



ECSS

ECSS DOORS Database Specification

Change Log

Version - Date	Comment
DOORS Task Force Report, Version 1.0, 01 Feb. 2012	Document provided for TA review/endorsement
DOORS Task Force Report, Version 2, 19 March 2015	Report updated to take into account requested modifications to the ECSS DOORS database before its first release.
DOORS Task Force Report, Version 3, 8 June 2015	Update during DOORS TF meeting 8 June 2015
DOORS Task Force Report, Version 4, 15 June 2015	Update requested from DOORS TF on 8 June, implemented by ES 15 June 2015.
ECSS-D-00-03C 7 May 2020	<p>This document is based on the former "ECSS DOORS Task Force Report".</p> <p>The ECSS Technical Authority decided to rename this document to fit the purpose of a database specification.</p> <p>Following changes were implemented:</p> <ul style="list-style-type: none">• Document name changed to "ECSS DOORS Database Specification" and ECSS number "ECSS-D-00-03" assigned.• Update Information about Object attributes to include the new attribute OLE Counter.• Updated information about the new attribute OUID.• Added information on how to treat deleted requirements.• Update section related with the supported DOORS version.• Updated the name of the ECSS modules "Rev.1" to "Rev. 1" and "corr-1" to "Corr. 1".

Table of content

1. Introduction	5
1.1 Executive Summary of the ECSS DOORS Expert Group	5
1.2 Executive Summary of the former ECSS DOORS Task Force	5
2. Definitions and acronyms	7
2.1 Definitions	7
2.2 Acronyms	8
3. Initial considerations	9
3.1 Hypothesis and constraints	9
3.2 Proposed approach	10
3.3 Recommendations for use of the ECSS DOORS Database	10
4. Solution for Publication of ECSS standards in the DOORS format	11
4.1 Overview	11
4.2 Inputs	11
4.3 Definition of DOORS attributes	11
4.3.1 Attributes for elements - identifier and content	11
4.3.2 Attributes for versioning and change tracking	13
4.3.3 Attributes for identifying relationships between different objects	16
4.3.4 Attribute for exporting objects' content	16
4.3.5 User-defined attributes from the existing industry/CNES DOORS database	17
4.3.6 Attributes for Database Problem Reports management	17
4.3.7 Summary of the attributes' definition	18
4.4 Definition of module "ECSS standard list"	20
4.5 DOORS modules and database organisation	20
4.6 DOORS Database Problem Reports (DPR) management	22
4.7 Definition of DOORS Views	23
4.8 Database consistency rules	24
4.9 Outputs	25
4.10 Processes	26
4.10.1 Proposition for update and verification - Mock-up phase	26
4.10.2 Database update and validation	26
4.10.3 Database release	26
4.10.4 Database maintenance	26
4.11 Planning & resources	27
4.11.1 Opportunities	27
4.11.2 Risks already mitigated	27
5. Considerations for potential evolutions of the ECSS DOORS database	28
6. Conclusion	28
ANNEX A ECSS standards – reference/version and "ECSS rank"	29

ANNEX B	Trade-off for requirement numbering.....	31
ANNEX C	Trade-off for “notes” and “expected output”	34
ANNEX D	Details of the mechanism for change tracking.....	36
ANNEX E	Library of scripts	40

1. Introduction

1.1 Executive Summary of the ECSS DOORS Expert Group

The ECSS Technical Authority decided in 2017 to create a new Group, called "ECSS DOORS Expert group" to further improve the usability of the ECSS DOORS database by all stakeholders.

The conclusions of this group led to amend the former "ECSS DOORS Task Force Report" into this document.

The following Executive Summary is kept as background information of the former ECSS DOORS Task Force

1.2 Executive Summary of the former ECSS DOORS Task Force

Both space agencies and industries manage requirements, including ECSS ones, as part of project activities. Often this task is performed making use of requirement management tools, e.g. databases. Therefore, it was considered beneficial for improving daily work to export the ECSS standards in DOORS format since DOORS is a requirement management application widely used amongst the European space actors.

Currently, two entities (ECSS Executive Secretariat and AIRBUS DS/TAS/CNES) have already imported the whole content of the ECSS standards in two slightly different databases. Anticipating on implementation and consolidation issues, the ECSS Steering Board decided to setup a dedicated Task Force.

The mandate of this Task Force was to determine the core characteristics needed for building an ECSS endorsed DOORS Database enabling requirements management both for agencies and industries, allowing its application in project development. This encompassed the following tasks, as defined by agreed TOR:

- Analysis of the needs of the various stakeholders prior defining the ECSS DOORS Database (including clarification of some hypothesis and constraints)
- Identification of core characteristics needed for the ECSS DOORS Database maintained and distributed by ECSS secretariat in support of the ECSS users
- Identification of potential synergies and exchange of data/tools between the two developments to avoid duplicating efforts
- Proposal of a structure and data model for an ECSS DOORS Database, together with its configuration control.

The mandate was restricted to the definition of the ECSS DOORS Database and excluded its implementation or defining solutions impacting the current ECSS process for publication and maintenance of the ECSS standards.

Therefore, in line with its mandate, the TF describes in this report:

- The consensus achieved for defining an ECSS DOORS Database. The solution proposed accounting of discussion and trade-off performed (e.g. organization, attributes, views, configuration control) is detailed and is provided for TA approval prior implementation.

- An approach for implementation of the proposed solution for creating and maintaining the ECSS DOORS database, together with a preliminary planning and identification of resources and risks/opportunities.

Finally, resulting from activities and discussions which took place, the TF identified areas for extending the TOR of the TF, e.g. to have a better integration and maintenance of both the ECSS DOORS Database and the Standards in MS Word format. These activities are of utmost importance in preparation for issue D. Therefore, it is expected to get TA approval for investigating upon those additional activities, as briefly described in the document.

2. Definitions and acronyms

2.1 Definitions

This list covers both general DOORS-related terms, as well as terms used by the ECSS DOORS Database Administrator. DOORS terms are identified in bold, terms defined by the TF are identified in italics.

A DOORS Database is structured in **projects**, which are made of several **folders** themselves constituted of **modules**. In the case of the ECSS DOORS database, the main terms are defined as follows:

Archive	<p>DOORS solution to group information (at project or module level) for distribution/exchange between two DOORS databases.</p> <p>Note: Successful exchange of information is only possible if the Archive is created using a DOORS major version lower or equal to the DOORS version of the database where the Archive is to be extracted.</p>
Attribute	<p>Information added to a DOORS object (e.g. for a requirement: ID number, version)</p> <p>Note: Some attributes are available by default in the system and others are specific to the ECSS DOORS database.</p>
Correction	<p>Correction of the ECSS DOORS database to resolve errors not detected prior to releasing the database. A typical example could be the correction of a mathematic symbol which has been incorrectly imported from MS Word into DOORS.</p>
Folder	<p>Sub-structure of project (e.g. ECSS M), used for organizing the database content</p>
Interleaved Note	<p>Note that is in the middle of the text of a normative provision.</p> <p>Note: Interleaved Notes trigger errors during the import of a MS Word file into a DOORS module and should be corrected during revision of an ECSS Standard.</p>
Link	<p>DOORS function implemented in one module or between two modules for identifying the relationship between two objects. In the ECSS DOORS Database, usage of links is restricted to links within one module / standard to document where a Table/Figure is called by a requirement or a requirement is called by another requirement.</p>
Normative provision	<p>General term used in ECSS for “Requirements”, “Recommendations” and “Permissions”</p> <p>Note: The clause “Nomenclature” in each ECSS Standard defines in detail the use of the three kinds of normative provisions.</p>
Module	<p>Element of folder which in the ECSS DOORS database contains the content of an ECSS standard. A Module is built of Objects.</p>

Object	Type of information contained in a module (Depending on the specific attribute values, an object represents a heading, general information, a requirement, a Table, a Figure, ...)
Project	Sub-structure of a DOORS database (e.g. ECSS current) created for organizing information for configuration purpose (e.g. creation of archive)
Script	Programming code for automating given actions on the content of the DOORS database (e.g. adding links between normative provisions and Figure in a module)
View	DOORS solution to select/filter available information in a module and present it (e.g. requirement review view). Elements of a view are the list and order of attributes displayed, optionally a filter only showing a subset of objects fulfilling a given criteria, optionally a sorting applied to the objects based on specific criteria, and optionally additional dynamic information retrieved from linked objects.

2.2 Acronyms

Corr	Corrigendum
CR	Change Request
DOORS	Distributed Object Orientated Requirements System (Name of requirement management tool (IBM® Rational® DOORS®))
DPR	Database Problem Report
ES	ECSS Executive Secretariat
ID or Id	Identifier
IE PUID	Project Unique Identifier (Name of ID compatible with RMF (RMF is an IBM add-on of DOORS))
OUID	Object Unique Identifier
RCM	Requirement Configuration Management
Rev	Revision
NA	Not Applicable

3. Initial considerations

3.1 Hypothesis and constraints

Existing implementations

Before the creation of the ECSS DOORS database, two entities (ECSS executive secretariat and AIRBUS DS/TAS/CNES) had already imported the whole content of the ECSS standards in two slightly different databases. To minimise the disruption of the database version existing in each entity, it was decided that the ECSS DOORS endorsed database:

- takes as initial source for enhancement the ECSS ES database version,
- takes due account of the existing implementation in the AIRBUS DS/TAS/CNES database (referenced later in the document as Industry/CNES DOORS database) and lessons learned.

Database version

DOORS is downward compatible, i.e. a DOORS archive generated under a smaller DOORS version number can be extracted into a database running in a newer version with a higher version number. As the opposite is not possible, and in order to allow extracting into as many existing databases as possible, it was confirmed that the ECSS DOORS database will always be made available under the lowest (oldest) DOORS version still under support by IBM. Currently, this is DOORS version 9.1.

DOORS versions used by ECSS partners in August 2019:

AIRBUS	DOORS 9.6
CNES	DOORS 9.3.0.5, plan to upgrade in 2020 to DOORS 9.7
DLR (GSOC)	DOORS 9.6
ESA / ECSS Executive Secretariat	DOORS 9.1, plan to upgrade to DOORS 9.6
TAS	DOORS 9.5, plan to upgrade to DOORS 9.6

ECSS Standards versions

With respect to the content of the databases, both entities had imported only ECSS Standards at version C and above. Currently, there are 2 documents in ECSS system still at issue A (ECSS-E-20-01A Rev.1 and ECSS-M-70A) , which will be updated and republished at version C following ECSS drafting rules or withdrawn.

It was confirmed:

- That there is no need to complement the ECSS DOORS database with standards at issue A or B.
- Since the only ECSS documents containing normative text are the standards, the Handbook and technical memoranda are not imported in the DOORS database.

The list of ECSS documents and the identification of their version can be found in the Module “ECSS Standard List” that is part of the ECSS DOORS database. The structure of this list is shown in ANNEX A .

Equivalence of all data formats for delivery of ECSS standards

Importing the content of the ECSS standards in DOORS in a unique database version allows ECSS to deliver these documents in a new format (DOORS archive, in addition to the existing formats MS Word and PDF) for feeding users' tools for requirement management. Therefore, the content of the ECSS DOORS database shall guaranty conformity with the published ECSS standards.

Existing processes

As many processes for publication and maintenance of ECSS standards are established and widely used, it was confirmed:

- that the use of ECSS standards shall remain possible without having to use the ECSS DOORS Database, and
- that the preparation, delivery and maintenance of the ECSS DOORS Database shall not impact the established ECSS processes for publication and maintenance of standards.

3.2 Proposed approach

Taking into account the objectives, constraints and the results of discussions, which took place during ECSS Task Force meetings; it was decided to organize the tasks in two steps:

- #1 – tasks to address issues for proposing an initial database design and process for publishing the ECSS standards in the DOORS format (see section 9), and
- #2 – tasks to investigate on potential evolutions linked to the use of the ECSS DOORS database (see section 10).

3.3 Recommendations for use of the ECSS DOORS Database

Providing the ECSS standards in DOORS is an alternative to ECSS standards in Word or PDF format. Which ECSS standard format to be used in a project is up to the user to decide.

The DOORS tool is a database that provides easier tracking and tailoring of requirements, and the DOORS format of the ECSS requirements is therefore recommended to be used in space projects

However, many ECSS requirements are process related, e.g. for instance requirements or recommendations on how to perform a certain task or specific process, as opposed to functional or performance related requirements. It should be noted that, in line with ECSS-E-ST-10-02 "Space engineering – Verification", the fact that ECSS makes a DOORS version of the ECSS standards available does not impose a recommendation for the usage of DOORS for the verification of process related requirements. This in fact would be considered as a potential major cost driver.

4. Solution for Publication of ECSS standards in the DOORS format

4.1 Overview

This section details the agreed design and way forward for establishing the ECSS endorsed DOORS database. It contains a high level description of the DOORS attributes and views added to the ES Database, and proposed processes and planning for implementation.

4.2 Inputs

For the creation of the first ECSS DOORS database the following main inputs were considered:

- the ES DOORS Database used as the basis for building the first ECSS DOORS Database
- ECSS Word / PDF files (for validation of the ECSS DOORS database)
- EXCEL tables (as extracted from Industry/CNES DOORS database) for each ECSS standard listing:
 - Requirement IDs, ECSS requirement identification (as per document) and requirement text
 - Requirement ID for Table and Figure called by a requirement and ECSS Table or Figure title
- A Document listing the ECSS standards together with their 3-digit <Document rank> as used in the Requirement IDs (see section 4.3.1.2).

NOTE: This document was the basis for the Module “ECSS Standard List”.

4.3 Definition of DOORS attributes

4.3.1 Attributes for elements - identifier and content

4.3.1.1 Syntax of IE PUID and OUID numbers

All normative provisions (i.e. requirements, recommendations and permissions) and Table and Figures called by normative provisions are identified with a unique IE PUID number

All non-normative objects are identified with the attribute OUID.

- Syntax of an IE PUID number for normative elements:

<ECSS Document prefix>_<ECSS Document rank><normative Running number>

(e.g. ECSS-Q-ST-30-11_0140001)

- Syntax of an OUID number for non-normative elements:

<ECSS Document prefix>_<ECSS Document rank><non-normative Running number>#

(e.g. ECSS-Q-ST-30-11_0140015#)

Where:

- **<ECSS Document prefix>** is the prefix of the document in which the object is located (e.g. in the example above: **ECSS-Q-ST-30-11**). This information is at start of the object ID in order to help the users when the object is used in document (e.g. project document or company document).
- **<ECSS Document rank>** is a 3 digit number which refers to document where the object has been created (e.g. in the example above: **014**). The values that this field can take are defined in ANNEX A . Note that this number is only different from the respective number corresponding to the <Document prefix> when an object has been moved from one standard to another.
- **<normative Running number>** is an integer number of 4 digits that identifies the requirement number within each document. It is an incremental number unique only to the DOORS module, starting from 0001 with an increment of one for each new requirement (e.g. in the example above: **0015**).
- **<non-normative Running number>** is an integer number of 4 digits that identifies the object number within each document. It is an incremental number unique only to the DOORS module, starting from 0001 with an increment of one for each new non-normative object (e.g. in the example above: **0001**).

NOTE Since in the Industry/CNES database, the numbers are coded with 3 digits, a 0 will need to be added in the correspondence table provided as an input

- #-symbol used as visible identifier to distinguish the OUID from an IE PUID.

For the implementation of these attributes two dedicated DOORS attributes called "IE PUID" and "OUID" were created. For all normative objects the column for the OUID will be filled with a copy of the IE PUID in the DOORS module.

In case a non-normative object is moved within a document the OUID is moved with the object and is kept unchanged.

NOTE For the first creation of the ECSS DOORS database, the content of the IE PUID attributes was not generated in the ECSS DOORS database itself but was imported from the Industry/CNES database (to avoid creating inconsistency since the IE PUID as already built in Industry/CNES DOORS database is not always consecutive).

Currently the creation of new DOORS modules or updates of existing modules is done using dedicated DOORS Scripts developed for the ES by company WIDETAILE. See ANNEX E for description of these scripts.

4.3.1.2 Normative Objects

In the Word / PDF versions of the ECSS standards, the requirements, permissions and recommendations are identified and numbered following the respective clause numbers. These identifiers are neither invariant to document re-structuring, nor do they provide an identifier which is unique over the whole set of standards. It was therefore decided to add a unique, invariant requirement identifier attribute.

To avoid impact with respect to the existing mechanism for requirement identification as defined and used in the Industry/CNES Database, the IE PUID (Requirement ID) is constituted of three fields and is built up as described in 9.3.1.1.

4.3.1.3 Tables and Figures called by normative provisions

Any Table or Figure called by at least one normative provision will have a unique IE PUID built with the same mechanism as for the Requirement ID (see section 9.3.1.1). This ID does not contain an indication of the object type (e.g. Table or Figure) and is not discriminated from IE PUID for normative objects.

The object pointed remains of the type Table or Figure (thus is not from the type Requirement).

4.3.1.4 Notes and expected outputs to normative provisions

Notes and expected outputs are informative text providing clarification or examples to a normative provision. Compliance to Notes and expected output is not expected. In the first issues of the ECSS DOORS Database Notes were part of the requirement object's "Object Text" attribute and are identified by indentation and the starting word "NOTE".

For Revisions of ECSS Standards all interleaved Notes shall be moved to the end of the text of the normative provision so that all Notes are shown in the column "ECSS Notes" of the respective DOORS module.

To avoid misuse of notes when generating the compliance matrix it was requested to extract the notes' text and move them in a dedicated attribute to the requirement. However, with respect to the published set of ECSS standards, this is not always possible since some Notes are interleaved with the requirement itself. Therefore, the accepted solution was to create a dedicated attribute called "ECSS Notes", which contains both notes and expected outputs and filled in as follows:

1. For an ECSS standard which contains at least one requirement where (a) note(s) is/are interleaved with the requirement's text, then the text of the Note stays all requirements remains as in the ECSS DOORS database, and the attribute "ECSS Notes" remains empty.
2. Else, the Note(s) and Expected output will be moved in the "ECSS Notes" attribute applying a script as defined in ANNEX E , and the corresponding text will be removed from the "Object Text" attribute.

The justification for keeping both information together comes from the fact that "expected outputs" are of seldom use in ECSS standards (only in E-ST-40 and Q-ST-80) and their use elsewhere is not used.

4.3.1.5 Information text with verbal forms: shall, should, may in descriptive text

As part of the descriptive text in some ECSS standards, the verbal forms: "shall", "should", "may" and "need not" are unduly used and shall be corrected during a revision of the respective ECSS Standard.

4.3.2 Attributes for versioning and change tracking

4.3.2.1 Overview

Between two consecutive versions of an ECSS standard (e.g. from "Issue C" to "Issue C Rev. 1") it is important to keep track of changes. This is particularly true to identify changes to a normative provision. For tracking these changes two attributes are used. The attribute "RCM Version" to identify the version of the normative provision and the attribute "ECSS Change Status" to identify a change in the DOORS module.

RCM Versions are maintained for normative provisions and for Figures and Tables referenced by them.

ECSS Change Status is maintained for all objects.

4.3.2.2 RCM Version of the normative provision (a.k.a. Requirement version)

When a normative provision is created, its attribute “RCM Version” will have the version number 1 and will be incremented if the normative provision undergoes modifications in the next issue of the standard.

Therefore, the DOORS module which corresponds to the first version of an ECSS standard imported in the ECSS DOORS Database will have all normative provisions at version 1.

The attribute “RCM Version” is also filled in for Tables and Figures called by normative provisions.

4.3.2.3 ECSS Change status

The attribute “ECSS Change Status” can have the following values:

- **Created:** an object (either normative or not) is created
- **Normative Change:** the text of a normative provision, or a Figure or Table referenced by a normative provision has changed
 - Note: Modifications to a Note of a normative provision do not trigger a change of the RCM version and are only marked as “Informative Change”.
- **Informative Change:** a change is done on a non-normative information. This includes the following changes:
 - the text of an “ECSS Note” of a normative provision has changed.
 - an interleaved note of a normative provision is moved to the attribute “ECSS Notes” while the requirement itself remains unchanged.
 - the text of an object which is neither a normative provision, nor a Figure or Table referenced by a normative provision has changed.
 - changes in non-normative text, e.g. titles.
- **Deleted:** when the content of normative information is deleted
 - Note: Deletion of non-normative objects are not tracked by the "ECSS Change Status" object attribute. The complete object is removed under the condition that deleting a complete clause does not renumber the following objects. Therefore, in the case the deletion of the non-normative object would result in renumbering of the following clauses only the heading of the non-normative object is kept with its content replaced by “<<deleted>>”.
- **Unchanged:** when no modification was made to an object in a revised module.

The "ECSS Change Status" attribute value is always w.r.t. the previously published version of any standard.

ANNEX D details the mechanism for managing evolutions of these two attributes (“RCM Version” and “ECSS Change Status”).

When a normative provision (resp. Table/Figure called by a one) is deleted, in addition to updating the attribute “ECSS Change Status”, the content of the text (resp. the Caption of the Table or Figure called by a normative provision) is replaced by ”<<deleted>>”.

Moving a normative provision, without changing the text, inside the standard is performed as a combination of “<<Deleted>>” (of the old normative provision) and “Created” of a new object (with a new IE PUID at version 1) when such move has an impact on the requirement application (e.g. the requirement application linked to the context of the section where the requirement is located). Else, neither the IE PUID, change status, nor the version are modified.

When a normative object is moved to a different standard:

- in the old standard, its text will be replaced with “<<deleted and moved to <new ID>”
- in the new standard, the ID will be <document prefix new standard>_<old document rank number><running number new standard>

4.3.2.4 Correction status

This attribute has been suppressed and replaced by the following attributes: “Reason for update” (see 9.3.2.6) and “Correction number” (see 9.3.2.7).

4.3.2.5 Cross references to Figure and Tables

The principles of tracking described in 9.3.2.2, 9.3.2.3, 9.3.2.4 applies also for Figures and Tables called by normative provisions as identified in ANNEX D . That means::

- when a normative Figure/Table is modified, the Figure/Table version is increased and the calling normative provision alongside (to be updated manually in DOORS)
- when a cross-reference to a Table/Figure is modified, but the referenced Table/Figure is not modified, then the requirement version is unchanged and the ECSS Change Status is marked as “Informative Change” (to be updated manually in DOORS)

NOTE The version number of the requirement and the element called (e.g. Table, Figure or another requirement) are managed independently.

- when a Table or Figure called by requirement is modified, the version of the requirement is also updated.

4.3.2.6 Reason for update

Any correction in the database results either from a CR or from a DPR. Whenever the attribute “ECSS change status” contains a value other than “Unchanged”, the attribute “Reason for update” will be filled in with one of the two following values:

- CR if the change is due to an evolution in the ECSS standard,
- DPR if the change is due to an error identified in the existing data base.

4.3.2.7 Correction number

The purpose of the attribute “Correction number” is to identify the CR and/or DPR which have induced a modification in the database.

In the case where the correction is due to a DPR, this attribute has to be initialised with the corresponding DPR reference from the DOORS module “ECSS DPR list”.

4.3.3 Attributes for identifying relationships between different objects

To reflect some relationships existing between different information in ECSS standards, it is proposed to build up DOORS links within each module to identify the relationship between two requirements in this module or the relationship between a requirement and Table/Figure.

Links will be established such that the target object is the referenced requirement, figure or table, while the source object is the referencing requirement. If a requirement has outgoing links, then it is referencing a different requirement, Table or Figure. If a requirement, Table or Figure has incoming links, then it is referenced by at least one other requirement.

The links will go through a dedicated link module called “References”.

This relationship will be stored in:

- the dedicated links. For information and visualization, the “Overview” View will contain a column titled "Cross-references" displaying the referencing / referenced objects IDs: “in link” and “out link” using a dedicated script as defined in ANNEX E).
- one physical attribute called “**ECSS Referenced ID**” that contains the list of Doors “Absolute Numbers” of the targeted objects. This attribute is populated when the module is delivered/exported using the script in ANNEX E and allows the recipient of the module file archive (.dma) to recreate the links in their DB.

Note: In the ECSS Referenced ID, the separator will be defined such that it is possible to use the IBM build in script “link by attribute”.

4.3.4 Attribute for exporting objects’ content

To allow some rough export of information in MS Word and thus to ease the validation of the ECSS DOORS Database, the attribute “Paragraph Style” will be created to store a reduced number of ECSS styles as defined in the published word document. The content of this attribute will be defined as shown in Table 4-1:

Table 4-1: Content of attribute “Paragraph Style”

Object in ES Database	Paragraph style	implementation in DOORS
Acronym	paragraph	<Object Text:paragraph>
Bibliography	paragraph	<Object Text:paragraph>
Definition	Heading + Level	<Object Heading:Heading level#>
Definition content	paragraph	<Object Text:paragraph>
Figure	paragraph	<Object Text:paragraph>
Heading	Heading + Level Note: The Level is identified in the built-in “Object level”	<Object Heading:Heading level#>
Information	paragraph	<Object Text:paragraph>
Normative reference	paragraph	<Object Text:paragraph>
Requirement	Requirement	<Object Text:Requirement><ECSS

Object in ES Database	Paragraph style	implementation in DOORS
		Notes:Note>
Table	paragraph	<Object Text:paragraph>
Cover	OLE object	<Object Text:paragraph>
Foot Note	Paragraph	<Object Text:paragraph>
Recommendation	Requirement	<Object Text:requirement>
Permission	Requirement	<Object Text:requirement>
Note: for Table and Figure, the way to generate the list of tables and figures is still pending.		

4.3.5 User-defined attributes from the existing industry/CNES DOORS database

The following user-defined attributes already exist in the industry/CNES DOORS and ECSS database, and will also be maintained in the new ECSS DOORS database:

ECSS Object Type: with possible values *Acronym, Bibliography, Definition, Figure, Heading, Table, Requirement...*

ECSS Section: containing the original section number in the published Word / PDF version of the standard

ECSS Source ID: containing the original section number or requirements identifier in the published Word / PDF version of the standard

ECSS Source Reference: containing the document ID of the original published Word / PDF version of the standard.

4.3.6 Attributes for Database Problem Reports management

A dedicated module will be created to manage Database Problem Reports (DPRs). The name of the module will be "ECSS DPR List".

The description of the module will be: "lists all the DPR impacting ECSS standards".

The DPRs will be stored in DOORS DB that will be located in the ECSS Website and it will be visible by all users.

Table 4-2 describes the attributes which are necessary to manage the DPR process. The DPR process is fully described at the end of this section.

Table 4-2: Attributes of the ECSS DPR List

#	Attribute name	Attribute type	Complement on content	Column name & content
1.	ECSS DPR Reference	Text	To be filled by ES	ECSS DPR Reference
2.	ECSS DPR initiator		To be filled by DPR initiator	
3.	ECSS DPR Name	Text	To be filled by DPR initiator.	ECSS DPR Name
4.	ECSS DPR Description	Text	To be filled by DPR initiator	ECSS DPR Description

#	Attribute name	Attribute type	Complement on content	Column name & content
5.	ECSS DPR origin standards	Text	To be filled by DPR initiator Lists the ECSS modules and list of affected objects (if relevant) concerned by the DPR.	ECSS DPR origin standards
6.	ECSS original DB version	Text	To be filled by DPR initiator DB version on which the DPR was raised	
7.	ECSS proposed modification		To be filled by DPR initiator	
8.	ECSS DPR Status	Enumerated:	Open: initial status just after the creation of the DPR, Rejected: in this case the attribute “ECSS DPR Justification” shall be filled in, To be implemented: the attribute “ Implemented ”	ECSS DPR Status
9.	ECSS DPR Justification	Text	Reason for rejecting the DPR	ECSS DPR Justification
10.	ECSS DPR agreed modification	Text	Description of the (to be) implemented change: there may be an initial implementation proposal which is superseded by the actual implementation description	
11.	ECSS updated DB version	Text	DB version number where the DPR is implemented	
12.	ECSS changed standards	Text	Lists the modified ECSS modules. Note: in the DB, any modified object will refer to the DPR, so no need to provide the detailed list of impacted objects.	ECSS changed standards

The “ECSS DPR list” module will contain the following view:

View: 01- DPR List

This view will display all the DPRs (no filter) and will contain all the attributes listed in the above table.

4.3.7 Summary of the attributes’ definition

4.3.7.1 Attributes at object level

Table 4-3 lists the attributes which will be present in the ECSS DOORS Database. Built-in Attributes are identified in purple, user-defined attributes already existing in the ECSS DOORS database are identified in green, new attributes to be added as defined in the above sections are identified in orange. For each attribute, some example values are provided for different object types.

Note: Naming of the attributes in the existing databases (ES and Industry/CNES) was respected.

Table 4-3: Attributes at object level

Attribute name	Attribute type	Attribute values	Comments
Object Text	Text	Free text; the main content of each object, e.g. the requirements text	
ECSS Object Type	Enumeration	Acronym, Bibliography, Definition, Figure, Heading, Table, Requirement, ...	Pre-defined list of possible values See 9.3.5
ECSS Section ID	String	e.g. 4.5.1	See 9.3.5
ECSS Source ID	String	e.g. 4.5.1 a	See 9.3.5
ECSS Source Reference	String	e.g. ECSS-E-ST-10-02C	See 9.3.5
ECSS Notes	Text	Free text	In some old DOORS modules the text of the note(s) or Expected output attached to the requirement.
IE PUID	String	<Document prefix>_<Document rank><normative Running number>	Only for requirements and tables/figures called by requirements or else it remains empty. See 9.3.1
OUID	String	<Document prefix>_<Document rank><non-normative Running number>#	For normative objects the OUID is a copy of the IE PUID. See 9.3.1
RCM Version	Enumeration	1, 2, 3, ...	Only for requirements and tables/figures called by requirements (Default value is 1) or else it remains empty. See 9.3.2.2.
ECSS Change Status	Enumeration	<i>Created, Normative Change, Informative Change, Deleted, or Unchanged</i>	Pre-defined list of possible values. See 9.3.2.3
Reason for update	Enumeration	<i>DPR, CR</i>	Both can be selected at a time. See 9.3.2.6.
Correction number	Text	<i>Free text</i>	This attribute must contain the list of CRs and/or DPRs that have originated the change. See 9.3.2.7.
Paragraph Style	Enumeration	<Object Text:paragraph>, <Object Heading: Heading 1>, or ...	Pre-defined list of possible values. See 9.3.4 where the different values of the enumeration are defined
ECSS Referenced ID		<i>List of DOORS "Absolute Numbers" ...</i>	Only for requirements referencing other requirements, tables or figures, else it remains empty. See 9.3.3.
OLE Counter	Integer	Number of OLE objects in the Object Text	The field is dynamically filled by DXL code.

4.3.7.2 Module Level

At module level, the following two attributes will be created”:

- **IE Requirement Number**, i.e. the highest requirement or Table/Figure (called by a requirement) number generated so far in the module, to be used for automatic generation of <normative running numbers> for later updates of the standard (e.g. requirements added).
- **OID Number**, i.e. the highest number generated so far in the module, to be used for automatic generation of <non-normative running numbers> for later updates of the standard.
- **Prefix**, which is the concatenation of “**ECSS document prefix**” and “**ECSS document rank**” as identified in the module “**ECSS standard list**” (see section 9.4). This is used to generate the IE PUID and OID as described in section 9.3.1

=> Either to create a macro for initialising this value at creation of the module or update of the macro to number requirements (TBC)

4.4 Definition of module “ECSS standard list”

Module name: ECSS standard list

Attributes:

- ECSS standards title
- ECSS document prefix
- ECSS document rank
- Latest ECSS version published
- First issue imported in DOORS
- ECSS latest correction ID
- Note for ECSS Std list
- Date of information

See table ANNEX A , where values of ECSS document prefix and ECSS document rank are defined.

This module will have to be made available like the ECSS document tree as a DOORS module archive.

4.5 DOORS modules and database organisation

The DOORS database contains all modules corresponding to the first version of the standard imported in DOORS. Revisions of ECSS Standards result in the creation of a new DOORS Module The superseded Module is moved into the folder "ECSS Superseded". As a consequence of building a module for each published version of a standard, it was agreed to update the DOORS database structure to separate physically those modules for the current version of published standards from those at earlier version. It is proposed to adopt the same approach as for the ECSS CD-ROM containing all ECSS standards.

The ECSS DOORS Database is as follows:

DOORS Database

ECSS – Current

E – Space Engineering

M – Space Project Management

Q – Space Product Assurance

S – ECSS System

ECSS – Superseded

E – Space Engineering

M – Space Project Management

Q – Space Product Assurance

S – ECSS System

Module: ECSS DPR list

Module: ECSS standard list

Where: **xxxxx** = Structure having the type “project” and **xxxxxx** = Structure having the type “folder”

Hereafter an example to summarize the process for updating DOORS reflecting the issue of a new version of a standard.

Let’s consider the process for updating the standard E-ST-XXX C Rev. 1 to E-ST-XXX C Rev. 2 that is ready for being published and for which the changes have to be reflected in DOORS.

At start the:

- Module **E-ST-XXX C Rev. 1** is in the folder “ECSS – Current”
- Module **E-ST-XXX C** is in the folder “ECSS – Superseded”

At the end the:

- Module **E-ST-XXX C Rev. 2** is in the folder “ECSS – Current”
- Modules **E-ST-XXX C** and **E-ST-XXX C Rev. 1** are in the folder “ECSS – Superseded”

Process description:

- 1) Copy current Module version of **E-ST-XXX C Rev. 1** (without copying the baselines) and move the original module into the folder “ECSS – Superseded”
- 2) Rename the copied module of **E-ST-XXX C Rev. 1** in “ECSS – Current” to **E-ST-XXX C Rev. 2**
- 3) Apply modifications to the current version of **E-ST-XXX C Rev. 2** as necessary to be consistent to changes identified in the Word file

4.6 DOORS Database Problem Reports (DPR) management

Errors in the DOORS database modules shall be corrected on the basis of Database Problem Reports (DPR). All DPRs shall be reviewed and dispositioned by the ECSS TA and implemented by ES.

All DPRs shall listed in the dedicated Module "ECSS DPR List" (see also 9.3.6).

As long as there is no DPR implemented the name of the Module shall be the same as the published standard. The moment a DPR is implemented in a DOORS module the name of that DOORS module shall be changed to identify the change (addition of "corr1" for the first correction from a DOORS DPR, and be incremented, e.g. "corr2", "corr3" for subsequent corrections of the same module). When a new Revision of that module is created the name of the module will the same as the published standard again.

Example: To correct a mistake in a DOORS module of ECSS-E-ST-20-07C Rev. 1 the name of the DOORS module shall be: ECSS-E-ST-20-07C Rev. 1 Corr. 1.

At start the:

- Module **E-ST-20-07C Rev. 1** is in the folder "ECSS – Current"
- Module **E-ST-20-07C** is in the folder "ECSS – Superseded"

At the end the:

- Module **E-ST-20-07C Rev. 1 Corr. 1** is in the folder "ECSS – Current"
- Modules **E-ST-20-07C** and **E-ST-20-07C Rev. 1** are in the folder "ECSS – Superseded"

Should another DPR be found for the same module E-ST-20-07C Rev. 1 corr-2 will be created in the folder "ECSS – Current" and the Module **E-ST-20-07C Rev. 1 Corr. 1** will be moved to the folder "ECSS – Superseded".

If E-ST-20-07C is updated to a Revision 2, while a version with "corr-x" exists the new Module will be called **E-ST-20-07C Rev. 2**.

When corrections to a DOORS Module are made, it is proposed to:

- indicate the reason for the change in the object attribute "Reason for update",
- indicate the reference of the CR (if known) and/or of the DPR which are at the origin of the correction in the object attribute "Correction number",
- deliver a "Release Note" listing the corrections in text form in addition to the module "ECSS DPR list".

4.7 Definition of DOORS Views

To display ECSS DOORS modules content e.g. allowing filtering, sorting and grouping information, several views will be developed as follows:

01 – Total View

The purpose of this view is to display all objects' attributes in the modules and is defined as follows:

Attribute name	ECSS Source Reference	ECSS section	ECSS Source Id	IE PUID	OID	RCM Version	Object Text	ECSS Notes	ECSS Object Type	ECSS OLE Counter	ECSS Change Status
----------------	-----------------------	--------------	----------------	---------	-----	-------------	-------------	------------	------------------	------------------	--------------------

continued

Reason for Update	Correction Number	ECSS Referenced ID	Cross-references	Paragraph Style
-------------------	-------------------	--------------------	------------------	-----------------

02 – Overview

The purpose of this view is to display major objects' attributes in the modules and is defined as follows:

Attribute name	ECSS Section	ECSS Source Id	IE PUID	OID	RCM Version	Object Text	ECSS Notes	ECSS Object Type	Cross-references
----------------	--------------	----------------	---------	-----	-------------	-------------	------------	------------------	------------------

03 – Identification of Changes

The purpose of this view is to display the differences existing between the current module version and its baseline 1.0 to identify the changes w.r.t. the previously published version of the standard. It is defined as follows:

Attribute name	ECSS Section	ECSS Source Id	ECSS Object Type	IE PUID	OID	RCM Version	Object Text	ECSS Notes	ECSS Change Status
----------------	--------------	----------------	------------------	---------	-----	-------------	-------------	------------	--------------------

Continued:

Reason for Update	Correction number
-------------------	-------------------

In this view, a filter will be defined to identify all objects which were modified.

This view has been deleted.

04 – Requirements

The purpose of this view is to display only the requirements and the tables and figures called by requirements (all objects having a non-empty value for IE PUID). It is defined as follows:

Attribute name	ECSS Source Id	IE PUID	RCM Version	Object Text	ECSS Notes
----------------	----------------	---------	-------------	-------------	------------

In this view, a filter will be defined to identify all requirements and table/figures called by requirements.

05 – Cross-references

The purpose of this view is to display only the Cross-references within a module. It is defined as follows:

Attribute name	ECSS Source Id	IE PUID	Object Text	ECSS Notes	Cross-references
----------------	----------------	---------	-------------	------------	------------------

4.8 Database consistency rules

This section describes the consistency rules that bind the different attributes in the database. In the database maintenance process these rules shall be fulfilled to ensure a consistent database.

Table 4-4 list the attributes which are concerned by a consistency rule.

Table 4-4: Attributes for Database consistency rules

Attribute name	Consistency rule	Connected attributes
RCM version	This attribute must be filled in with an integer value for any requirement ("ECSS Object Type" = 'Requirement'). See 9.3.2.2.	ECSS Object Type
RCM version	The RCM version evolution shall be consistent with the modification of the attribute "ECSS Change Status". If "ECSS Change Status" is 'Created', "RCM Version" shall be equal to 1. If "ECSS Change Status" is different from 'Created', "RCM Version" shall be > 1. The corresponding consistency rules are described in section 9.3.2.2 and in ANNEX D .	ECSS Change Status
ECSS Object Type	This attribute is set to 'Requirement' if the attribute "Object text" contains a 'shall' or "shall not". This attribute is set to 'Recommendation' if the attribute "Object text" contains a 'should' or 'should not'. This attribute is set to 'Permission' if the attribute "Object text" contains a 'may' or 'need not'. If the text of an Object Type 'Requirement', 'Recommendation' or 'Permission' does not contain the respective verb of a provision, it is considered as badly written and a CR needs to be raised to correct this error. Deleted 'Requirements' shall be marked with the statement <<deleted>> to avoid any renumbering. The 'IE_PUID" and the 'Object Type' stays unchanged.	Object text

Attribute name	Consistency rule	Connected attributes
IE PUID	This field must be filled in if the attribute "ECSS Object Type" is set to 'Requirement', 'Recommendation' or 'Permission' (see 4.3.1.2). This field must be filled in for tables or figures called by at least one normative object (see 9.3.1.3).	ECSS Object Type
IE Requirement Number	This attribute must contain the highest requirement or Table/Figure (called by a requirement) number generated so far in the module (see 9.3.7.2).	
IE PUID	This attribute must have the structure defined in section 4.3.1.2. Specially, the running number must contain 4 digits.	
OUID	If the attribute IE PUID is filled ("ECSS Object Type" is set to Requirement, Permission or Recommendation) this field is equal to the IE PUID.	ECSS Object Type, IE PUID
OUID Number	This attribute must contain the highest non-normative object number generated so far in the module (see 9.3.7.2).	
ECSS change status	If the value of this attribute is 'Created', the value of the attribute 'RCM version' must be set to 1. The corresponding consistency rules are described in section 9.3.2.2 and in ANNEX D .	RCM version
Correction number	This attribute must be filled in when the attribute "Reason for update" is not empty (see section 9.3.2.6).	Reason for update
Cross-references	If a Table (or a Figure) is called by a requirement, it must have a cross-reference. See 9.3.2.7.	

4.9 Outputs

The following outputs will be generated and delivered to ECSS members via the ECSS website or upon request:

- Projects archive (.dpa) for current and superseded standards, consisting of:
 - All modules for ECSS standard, including baselines
 - ECSS standard list
- Release Note providing information concerning the structure, content and changes done to the ECSS DOORS database.

4.10 Processes

4.10.1 Proposition for update and verification - Mock-up phase

The proposition for upgrading the ECSS DOORS database is based on this report and the realisation of a mock-up for validating the solution proposed as part of the report. This approach has allowed reducing the risks and later decrease implementation duration and cost.

The mock-up and tests were finalized end 2011 and results were fully reviewed by the TF in January 2012 for updating/consolidating this report:

- Achievements:
 - Verification of the main issues for importing requirement numbers, creating internal links using scripts, creating new attributes and views.
 - Specification of a template for DOORS module for the definition of the attributes and views to be imported/copied
 - Confirmation that no major discrepancies between ES Database and Industry/CNES Database exist
 - Scripts update

Finalization of the Mock-up phase is in line with the report submission to TA decision.

4.10.2 Database update and validation

Updates of the ECSS DOORS database shall be done by the ECSS DOORS Expert Group members before recommendation to the ECSS TA for release of the new version of the database.

To support the revision of the Modules a control script named "Coherence.dxl" was created and is maintained by the Expert Group.

4.10.3 Database release

After validation by the TA, the ECSS DOORS database and accompanying document will be issued by ECSS ES in the form of a DOORS archive containing the published ECSS standards "Current" and "Superseded" (.dpa) and a DOORS Release NOTE.

This information will be uploaded and made available for download from the ECSS Website, which contains the standards in pdf and word format.

4.10.4 Database maintenance

After ECSS DOORS database delivery the following activities will be performed:

- a. When a new ECSS standard, a new issue or revision of an ECSS Standard is ready to be published, the DOORS database will be updated by ES (i.e. the module created and verified by ES). Before its formal publication the update will be sent to the TA for approval for publication.
- b. When discrepancies on published DOORS modules are identified (e.g. from ES or users), Database Problem Reports (DPR) related to DOORS database shall be issued and sent to ES, to be dispositioned by the TA. The corresponding corrected version of the module will then be generated and validated and published following the above approach (see 9.10.4a).

4.11 Planning & resources

As stated in the process section, the consolidated proposed solution based on activities performed as part of the mock-up phase will be finalized in the report delivered for TA approval. As part of the approval process each actor will have to define in advance the impacts/constraints for integrating the ECSS DOORS database in place of the existing solution.

Upon approval of the proposed solution and resources allocated (e.g. for the validation phase) the activities will start (T0).

4.11.1 Opportunities

Activities performed as part of the DOORS TF is the first opportunity at ECSS level to address issues related to management issues of standards' requirement using a DOORS database.

Currently the TF is mainly involved in step #1 to define the ES Database update and provide support to ES during the database update.

Performing Step #2 is an opportunity to continue team wise issues related to use of the database for standard management, e.g. their maintenance including for example update of document and management of CRs.

4.11.2 Risks already mitigated

Table 4-5 identifies those risk addressed and mitigated as part of the TF activities.

Table 4-5: List of already mitigated risks

Risk	Discussion & Mitigation action
Different approaches for requirement capture & database content	<p>Since both approaches are already in use in both sides it was necessary to maintain compatibility that the ECSS database respects attributes naming/content when already defined.</p> <p>Selection of ES database adding existing attributes name/content for requirement numbering from Industry/CNES Database</p>
No impact to ECSS process and for users working with ECSS WORD version	<p>The new requirement ID generated in DOORS will exist in parallel to the document in word format.</p> <p>A correspondence table will be built and distributed for those users not using DOORS but willing to be aware of the existing correspondence between the IE PUID and the current numbering as in the word version.</p>
Try to resolve all writing problems when importing standards in DOORS database	<p>Some problems (e.g. the identification of notes in requirements, addition of requirements when revising standards) are related to drafting rules for document writing/maintenance and are not specific to DOORS import.</p> <p>It is more pragmatic to address these issues in relevant document/standard. Therefore, some updates of the drafting rules are necessary and some CRs will be generated by Eurospace.</p>

5. Considerations for potential evolutions of the ECSS DOORS database

The report provides a definition of the ECSS DOORS database and solution for building it as an update of the existing ESA ECSS DOORS database for ECSS C version.

In addition, during TF meetings, some potential evolutions were identified but not deeply addressed since not part of the TF mandate. Some potential evolutions concern for example the:

- use of the ECSS DOORS Database as the unique source for publishing and maintaining the different formats of ECSS Standards, i.e. WORD, PDF, and DOORS,
- implementation of the new requirement identifier number (IE PUID) instead or together with the requirement identifier currently used in ECSS standards e.g. for issue D of standards,
- addition of dedicated attributes and their content in line with pre-tailoring activities, when available.

6. Conclusion

The Tasks of the TF as defined in the TOR were completed and the following were provided for TA decision:

- Section 9 that provides a detailed description of the proposed solution prior starting implementation
- Maintenance of the DOORS TF during implementation phase and commitment of resources
- Extension of TOR of the TF for investigating upon activities identified in section 10 for better integration and maintenance of both the ECSS DOORS Database and the Standards in word format
- Mechanism for approving the DOORS modules

ANNEX A ECSS standards – reference/version and “ECSS rank”

The DOORS database contains a module titled "ECSS standard list" containing the following information:

- ID
- ECSS standard title
- ECSS document prefix
- ECSS document rank
- Latest ECSS version published
- First issue imported in DOORS
- Latest ECSS version in DOORS
- ECSS latest correction ID
- Present in Current Folder
- Note for ECSS Std list
- Date of information

An example of the "ECSS standard list" module from the DOORS database is shown in Figure A-2.

Figure A-1: Example of “ECSS Standards list”

ID	ECSS standard title	ECSS document prefix	ECSS document rank	Latest ECSS version published	First issue imported in DOORS	Latest ECSS version in DOORS	ECSS latest correction ID	Present in Current Folder	Note for ECSS Std list	Date of information
279	Software	ECSS-E-ST-40	86	C	C	C		Yes		2-Jun-14
281	Communications	ECSS-E-ST-50	97	C	C	C		Yes		2-Jun-14
282	Space data links - Telemetry synchronization and channel coding	ECSS-E-ST-50-01	24	C	C	C		Yes		2-Jun-14
283	Ranging and Doppler tracking	ECSS-E-ST-50-02	87	C	C	C		Yes		2-Jun-14
284	Space data links - Telemetry transfer frame protocol	ECSS-E-ST-50-03	88	C	C	C		Yes		2-Jun-14
285	Space data links - Telecommand protocols, synchronization and channel coding	ECSS-E-ST-50-04	89	C	C	C		Yes		2-Jun-14
286	Radio frequency and modulation	ECSS-E-ST-50-05	90	C Rev.2	C Rev.1	C Rev.2		Yes	C Rev.1 was first version in DOORS	2-Jun-14
386	SpaceFibre - Very high-speed serial link	ECSS-E-ST-50-11	141	C	C	C		Yes		15-May-19
287	SpaceWire - Links, nodes, routers and networks	ECSS-E-ST-50-12	91	C	C	C		No	Superseded by complete new module with no traceability between Issue C (rank number 91) and Revision 1 (rank number 142).	15-May-19
387	SpaceWire - Links, nodes, routers and networks	ECSS-E-ST-50-12	142	C Rev.1	C Rev.1	C Rev.1		Yes	No traceability between Issue C (rank number 91) and Revision 1 (rank number 142).	15-May-19
288	Interface and communication protocol for MIL-STD-1553B data bus onboard spacecraft	ECSS-E-ST-50-13	101	C	C	C		Yes		2-Jun-14
289	Spacecraft discrete interfaces	ECSS-E-ST-50-14	94	C	C	C		Yes		2-Jun-14
372	CANbus extension protocol	ECSS-E-ST-50-15	127	C	C	C		Yes		25-May-15
291	SpaceWire protocol identification	ECSS-E-ST-50-51	107	C	C	C		Yes		2-Jun-14
292	SpaceWire - Remote memory access protocol	ECSS-E-ST-50-52	108	C	C	C		Yes		2-Jun-14
293	SpaceWire - CCSDS packet transfer protocol	ECSS-E-ST-50-53	109	C	C	C		Yes		2-Jun-14
294	Control performances	ECSS-E-ST-60-10	102	C	C	C		Yes		2-Jun-14
295	Star sensor terminology and performance specification	ECSS-E-ST-60-20	92	C Rev.2	C Rev.1	C Rev.2		Yes	C Rev.1 was first version in DOORS	15-May-19
380	Gyro terminology and performance specification	ECSS-E-ST-60-21	135	C	C	C		Yes		15-Feb-17

ANNEX B Trade-off for requirement numbering

NOTE Every time the term “requirement” is used in this Annex it is used in the sense of the new term “normative provisions” that includes also “Recommendations” and “Permissions”.

B.1 Overview

Currently, in the Word / PDF versions of the ECSS standards, the requirements are identified and numbered following the respective section numbers. These identifiers are neither invariant to internal standard document re-structuring, nor do they provide a unique identifier over the whole set of standards (i.e. the identifier is only unique within the standard itself). Therefore, it was agreed to add an identifier attribute unique in the ECSS system and invariant to its location in a standard.

A requirement numbering approach based on a compound requirement identifier with a prefix (e.g. doc name or acronym) and a counter is a common requirement management best practice.

Different approaches are possible for implementing a requirement numbering and a counter.

This annex is a summary of the assessment of the different solution envisaged by the task force:

- review of trade-off is already done (e.g. ESA, Industry/CNES...)
- analyse constrains and impacts on current implementations (e.g.: ESA, industry/CNES)

B.2 Trade-offs

B.2.1 Prefix information

Decision: <Document prefix>_<Document rank>

Issues: Based on best practices:

- be able to move requirements from one to other ECSS: the prefix shall be not dependent name a document, a number shall be allocated to each ECSS (**Document rank**: allocation of number independent to standard the structure as initiated by Industry/CNES and to be completed/maintained by ES – see ANNEX A).
- when a requirement is called on non ECSS documents or discussion is more comfortable to have a document prefix

B.2.2 Requirement counter

Decision: requirements sequential counter for each module (i.e. by standard) with numbering of table and figures called by a requirement. No space for adding additional requirements.

Analysis of possible solutions:

a) *Absolute number (using database identifier for all standards)*

Issues: Moving a requirement from one ECSS document to another ECSS document, delete the requirement from one module and create a new requirement in another module, with a new absolute number.

In this case, the requirement identifier changes while the requirement text is not changed.

Decision: Not compliant with requirements

b) Sequential counter for each module (i.e. per document) or global for the whole ECSS standard database

Issues:

- The module counter is simple to implement: it does not require any synchronisation mechanism when creating a new requirement. It provides compatibility with industry solutions.
- The global counter alternative requires to implement a global counter recording the last allocated number, to implement a global service and to manage locks to ensure unicity of the requirement number. Therefore, the service may not be free when new requirements are created. Moreover, all the modules shall be hosted in the same database when creating the requirements, to be able to use the global counter service, assigned to a database.

Decision: Module counter

c) Sequential number with space for potential insertion of requirement

Issues:

- The sequential numbering is very simple: there is no need for recording the number of the previous requirement and of the next requirement, a requirement can be moved without managing its relative positions w.r.t. its neighbouring requirements
- The sequential numbering is compatible with existing numbering systems by industry
- The requirement numbering system allows to uniquely identify a requirement, not to locate it. Locating a requirement inside a document is made through DOORS navigation features (filtering, sorting, finding) and through the ECSS standard number.
- if needed, in the matrix document for example, the requirements numbers may be complemented by the ECSS standard section numbers which contain the requirement.
- No use of a requirement number increment >1 , from one requirement to the next created, since the requirement numbering system will not allow any requirement number insertion between two existing requirements.

Decision: The numbering system allows a sequential numbering of the requirements with a requirement number increment equal to 1.

d) Identical or different counter for table and figures

Tables and figures linked part or requirement in ECSS database are in separate objects, merge objects consider complex and source for other potential problems (e.g. when exporting back to Word). The solution for helping user in identifying those table and figures provide a dedicated s t number to those tables of figures

Issues:

- Add object identification on the prefix (e.g. REQ, FIG ...) would have high impact on existing Industry/CNES documents that were already published (e.g. for reusing/capitalising results from activities performed on ongoing projects). This impact in term of modification and revalidation led to reject this solution.
- A separate counter is more complex for use without clear added value

Decision: use requirement counter for Table and Figure

B.2.3 Recommendations and permissions numbering

Recommendations and Permissions are treated the same way as Requirements. Therefore, in most parts of this document the general term “Normative provision” is used.

B.3 Summary of the main evolution w.r.t. Industry/CNES requirement numbering

- Added requirement numbering ID to figures and tables that are called by a requirement (ID build with the same pattern as for a requirement)
- Running counter of 4 digits instead of 3 digits (i.e. adding before the existing number).

ANNEX C Trade-off for “notes” and “expected output”

In published standard from WORD:

- the notes and expected output are identified with a specific document style
- the notes are inside requirement text (requirements with bullets) or at the end of requirement text
- expected outputs are at the end of the requirements

A clarification is strongly requested by ECSS user's in order to be focused on requirement only.

The problem with notes / expected output is that sometimes notes call documents and erroneously these documents are sometimes made applicable (i.e. misuse between documents identified in Normative Reference and Bibliography). This could be resolved through a recommendation on the process how to make documents applicable (by only making documents explicitly applicable, and not accepting implicit applicable documents through references in requirements / notes)

This problem is identified and understood among the TF members.

This problem is not implicit to DOORS but applies to Word and DOORS in the same way. It is however highlighted through DOORS. But any modification in the DOORS approach would not solve the issue, it must be resolved on a higher level, not through the implementation of the DOORS DB.

Currently, in the current ESA DOORS DB, notes are part of the requirement object's "Object Text" attribute. Notes are identified by indentation and the starting word "NOTE".

The trade-off for ESA DOORS database is between use "as is", hybrid solution,

A generic solution where the Notes are captured as DOORS attributes cannot be implemented at the current state based on the existing Word files / ECSS DOORS DB (mainly due to cases where NOTES are interleaved with requirements text.)

The following hybrid solution was agreed:

- For ECSS where some NOTES are interleaved with requirements text No change to the current DOORS DB (i.e. Notes are part of the requirements' "Object Text"). Eurospace will issue a CR for ECSS modification following rule:
 - 1 Note in one requirement → one entry in "Notes" attribute
 - n Notes in one requirement → one entry with n paragraphs in "Notes" attribute
 - (1, n) Note(s) interleaved with requirement text → one entry with (1, n) paragraphs in "Notes" attribute, but with the Notes reworded to clarify (because they will be taken out of the specific context)
- For ECSS (~30%) where Notes (and expected outputs) are not interleaved with requirements text the notes and expected outputs will be imported as requirement attribute during ESA database update.

Hybrid solution possible as the attribute solution can be applied to individual modules one at a time (i.e. not necessarily to all standards in the DB at the same time); but no hybrid solution within one module.

Next Step: all Notes (and expected outputs) as attributes to requirement objects

When updating a standard and in resolution of raised CR, the Word document will be updated such that all NOTES can be transferred in the dedicated attribute, which will then be done in updating the dedicated version of the module of the ECSS database.

ANNEX D Details of the mechanism for change tracking

The following tables summarizes the various potential changes to object of the DOORS module and their impact on:

- RCM version (including requirements and tables, figures called by a requirement)
- ECSS Change Status value.

Abbreviations used in this table are:

- ID is either the IE PUID or the OUID where: **A_α_** is the Prefix **ECSS-Q-ST-20_0038** and **x,y** the number of the normative provision or non-normative object (*NOTE: OUIDs have an extra “#” after the number*)
- Version “n” is the current version, n++ are following issues

Nature of modification	ID (IE PUID or OUID)	RCM Version	ECSS Change Status	Comment
New normative provision, Figure or Table called by a normative provision in new document	New PUID	1	Created	Figures and Tables called by normative provisions are treated like a normative provision
New normative provision, Figure or Table called by a normative provision in existing standard	New PUID	1	Created	This applies for new issues of Standards.
Update of existing normative provision, Figure or Table called by a normative provision	Existing PUID	n++	Normative Change	
Update of note (attribute) in existing normative provision, Figure or Table called by a normative provision w/o modification of requirement text	Existing PUID	n	Informative change	
Update of note (attribute) in existing requirement with modification of requirement text	Existing PUID	n++	Normative change	
Update of cross-references only w/o changing the remaining text of the normative provision	Existing PUID	n++	Normative change	For example: Renumbering of Figure or Table due to insertion of new ones.
Modification of definitions	Existing OUID	n/a	Informative Change	
Deletion of existing normative provision or Figure or Table called by normative provision	Existing PUID	n++	Deleted	Text changed to <<deleted>>

Nature of modification	ID (IE PUID or OUID)	RCM Version	ECSS Change Status	Comment
Demotion of a requirement into permission or recommendation	Existing PUID	n++	Normative Change	This rule was changed in 2019 when it was decided that not only Requirements, but also Recommendations and Permissions will have complete change tracking with "RCM Version" and "ECSS Change Status". Therefore, the term "requirement" was replaced in the whole document by "normative provision" and a dedicated definition added at the beginning of the document.
Splitting of existing normative provision or Figure or Table called by normative provision	Existing PUID	Existing: n++	Normative Change	Both requirements shall be analysed in terms of attributes and links
	New PUID	New: 1	Created	No further traceability required on the nature of the change
Merging of existing normative provision, Figure or Table called by normative provision	Existing PUID	Remaining norm. provision: n++	Normative Change	Text changed to <<deleted and merged into requirement xx>>
	Existing PUID	Other norm. provision: n++	Deleted	
Deleted requirement that is reintroduced in the next revision of a Standard.	Existing PUID	n++	Normative Change	PUID number stays the same. The ECSS Change Status changed from "Deleted" to "Created" and the RCM Version incremented.

Nature of modification	ID (IE PUID or OUID)	RCM Version	ECSS Change Status	Comment
Normative provision(s) calling Figure/Table				
Update of normative provision calling a Figure or Table	Existing PUID	Norm. provision: n++	Normative Change	Only the RCM version and ECSS Change Status for the modified normative provision is changed
	Existing PUID	Table/Figure: n	Unchanged	
Update of normative Figure/Table	Existing PUID	Norm. provision: n	Unchanged	Only the RCM version and ECSS Change Status for the modified normative Figure/Table is changed
	Existing PUID	Norm. Table/Figure: n++	Changed	
Update of normative provision and referenced Figure/Table	Logical combination of the above			

Nature of modification	ID (IE PUID or OUID)	RCM Version	ECSS Change Status	Comment
New standard issue (Revision)				
New standard issue, but no change in normative provision	Existing PUID	n	Unchanged	For all objects which are unchanged
New standard issue; deleted normative provision (deleted in previous standard issue)	Existing PUID	n	Unchanged	<p>The as <<deleted>> marked normative provision stays in the module with its PUID number. Only the ECSS Change Status is updated to "Unchanged."</p> <p>Object removed from DOORS module for new issue</p> <p>Note 1: At this time this has never been implemented.</p> <p>When the case will come the following will be done in the DOORS module: 1. The requirement will be deleted from DOORS (and purged). 2. If all requirements of a clause are deleted (e.g. the clause became empty), then the clause header will be replaced by "<<deleted>>" in order to keep the overall document hierarchy in DOORS.</p> <p>The Word file will still contain all deleted requirements marked with "<<deleted>>".</p> <p>Note 2: There is no risk reusing the PUID since the latest value is kept per module independent of any deletion.</p>

Nature of modification	ID	RCM Version	ECSS Change Status	Comment
Move inside document				
Move of normative provision inside document w/o modification of the requirement objective (*)	PUID removed and replace by new OUID A- α _x	Old: no version n	Informative change Unchanged	Text changed to <<moved to requirement xx>> Note: Placeholder to keep the document structure. Requirement moved to the new location together with its IE PUID number but without change of RCM Version of Change Status. Previous location of norm. provision The move shall be listed in the change log
Move of normative provision inside document with modification of the requirement objective (*) (e.g. contextual link to section)	A- α _x	Old: n++	Deleted	Factual delete and create Old text: <<deleted>>
	A- α _y	New: 1	Created	New text
Move of informative object inside a document				Text of informative object together with its OUID move to new place. Change Status stays unchanged in the object. Note: Should the move affect the structure of normative clauses than the Heading shall be kept and marked as <<deleted>>.
Move outside document				
Move of normative provision outside w/o modification of the requirement objective(*)	A- α _x	Old: n++	Deleted	Old text: <<deleted and moved to B- α _y>>
	B- α _y	New: 1	Created	
Move outside with modification of the requirement objective (*) (e.g. contextual link to section)	A- α _x	Old: n++	Deleted	Factual delete and create Old text: <<deleted>>
	B- β _y	New: 1	Created	New requirement text

“modification of the requirement objective” can be a modification of the requirement text or a change of the context when the requirement is moved into a different section.

ANNEX E Library of scripts

The following DOORS scripts were created for the maintenance of the ECSS DOORS database:.

DOORSrb script

Software package to generate:

- new DOORS modules using as input the MS Word files of ECSS Standards
- updates of an existing DOORS module using as input the revision tracked MS Word file of the ECSS Standard and the DOORS module or the previous revision.

exportIEPUIDs script

Software package to export the IE PUID and OUID numbers from a DOORS module as Bookmarks into the MS Word version of the respective ECSS Standard.

NOTE: The imported IE PUID numbers in the MS Word file are visible and OUID bookmarks only shown at the beginning of the non-normative object by a bookmark sign.

In addition several DOORS scripts were developed to export data from the ECSS DOORS database. For example export of all Terms and Definitions or Abbreviated terms.

Coherency script

DOORS script to track incoherencies in the DOORS modules and between current and previous DOORS modules (e.g. module of ECSS-E-ST-10C Rev.1 with module of ECSS-E-ST-10C).