.5.2 - Step 2 - Extension point constraints](#) should be applied to each child `tdf:Assertion` element of the `tdf:TrustedDataCollection` element.

4.6.3 - Step 3 – TDC structure constraints

This step is intended to verify that ISM.XML^[15] markings within the TDC structure are consistent. By treating *structured content* within *TDF extension points* as black boxes, only the ISM.XML^[15] markings within the TDC structure will be validated. This includes ISM.XML^[15] markings within `tdf:HandlingAssertions` and `tdf:StatementMetadata`. This is also the place to perform any NTK, ARH, and IC-EDH.XML^[3] validation that is specific to the TDC structure itself.

[Definition: A TDF skeleton whose root element is `tdf:TrustedDataCollection` is referred to as a *TDC skeleton*.]

The `tdf:TrustedDataCollection` element MUST be converted into a *TDC skeleton*, which MUST be validated in isolation against the normative portions of the ISM.XML^[15] specification version in use by the TDC. Additional validation MAY be performed during this step.

4.6.4 - Step 4 – ISM consistency constraints

This step is intended to verify that ISM.XML^[15] markings contained within *structured content* match the corresponding ISM.XML^[15] markings within the TDC structure. This step has several sub-steps because Assertions with resource level ("banner level") ISM.XML^[15] markings require slightly different processing than non-resource level ISM.XML^[15] markings.

4.6.4.1 - Step 4a – Consistency constraints for Assertions with resource level portion markings

This step is intended to verify the consistency of ISM.XML^[15] markings contained within child `tdf:Assertion` elements of the `tdf:TrustedDataCollection` element. The rules outlined in [Section 4.5.4.1 - Step 4a – Consistency constraints for Assertions with resource level portion markings](#) should be applied to each child `tdf:Assertion` element within the TDC.

4.6.4.2 - Step 4b – Consistency constraints for Assertions with non-resource level markings

This step is intended to check the consistency of ISM.XML^[15] markings within child `tdf:Assertion` elements which do not have corresponding resource level ISM.XML^[15] portion markings in the TDC structure (Assertions not checked in step 4a).

The `tdf:TrustedDataCollection` element MUST be modified to replace *structured content* meeting the following criteria with a *placeholder element*:

1. The *structured content* contains ISM.XML^[15] markings.
2. The ISM markings contained within the *structured content* are from a different version of the ISM.XML^[15] specification as the ISM markings within the TDC structure. See [Section 4.3 - How to determine the ISM version within structured content](#).

The modified `tdf:TrustedDataCollection` element MUST be validated in isolation against the normative portions of the ISM.XML^[15] specification version in use by the TDC.

Replacing all of the *structured content* containing ISM.XML^[15] markings from different versions allows the ISM.XML^[15] business rules for the version used within the TDC structure to run correctly. The ISM.XML^[15] business rules will use the tdc `tdf:HandlingAssertion` as the resource level (“banner level”) markings and treat the ISM.XML^[15] markings in the rest of the TDC as portion markings.

For example, if the TDC `tdf:HandlingAssertion` contained `@ism:classification="U"` and the structured content of an Assertion not checked in step 4a (using the same ISM.XML^[15] version) contained `@ism:classification="TS"`, then the ISM.XML^[15] business rules would throw an error saying that unclassified documents must not contain TS portions.

4.6.5 - Step 5 - Recursive Validation

A `tdf:TrustedDataCollection` element supports recursion by allowing child `tdf:TrustedDataObject` and `tdf:TrustedDataCollection` elements. Each `tdf:TrustedDataObject` element must be validated according to the steps outlined in [Section 4.5 - TDO Validation Steps](#). Each `tdf:TrustedDataCollection` element must be validated according to the steps outlined in [Section 4.6 - TDC Validation Steps](#).

Appendix A Feature Summary

The following table summarizes major features by version for this TDF and all dependent specs.

Table 4 - Feature Summary Legend

Key	Description
F	Full (able to comply and verified by spec to some degree)
P	Partial (Able to comply but not verifiable)
N	Non-compliance (Can't comply)
N/A	Not Applicable. Feature is no longer required.
Cell Colors represent the same information as the Key value	

A.1. IC-TDF Feature Summary

A.1.1. Features from V2014-DEC to V2021-NOV

Table 5 - IC-TDF Feature comparison V2014-DEC to V2021-NOV

Required date	Feature	V2014-DEC	V2014-DECr2017-JUL	V2019-MAR	V2021-NOV
	Multiple versions of ISM.XML ^[15] (V9 - 2016-SEP r2018-JUL)	F	F	N/A	N/A
	Multiple versions of NTK.XML (V7 - v2016-SEP)	F	F	N/A	N/A
	Multiple versions of ARH.XML (V1 - V3)	F	F	N/A	N/A
	Support ISM.XML ^[15] 2019-MAR (incorporated ARH/NTK) and later	N	N	F	F
	Depends on BASE-TDF.XML ^[2]	N	N	N	F

A.1.1.1. Features Partial and N/A from V2014-DEC to V2021-NOV

Table 6 - IC-TDF Feature comparison V2014-DEC to V2021-NOV

Required date	Feature	V2014-DEC	V2014-DECr2017-JUL	V2019-MAR	V2021-NOV
	Multiple versions of ISM.XML ^[15] (V9 - 2016-SEP r2018-JUL)	F	F	N/A	N/A
	Multiple versions of NTK.XML (V7 - v2016-SEP)	F	F	N/A	N/A
	Multiple versions of ARH.XML (V1 - V3)	F	F	N/A	N/A
	TDC scope "PAYL"	N/A	N/A	N/A	N/A

A.1.2. Features from V1 to V2014-DEC

Table 7 - IC-TDF Feature comparison V1 to V2014-DEC

Required date	Feature	V1	V2	V3	V2014-DEC
	Multiple versions of ISM.XML ^[15] (V9 - 2016-SEP-2018-JUL)	N	F	F	F
	Multiple versions of NTK.XML (V7 - v2016-SEP)	N	F	F	F
	Multiple versions of ARH.XML (V1 - V3)	N	F	F	F
	Multiple versions of IC-EDH.XML ^[3] (V1 - Current)	N	F	F	F
	TDC scope "PAYL"	F	N/A	N/A	N/A
	TDC scopes "DESC_TDO", "DESC_PAYL", and "TDC_MEMBER"	N	F	F	F
	Multiple bindings in Assertions and <code>tdf:HandlingAssertions</code>	N	F	F	F
	Version decoupling, allowing import of any version of ISM.XML ^[15] and other dependent specifications at or above ISM.XML ^[15] v9+, NTK.XML ^[16] v7+, ARH.XML ^[1] v1+, and IC-EDH.XML ^[3] v1+	N	F	F	F
	Vector encryption	N	N	N	F

A.1.2.1. Features Partial and N/A from V1 to V2014-DEC

Table 8 - IC-TDF Feature comparison V1 to V2014-DEC

Required date	Feature	V1	V2	V3	V2014-DEC
	TDC scope "PAYL"	F	N/A	N/A	N/A

Appendix B Change History

The following table summarizes the version identifier history for this DES.

Table 9 - DES Version Identifier History

Version	Date	Purpose
1	July 17, 2012	Initial Release
2	January 21, 2013	Routine revision to technical specification. For details of changes, see Section B.6 - V2 Change Summary
3	August 16, 2013	Routine revision to technical specification. For details of changes, see Section B.5 - V3 Change Summary
2014-DEC	December 4, 2014	Routine revision to technical specification. For details of changes, see Section B.4 - V2014-DEC Change Summary
2014-DEC-r2017-JUL	July 21, 2017	Routine revision to technical specification. For details of changes, see Section B.3 - V2014-DEC-r2017-JUL Change Summary
2019-MAR	March 8, 2019	Routine revision to technical specification. For details of changes, see Section B.2 - 2019-MAR Change Summary
2021-NOV	December 3, 2021	Routine revision to technical specification. For details of changes, see Section B.1 - 2021-NOV Change Summary

B.1 - 2021-NOV Change Summary

Significant drivers for Version 2021-NOV include:

- Community Change Requests

The following table summarizes the changes made to V2019-MAR in developing 2021-NOV.

Table 10 - Data Encoding Specification 2021-NOV Change Summary

#	Change	Artifacts changed	Compatibility Notes
1	Rule numbers missing from CVE and Schema checks. (CR-2019-083)	Schematron	No impact to systems.

#	Change	Artifacts changed	Compatibility Notes
2	Added a new "Step 4b – Consistency constraints for non EDH handling Assertions with resource level portion markings" to reduce the risk of a non EDH handling assertion having inconsistent security markings. (CR-2017-207)	Documentation	Implementations of TDF validation need to add another sub-step to step 4.
3	Updated rows in the Dependency Table to point to the appropriate Authoritative Source. (CR-2020-014)	Documentation	No impact to systems.
4	Add HashVerification to ReferenceValueType (CR-2019-014)	Documentation Schema	No impact to systems.

#	Change	Artifacts changed	Compatibility Notes
5	Update IC-TDF to use BASE-TDF (CR-2019-058, CR-2019-019)	CVE CVEnum-TDFAppliesToState removed. CVEnum-TDFHashAlgorithm removed. CVEnum-TDFSignatureAlgorithm removed. Documentation Schema Schematron IC-TDF-ID-00001 deleted IC-TDF-ID-00002 deleted IC-TDF-ID-00006 deleted IC-TDF-ID-00007 deleted IC-TDF-ID-00008 deleted IC-TDF-ID-00009 deleted IC-TDF-ID-00010 deleted IC-TDF-ID-00011 deleted IC-TDF-ID-00012 deleted IC-TDF-ID-00013 deleted	Systems need to be updated to accommodate this change.

#	Change	Artifacts changed	Compatibility Notes
		IC-TDF-ID-00014 deleted	
		IC-TDF-ID-00015 deleted	
		IC-TDF-ID-00025 deleted	
		IC-TDF-ID-00032 deleted	
		IC-TDF-ID-00035 deleted	
		IC-TDF-ID-00038 deleted	
		IC-TDF-ID-00039 deleted	
		IC-TDF-ID-00040 deleted	
		IC-TDF-ID-00041 deleted	
		IC-TDF-ID-00058 added	
6	Fix bug in ReferenceValueType to allow hashes without chunking and add IC-SF environment validation rule (CR-2021-008)	Documentation Schema Schematron IC-TDF-ID-00057 added	Data generation and ingestion systems need to be updated.

B.2 - 2019-MAR Change Summary

Significant drivers for Version 2019-MAR include:

- Community Change Requests

The following table summarizes the changes made to V2014-DECr2017-JUL in developing 2019-MAR.

Table 11 - Data Encoding Specification 2019-MAR Change Summary

#	Change	Artifacts changed	Compatibility Notes
1	Cleanup obsolete rules after ARH and NTK consolidation into ISM (CR-2018-095).	Schematron IC-TDF-ID-00037 deleted. IC-TDF-ID-00047 deleted. IC-TDF-ID-00048 deleted.	Systems need to be updated to accommodate this change.
2	Added ISM.XML attributes to Schematron files to mark up the documentation. (CR-2017-303)	Schematron	No impact to systems.
3	Create RelaxNG CVE Fragments for IRM and update CVE Schema Version. (CR-2017-173)	CVEs	No impact to systems.
4	Added schema PDF. (CR-2018-015)	Documentation	No impact to systems.
5	Changed "Multipurpose Internet Mail Extensions" to "Media Type". (CR-2018-055)	Documentation	No impact to systems.
6	Updated CSV generation to include a column for deprecation date information. (CR-2018-081)	CSV	Systems using CSVs no longer have to look to the XML or JSON for the deprecation date information.
7	Updated documentation to use the specification framework. (CR-2018-126)	Documentation	No impact to systems.
8	Update Schematron Rules relating to Min version to check infrastructure (CR-2018-133)	Schematron ValidateValidationEnvCV E added ValidateValidationEnvSchema added IC-TDF-ID-00036 modified IC-TDF-ID-00045 modified IC-TDF-ID-00056 added	Data validation systems need to be updated to accommodate the changes to the rules.

#	Change	Artifacts changed	Compatibility Notes
9	Fix validity of JSON-LD CVEs. (CR-2018-144)	CVE	Data generation and ingestion systems using JSON need to be updated to accommodate the changes.
10	Removed the Dependency Over Time table. (CR-2018-152)	Documentation	No impact to systems.
11	Updated the TDF Structure chapter to include verbiage regarding using a HandlingAssertion as a regular Assertion. (CR-2017-291)	Documentation	No impact to systems.

B.3 - V2014-DEC-r2017-JUL Change Summary

Significant drivers for Version 2014-DEC-r2017-JUL include:

- Community Change Requests

The following table summarizes the changes made to 2014-DEC in developing V2014-DEC-r2017-JUL.

Table 12 - Data Encoding Specification V2014-DEC-r2017-JUL Change Summary

#	Change	Artifacts changed	Compatibility Notes
1	Change the IC-TDF-ID-00046 Rule to Handle ism:DESVersion Values with dash "-" token separators.(CR-2016-081)	Schematron CompareVersionsInSkel eton revised	Allows DESVersion attribute to contain version and revision numbers separated by a "-".
2	Reorganize IC-TDF Schematron Rules Folder To Handle Deleted Rules.(CR-2016-084)	Schematron IC-TDF-ID-00020 deleted IC-TDF-ID-00021 deleted IC-TDF-ID-00022 deleted IC-TDF-ID-00023 deleted IC-TDF-ID-00024 deleted	Simplifies processing of unit tests for schematron rules.

#	Change	Artifacts changed	Compatibility Notes
3	Changed "TDO" to "TDC" in rule text of IC-TDF-ID-00005. (CR-2017-025)	Schematron IC-TDF-ID-00005 modified	Minimal impact to generation and ingestion systems.
4	Referenced the "Assertion Scopes" section in Chapter 2 of the IC-TDF DES document in the scope reference documentation. (CR-2016-008)	Documentation Schema	No impact to generation and ingestion systems.
5	Bug in TDF Rule 00014 in v2014v12; allow tdf:EncryptionInformation elements to be nested in tdf:WrappedKey elements. (CR-2017-016)	Schematron IC-TDF-ID-00014 modified	Data generation and ingestion systems need to be updated to use the modified schematron rules.
6	There were a few typos throughout the documents. The mistakes were things like "guarenteed", "encyrption", "pertinate", "identifer" and "encapslating" and others. They were all in the comment sections. (CR-2017-099)	Documentation Schema	No impact to generation and ingestion systems.
7	Added IC-TDF-ID-00055 Rule to enforce at most 1 handling Assertion scoped PAYL containing EDH for unencrypted TDO (CR-2016-037)	Schematron IC-TDF-ID-00055 added	No impact to generation and ingestion systems.
8	Create JSON version of CVEs in IC-TDF (CR-2017-054)	CVEs	No impact to systems.
9	Create CSV version of CVEs in IC-TDF (CR-2017-032)	CVEs	No impact to systems.
10	Updated tdf:version enforcement rule to be warning and handle trailing version text (CR-2017-082, CR-2017-027)	Schema Schematron IC-TDF-ID-00054 added IC-TDF_XML.sch modified	Data generation and ingestion systems need to be updated to accommodate the changes to the rules.
11	Added inverse dependency section and definitions for Dependencies and Inverse Dependencies. (CR-2017-112)	Documentation	No impact to systems.

#	Change	Artifacts changed	Compatibility Notes
12	The schema change logs will no longer be maintained as of the 2017-JUL release. The existing change logs will only serve as legacy information. For changes to schema as of and after 2017-JUL, reference the change history in the DES.	Schema	No impact to systems.
13	Added the revision constraint section since this is the first revision of ISM.	Documentation	Data generation and ingestion systems will may need to be updated to properly validate against the right revisions of specifications.
14	Updated rule IC-TDF-ID-00042 to require that the first tdf:HandlingAssertion include an EDH. (CR-2017-142)	Schematron IC-TDF-ID-00042 modified	Data generation and ingestion systems will may need to be updated to properly position a RevisionRecall Assertion.
15	Updated rule IC-TDF_ID_00017 to properly require an EDH with Scope TDC to have @ism:resourceElement="true" . This was aligning IC-TDF_ID_00017 with the existing logic in IC-TDF_ID_00016 which had done it correctly for TDOs. (CR-2017-198)	Schematron IC-TDF-ID-00017 modified	Data generation and ingestion systems will may need to be updated to properly ensure the first Assertion has @ism:resourceElement="true"
16	Enable use of ARH or EDH instead of only EDH for describing the classification of Encrypted Assertions. (CR-2017-202)	Schematron IC-TDF-ID-00030 modified	Data generation and ingestion systems will may need to be updated to properly allow and process ARH for security of encrypted Assertions.
17	Added @id and @role to all sch:rule elements, in support of commercial tools warnings and errors and to support open source unit testing frameworks. (CR-2017-216)	All non-abstract Schematron rules modified	No impact to existing systems. Additional capabilities.

#	Change	Artifacts changed	Compatibility Notes
18	Updated rule documentation to remove use of "we". (CR-2017-208)	Schematron IC-TDF-ID-00001 modified IC-TDF-ID-00002 modified IC-TDF-ID-00003 modified IC-TDF-ID-00004 modified IC-TDF-ID-00005 modified IC-TDF-ID-00006 modified IC-TDF-ID-00007 modified IC-TDF-ID-00008 modified IC-TDF-ID-00009 modified IC-TDF-ID-00010 modified IC-TDF-ID-00011 modified IC-TDF-ID-00012 modified IC-TDF-ID-00013 modified IC-TDF-ID-00014 modified IC-TDF-ID-00015 modified IC-TDF-ID-00017 modified	No impact to systems.

#	Change	Artifacts changed	Compatibility Notes
		IC-TDF-ID-00018 modified IC-TDF-ID-00019 modified IC-TDF-ID-00025 modified IC-TDF-ID-00026 modified IC-TDF-ID-00027 modified IC-TDF-ID-00032 modified IC-TDF-ID-00036 modified IC-TDF-ID-00037 modified IC-TDF-ID-00039 modified IC-TDF-ID-00041 modified IC-TDF-ID-00045 modified IC-TDF-ID-00055 modified	
19	Update prose to align with current specifications. Specifically, change e-mail address to ic-standads-support@iarpa.gov, update dependency table to standardize wording. (CR-2017-235)	Documentation	No impact to systems.
20	Update the version numbering EBNF to reflect the existence of Revisions. (CR-2017-237)	Documentation	No impact to systems.

B.4 - V2014-DEC Change Summary

Significant drivers for Version 2014-DEC include:

- Addition encryption algorithm support

The following table summarizes the changes made to V3 in developing V2014-DEC.

Table 13 - Data Encoding Specification V2014-DEC Change Summary

#	Change	Artifacts changed	Compatibility Notes
1	Changed DESVersion to represent the year and month of release. Also allowed for extension of specification by adding a '-' followed by a string to denote a custom implementation.	DES Schema Schematron IC-TDF-ID-00036 revised IC-TDF-ID-00037 revised IC-TDF-ID-00045 revised	Data generation and ingestion systems need to be updated to use the modified version numbering and schematron rules.
2	Improved encryption support by allowing for Initialization Vector, tweak, nonce, hash algorithm, Mask Generation Function, Additional Authentication Data, authentication tag, key encoding format, and Provable Data Possession wrapped keys.	Schema	Data generation and ingestion systems need to be updated to support the new components.
3	Required HandlingAssertions to come first in a TDO/TDC and required that the HandlingAssertion scoped TDO come first in a TDO.	Schema Schematron IC-TDF-ID-00042 Added	Data generation and ingestion systems need to be updated enforce the proper ordering.
4	Added rule to enforce presence of NTK at "top" level if NTK is present in any part of the TDF skeleton structure.	Schematron IC-TDF-ID-00043 Added IC-TDF-ID-00044 Added	Data generation and ingestion systems may need to be updated correctly place NTK.
5	Added Version Declarations section to describe handling and inheritance of specification versions in Assertions and Payloads.	Documentation	Data generation, ingestion, or manipulation systems may need to be updated to properly handle version declarations.

#	Change	Artifacts changed	Compatibility Notes
6	Updated Schema and Schematron rules to deal with Revision Recall handling Assertion.	Schematron Schema IC-TDF_ID_00004 Changed IC-TDF_ID_00005 Changed IC-TDF_ID_00016 Changed IC-TDF_ID_00045 Added	Data generation and ingestion systems will need to be updated to use the new Revision Recall handling Assertion.

B.5 - V3 Change Summary

Significant drivers for Version 3 include:

- Improve support of Onion encryption
- Support of Suite-B encryption

The following table summarizes the changes made to V2 in developing V3.

Table 14 - Data Encoding Specification V3 Change Summary

#	Change	Artifacts changed	Compatibility Notes
1	Updated the EncryptionInformation group to support onion encryption and Suite-B algorithms.	Documentation Schema Schematron IC-TDF-ID-00040 Added IC-TDF-ID-00041 Added	Data generation and ingestion systems need to be updated to understand the new schema structure.

#	Change	Artifacts changed	Compatibility Notes
2	Removed the value 'TDO PAYL' as an allowable value from the enumeration for the scope attribute. Removed the schematron rules that were looking for the 'TDO PAYL' scope.	Schema Schematron IC-TDF-ID-00020 Removed IC-TDF-ID-00021 Removed IC-TDF-ID-00022 Removed IC-TDF-ID-00023 Removed IC-TDF-ID-00024 Removed	Data generation and ingestion systems need to be updated to the new schema structure and to no longer enforce the schematron rules.

B.6 - V2 Change Summary

Significant drivers for Version 2 include:

- See ISM.XML^[15] V10 drivers
- See IC-EDH.XML^[3] V2 drivers

The following table summarizes the changes made to V1 in developing V2.

Table 15 - Data Encoding Specification V2 Change Summary

#	Change	Artifacts changed	Compatibility Notes
1	Added Schematron rules to require the specification of the <code>@tdf:issuer</code> attribute and either the <code>@tdf:subject</code> or <code>@tdf:serial</code> attribute for the <code>tdf:signer</code> element.	Schematron IC-TDF_ID_00038.sch	Data generation and ingestion systems need to be updated enforce the new rules.
2	Added Schematron rules to ensure that the versions of the imported specs meet the minimum allowed versions.	Schematron IC-TDF-ID-00036 Added IC-TDF-ID-00037 Added	Data generation and ingestion systems need to be updated enforce the new rules.

#	Change	Artifacts changed	Compatibility Notes
3	Updated the GUIDE id in the example files to comply with the updated regex in IC-EDH-ID-00007. The updated rule ensures there are no additional characters before or after the id.	Examples	Data generation and ingest systems complying with the GUIDE id rules do not need to be updated. Systems that were allowing invalid GUIDE ids will need to be updated to comply with the constraint rule.
4	Added validation strategy to the DES Version.	DES	Systems performing validation of the TDF should follow the appropriate validation strategy to ensure thorough and complete validation.
5	Added requirements for References to have external security markings.	IC-TDF-ID-00033 added IC-TDF-ID-00034 added	Data generation and ingest systems will be required to comply with the new rules.
6	Added scopes " DESC_TDO ", " DESC_PAYL ", and " TDC_MEMBER " for use within TDC Assertions to disambiguate trusted data collection scope meaning.	Schema IC-TDF-ID-00007 modified IC-TDF-ID-00035 added	Data generation and ingest systems will be required to comply with the new rules.
7	Deprecated scope " PAYL " for use within TDC Assertions.	IC-TDF-ID-00007 modified	Data generation and ingest systems will be required to comply with the new rules.
8	Added support for multiple bindings within Assertions and HandlingAssertions.	Schema DES	Data generation and ingest systems need to be updated to support the new schema structure.
9	Version decoupling, allowing import of any version of ISM.XML ^[15] and other dependent specifications at or above ISM.XML ^[15] v9+, NTK v7+, ARH v1+, and IC-EDH.XML ^[3] v1+.	DES	Data ingestion systems need to be aware of this change and ensure they check appropriate dependent spec versions for validation.
10	Updated Schema to ISM.XML ^[15] v10.	Schema	Updated the Schema itself to use ism:DESVersion to 10 to mark the xsd schema instance with classification markings.

#	Change	Artifacts changed	Compatibility Notes
11	Added rule to only allow HandlingAssertions with scope of Payload to use of the appliesToState attribute because only the Payload can have encrypted or unencrypted states.	Schematron IC-TDF-ID-00039 added	Data generation and ingest systems will be required to comply with the new rules, however this rule should prevent systems from having to deal with a nonsensical case.

Appendix C List of Abbreviations

This appendix lists all the acronyms and abbreviations referenced in this encoding specification.

ARH	Access Rights and Handling
CVE	Controlled Vocabulary Enumeration
DES	Data Encoding Specification
DNI	Director of National Intelligence
DOMEX	Document and Media Exploitation
EDH	Enterprise Data Header
IC	Intelligence Community
IC CIO	Intelligence Community Chief Information Officer
ICD	Intelligence Community Directive
IC EA	Intelligence Community Enterprise Architecture
IC ESB	Intelligence Community Enterprise Standards Baseline
ICPM	Intelligence Community Policy Memorandum
ICS	Intelligence Community Standard
IEC	International Electrotechnical Commission
ISM	Information Security Markings
ISO	International Organization for Standardization
MAC	Multi Audience Collection
MIME	Media Type
NTK	Need-To-Know Metadata
TDC	Trusted Data Collection
TDF	Trusted Data Format
TDO	Trusted Data Object
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
XML	Extensible Markup Language

Appendix D Bibliography

[1] ARH.XML

Office of the Director of National Intelligence. *XML Data Encoding Specification for Access Rights and Handling (ARH.XML)*.

Available online Intelink-TS at: <https://go.ic.gov/FTSj6AO> (case sensitive – Foxtrot Tango Sierra juliet 6 Alpha Oscar)

Available online Intelink-U at: <https://w3id.org/ic/standards/ARH>

Available online at: <https://w3id.org/ic/standards/public>

[2] BASE-TDF.XML

Office of the Director of National Intelligence. *XML DES Encoding Specification for Trusted Data Format - Base (BASE-TDF.XML)*.

Available online Intelink-TS at: <https://go.ic.gov/GC4VEXo> (case sensitive – Golf Charlie 4 Victor Echo Xray oscar)

Available online Intelink-U at: <https://w3id.org/ic/standards/BASE-TDF>

Available online at: <https://w3id.org/ic/standards/public>

[3] IC-EDH.XML

Office of the Director of National Intelligence. *XML Data Encoding Specification for Enterprise Data Header (IC-EDH.XML)*.

Available online Intelink-TS at: <https://go.ic.gov/5Pg1r8s> (case sensitive – 5 Papa golf 1 romeo 8 sierra)

Available online Intelink-U at: <https://w3id.org/ic/standards/EDH>

Available online at: <https://w3id.org/ic/standards/public>

[4] IC-SF.XML

Office of the Director of National Intelligence. *Intelligence Community Specification Framework (IC-SF.XML)*.

Available online Intelink-TS at: <https://go.ic.gov/pNFyuVg> (case sensitive – papa November Foxtrot yankee uniform Victor golf)

Available online Intelink-U at: <https://w3id.org/ic/standards/IC-SF>

Available online at: <https://w3id.org/ic/standards/public>

[5] ICD 208

Office of the Director of National Intelligence. *Write For Maximum Utility*. Intelligence Community Directive 208. 17 December 2008.

Available online at: http://www.dni.gov/files/documents/ICD/icd_208.pdf

[6] ICD 209

Office of the Director of National Intelligence. *Tearline Production and Dissemination*. Intelligence Community Directive 209. 6 September 2012.

Available online at: <http://www.dni.gov/files/documents/ICD/ICD%20209%20Tearline%20Production%20and%20Dissemination.pdf>

[7] ICD 500

Office of the Director of National Intelligence. *Director of National Intelligence Chief Information Officer*. Intelligence Community Directive 500. 7 August 2008.

Available online Intelink-TS at: <https://go.ic.gov/U7v6ZRL> (case sensitive – Uniform 7 victor 6 Zulu Romeo Lima)

Available online at: http://www.dni.gov/files/documents/ICD/ICD_500.pdf

[8] ICD 501

Office of the Director of National Intelligence. *Discovery and Dissemination or Retrieval of Information within the Intelligence Community*. Intelligence Community Directive 501. 21 January 2009.

Available online Intelink-TS at: <https://go.ic.gov/fTBM8OS> (case sensitive – foxtrot Tango Bravo Mike 8 Oscar Sierra)

Available online at: http://www.dni.gov/files/documents/ICD/ICD_501.pdf

[9] ICD 710

Office of the Director of National Intelligence. *Classification Management and Control Markings System*. Intelligence Community Directive 710. 21 June 2013.

Available online Intelink-TS at: <https://go.ic.gov/oSj9K7O> (case sensitive – oscar Sierra juliet 9 Kilo 7 Oscar)

Available online at: http://www.dni.gov/files/documents/ICD/ICD_710.pdf

[10] ICPM 2007-200-2

Office of the Director of National Intelligence. *Preparing Intelligence to Meet the Intelligence Community's Responsibility to Provide*. Intelligence Community Policy Memorandum 2007-200-2. 11 December 2007.

Available online at: <http://www.dni.gov/files/documents/IC%20Policy%20Memos/ICPM%202007-200-2%20Responsibility%20to%20Provide.pdf>

[11] ICPM 200-01

Office of the Director of National Intelligence. *Intelligence Community Standards and Procedures for Revised or Recalled Intelligence Products*. Intelligence Community Policy Memorandum 2020-200-01. 27 February 2020.

Available online at: https://www.dni.gov/files/documents/ICPM_2020_200-01_U-FOUO_SIGNED-FINAL_Redacted.pdf

[12] ICS 500-20

Director of National Intelligence Chief Information Officer. *Intelligence Community Enterprise Standards Compliance*. Intelligence Community Standard 500-20. 16 December 2010.

Available online Intelink-TS at: <https://go.ic.gov/kh8NMVJ> (case sensitive – kilo hotel 8 November Mike Victor Juliet)

Available online Intelink-U at: <https://w3id.org/ic/standards/policy/ICS500-20>

[13] ICS 500-21

Director of National Intelligence Chief Information Officer. *Tagging of Intelligence and Intelligence-Related Information*. Intelligence Community Standard 500-21. 28 January 2011.

Available online Intelink-TS at: <https://go.ic.gov/0Agmenr> (case sensitive – 0 Alpha golf mike echo november romeo)

Available online Intelink-U at: <https://w3id.org/ic/standards/policy/ICS500-21>

[14] IRM.XML

Office of the Director of National Intelligence. *XML Data Encoding Specification for Information Resource Metadata (IRM.XML)*.

Available online Intelink-TS at: <https://go.ic.gov/pOKLbmx> (case sensitive – papa Oscar Kilo Lima bravo mike xray)

Available online Intelink-U at: <https://w3id.org/ic/standards/IRM>

Available online at: <https://w3id.org/ic/standards/public>

[15] ISM.XML

Office of the Director of National Intelligence. *XML Data Encoding Specification for Information Security Markings (ISM.XML)*.

Available online Intelink-TS at: <https://go.ic.gov/qoNICy7> (case sensitive – quebec oscar November India Charlie yankee 7)

Available online Intelink-U at: <https://w3id.org/ic/standards/ISM>

Available online at: <https://w3id.org/ic/standards/public>

[16] NTK.XML

Office of the Director of National Intelligence. *XML Data Encoding Specification for Need-To-Know Metadata (NTK.XML)*.

Available online Intelink-TS at: <https://go.ic.gov/6wFIZpE> (case sensitive – 6 whiskey Foxtrot India Zulu papa Echo)

Available online Intelink-U at: <https://w3id.org/ic/standards/NTK>

Available online at: <https://w3id.org/ic/standards/public>

[17] PUBS.XML

Office of the Director of National Intelligence. *XML Data Encoding Specification for Intelligence Publications (PUBS.XML)*.

Available online Intelink-TS at: <https://go.ic.gov/u6bb18P> (case sensitive – uniform 6 bravo bravo 1 8 Papa)

Available online Intelink-U at: <https://w3id.org/ic/standards/PUBS>

Available online at: <https://w3id.org/ic/standards/public>

[18] REVRECALL.XML

Office of the Director of National Intelligence. *XML Data Encoding Specification for Revision Recall (RevRecall.XML)*.

Available online Intelink-TS at: <https://go.ic.gov/cC4WFa0> (case sensitive – charlie Charlie 4 Whiskey Foxtrot alpha 0)

Available online Intelink-U at: <https://w3id.org/ic/standards/REVRECALL>

Available online at: <https://w3id.org/ic/standards/public>

[19] Schematron

International Organization for Standardization (ISO). *Information technology -- Document Schema Definition Language (DSDL) -- Part 3: Rule-based validation -- Schematron*. ISO/IEC 19757-3:2006.

ISO Spec Available online at: <http://standards.iso.org/ittf/PubliclyAvailableStandards/index.html>

StyleSheets for compiling Available online at: <http://code.google.com/p/schematron/>

[20] SRC.XML

Office of the Director of National Intelligence. *XML Data Encoding Specification for Source Citations (SRC.XML)*.

Available online Intelink-TS at: <https://go.ic.gov/IFrOEsU> (case sensitive – lima Foxtrot romeo Oscar Echo sierra Uniform)

Available online Intelink-U at: <https://w3id.org/ic/standards/SRC>

Available online at: <https://w3id.org/ic/standards/public>

[21] XSLT2

World Wide Web Consortium (W3C). *XSL Transformations (XSLT) Version 2.0*. W3C Recommendation 23 January 2007.

Available online at: <http://www.w3.org/TR/xslt20/>

Appendix E Points of Contact

The Intelligence Community Chief Information Officer (IC CIO) facilitates one or more collaboration and coordination forums charged with the adoption, modification, development, and governance of IC technical specifications of common concern. This technical specification was produced by the IC CIO and coordinated with these forums, approved by the IC CIO or a designated representative, and made available at the following Director of National Intelligence (DNI)-sponsored web sites.

Public Website: <https://w3id.org/ic/standards/public>

Intelshare: <https://w3id.org/ic/standards/data-specs>

Direct all inquiries about this IC technical specification, IC technical specification collaboration and coordination forums, or IC element representatives involved in those forums, to the IC CIO.

E-mail: ic-standards-support@odni.gov.

Appendix F IC CIO Approval Memo

An IC CIO Approval Memo should accompany this enterprise technical data specification bearing the signature of the IC CIO or an IC CIO-designated official(s). If an IC CIO Approval Memo is not accompanying this specification's version release package, then refer back to the authoritative web location(s) for this specification to see if a more complete package or a specification update is available.

Specification artifacts display a date representing the last time a version's artifacts as a whole were modified. This date most often represents the conclusion of the IC Element collaboration and coordination process. Once the IC Element coordination process is complete, the specification goes through an internal IC CIO staffing and coordination process leading to signature of the IC CIO Approval Memo. The signature date of the IC CIO Approval Memo will be later than the last modified date shown on the specification artifacts by an indeterminable time period.

Upon signature of the IC CIO Approval Memo, IC Elements may begin to use this specification version in order to address mission and business objectives. However, it is critical for IC Elements, prior to disseminating information encoded with this new specification version, to ensure that key enterprise services and consumers are prepared to accept this information. IC Elements should work with enterprise service providers and consumers to orchestrate an orderly implementation transition to this specification version in concert with mandatory and retirement usage decisions captured in the Intelligence Community Enterprise Standards Baseline (IC ESB) as defined in ICS 500-20^[12].