# Impact Analysis Report / RFC-Proposal

**Section 1: Meta-data**

|  |  |
| --- | --- |
| **RFC ID** | **RFC\_DDCOM\_0027** (RTC-55494) |
| **Related Incident ID** | - |
| **RFC Initiator / Organization** | DG TAXUD/B3 |
| **CI** | DDCOM-20.3.0-v1.00 |
| **Type of Change** | **Standard** **Emergency** |
| **Nature of Change** | Justification for Evolutive   |  | | --- | |  | |
| **RFC Source** | |  |  | | --- | --- | | **Legal & Policy Change**  **Organisational Changes** | **Business Change**  **IT Change** | |
| **Review by Business User recommended?** | **Yes No** |

***Change Summary***

|  |
| --- |
| **DDCOM-20.3.0-v1.00: Correction for XML messages: numerical fields (token) and text fields (value zero)** |
| The DDCOM needs to be enhanced, in order to clarify the following:   * The usage of the xsd element **<xs:token>** that is used as a base class for all alpha numeric types, according to table 43; * For the use of numerical fields, the permitted values are not only impacted by business rules and codelists but **also by xsd patterns** (for XML messages).   See also:   * RFC\_NCTS\_0179\_CUSTDEV3-IAR-RTC60124-v1.00(SfA-NPM+IMP).docx and * RFC\_AES\_0132\_CUSTDEV3-IAR-RTC60125-v1.00(SfA-NPM).docx. |

**Section 2: Problem statement**

|  |
| --- |
| Α. Analysis on alpha numeric fields and the use of <xs:token> element  The current version of DDCOM (section “**V.2.1.1.2** Text fields”) mentions that :  **V.2.1.1.2 Text fields**  Leading and trailing spaces (both normal spaces and non-breaking spaces) shall not be used within text fields.  The EDIFACT separator characters (see EDIFACT message formatting) can be used within such a field. The EDIFACT release character (?) can be used to include the separator characters in fields.  For XML, certain characters cannot be used in its content because they have special meaning. Adding control characters (‘<’, ‘>’ etc) into XML data could cause the parser to misunderstand the resulting data. The solution (see [S4]) is to escape the control characters so that the parser can interpret them correctly as data, and not confuse them for markup. These characters have to be escaped with the following predefined entities. To use one of the characters listed below, substitute it with the appropriate string.    Table : characters to be escaped with predefined entities  (…)  This specification was and remain valid for EDIFACT messages.  It should be corrected / clarified for XML messages.  With the use of xsd element **<xs:token>**:   * any leading or trailing whitespaces are possible and skipped, * any whitespaces in the middle of strings are possible and considered as a single character,   during the xsd validation  Some examples are given below, based on the CD001C message for the data item CD001C/Consignment/DepartureTransportMeans/identificationNumber:  for which, the XSD includes the following part:  <xs:simpleType name="IdentificationNumberContentType02">  <xs:restriction base="xs:token">  <xs:pattern value=".{1,35}" />  </xs:restriction>  </xs:simpleType>  **Example 1:** 35 characters - It does **not** violate the pattern ".{1,35}" (token)    **Example 2:** 35 characters + trailing spaces - It does **not** violate the pattern ".{1,35}" (token)    The trailing spaces are ignored.  (idem for leading spaces)  **Example 3:** 35 characters with multiple spaces - It does **not** violate the pattern ".{1,35}" (token)    Each of the *triple spaces* in the middle of the string are treated as 1 character.  **Example 4:** 37 characters with multiple spaces - It violates the pattern ".{1,35}" (token)    Each of the *double spaces* in the middle of the string are treated as 1 character.  Β. Analysis on numerical fields and the applicable values  The Section “V.2.1.1.1 Numerical fields” of the DDCOM documents, describes the use of numerical fields, which can be either cardinal values (i.e. positive integers) or a decimals. The appropriate values for these fields are affected by the business rules and the code lists that are applied to each one. Extract from **DDCOM-20.3.0-v1.00**:  **V.2.1.1.1 Numerical fields**  Concerning numerical fields, it should be noted that these are either a cardinal value (positive integer value) or a decimal value12, unless otherwise specified by a Codelist or rule applied to the numerical field.  In order to enable the RFC-Proposals defined for DDNTA-5.14.1-v1.00 and DDNXA-5.14.1-v1.00, the DDCOM needs to be modified to mention that the value ‘O’ is controlled by XSD patterns for the XML messages.  (for NCTS-P5, see the bundle **RFC\_NCTS\_0123**\_RTC-58676+**RFC\_NCTS\_0122**\_51543; for AES, see the bundle **RFC\_AES\_0120**\_51542+**RFC\_AES\_0121**\_RTC-58674,)  The DDCOM document should depict that the value of these numerical fields is not only affected by business rules and codelists but also from xsd patterns. |

**Section 3: Description of proposed solution**

|  |
| --- |
| The following updates will be performed into the **DDCOM-20.3.0-v1.00** (~~deleted text strikethrough and red colour~~ and added text in yellow).  A. Update of section V.2.1.1.2 Text fields  “**V.2.1.1.2** Text fields”:  For EDIFACT:  Leading and trailing spaces (both normal spaces and non-breaking spaces) shall not be used within text fields.  The EDIFACT separator characters (see EDIFACT message formatting) can be used within such a field. The EDIFACT release character (?) can be used to include the separator characters in fields.  For XML:   * ~~The XSD element <xs:token> is used for the ICS-P1, NCTS-P5 and AES-P1.~~   ~~Consequently, spaces can be used either as leading, trailing or in the middle of a text field. This practically means that:~~  ~~• spaces at the beginning of a text field (leading), are skipped during the syntax validation;~~  ~~• spaces at the end of a text field (trailing), are skipped during the syntax validation;~~  ~~• spaces in the middle of a text field, are considered as a single character.~~  ~~The semantic validation (using the Rules/BRTs/TRTs implementation) shall also be aligned with this behaviour of <xs:token>.~~   * The XSD element <xs:token> is used for the text fields of **ICS-P1**.   This practically means that during the syntax validation:  • spaces at the beginning of a text field (leading) are skipped;  • spaces at the end of a text field (trailing) are skipped;  • spaces in the middle of a text field are considered as a single character.   * The XSD element <xs:normalizedString> is used for the text fields of **AES-P1 and NCTS-P5**.   This approach offers a predictable length for each data to be inserted into the database[[1]](#footnote-1), in comparison with <xs:token>. So, the NTA or NECA application may load first and check after the loading (based on the persisted data) and is not obliged to check the message on-the-fly before inserting in the database.  Specifically during the Transitional Period (and after it for the L³ movements ) and in order to enable a smooth transition:   1. the Common Domain XML messages (produced by upgrading ECS-P2 and NCTS-P4 EDIFACT messages) could include:    1. non-breaking spaces in the middle of a data item (to remain compatible with ECS-P2 and NCTS-P4);  and (if the legacy NECA or NTA are correctly aligned to DDCOM) there will be:    2. no spaces (and no non-breaking spaces) at the beginning of a text field (leading);    3. no spaces (and no non-breaking spaces) at the end of a text field (trailing). 2. the External Domain XML messages[[2]](#footnote-2) produced by traders aligned to NCTS-P5 or AES-P1:    1. shall not include spaces (not even non-breaking spaces) at the beginning of a text field (leading);    2. shall not include spaces (not even non-breaking spaces) at the end of a text field (trailing);    3. may include multiple consecutive spaces in the middle of a text field, that are considered as a multiple characters.   Therefore, the text fields of those External Domain NCTS-P5 and AES-P1 messages shall be validated using XSD element <xs:normalizedString> (as all other NCTS-P5 and AES-P1 IEs), in combination with the pattern: <xs:pattern value="\P{Z}(.\*\P{Z})?"/>.  **After the end of the TP and the end of L³ Period**, all the NCTS-P5 and AES-P1 messages shall be validated using XSD element <xs:normalizedString>, in combination with the pattern: <xs:pattern value="\P{Z}(.\*\P{Z})?"/>.  This progressive implementation of strict validation will ensure smooth transition from EDIFACT to XML.  This practically means that during the syntax validation of the NCTS-P5 and AES-P1 IEs:   * Spaces (both normal spaces and non-breaking spaces) in the middle of a text field are always counted as normal characters; * Spaces (both normal spaces and non-breaking spaces) at the beginning or at the end are not allowed (based on the pattern: <xs:pattern value="\P{Z}(.\*\P{Z})?"/> applied).   The semantic validation (using the Rules/BRTs/TRTs implementation) shall also be aligned with the above behavior.   * ~~For XML,~~ Certain characters cannot be used in its content because they have special meaning. Adding control characters (‘<’, ‘>’ etc) into XML data could cause the parser to misunderstand the resulting data. The solution (see [S4]) is to escape the control characters so that the parser can interpret them correctly as data, and not confuse them for markup. These characters have to be escaped with the following predefined entities. To use one of the characters listed below, substitute it with the appropriate string.   (…)  B. Update of section V.2.1.1.1 Numerical fields  **V.2.1.1.1 Numerical fields**  ~~Concerning numerical fields, it should be noted that these are either a cardinal value (positive integer value) or a decimal value~~~~12~~~~, unless otherwise specified by a Codelist or rule applied to the numerical field.~~  **For EDIFACT messages:**  A numerical field shall include (both):   * either a cardinal value * or a decimal value,   unless otherwise specified by a Codelist or rule applied to the numerical field.  Only the **strictly positive values** are valid and the numeric value zero (0) is not considered as positive integer (idem for 0.0 , not a positive decimal value). The only exceptions are:   * for the time numerical fields where the numeric value zero (0) may be used, * for the fields specified with a Codelist or a Rule applied.   **For XML messages:**  A numerical Data Item shall include:  - either a cardinal value,  - or a decimal value,  as specified by the XSD pattern included in the Appendix X ~~(highlighted by a guideline in Appendix Q2)~~, possibly complemented by a Codelist or a Rule applied to this numerical Data Item.  By default, only the strictly positive values are valid and the numeric value ‘0’ (zero) is not considered as positive integer (idem for ‘0.0’ or 0.000 or similar), not a positive decimal value).  If the value ‘0’ (zero) can be (exceptionally) included in a numerical Data Item, then these exceptions are highlighted by a guideline in Appendix Q2.  **For EDIFACT and XML messages:**  The decimal separator is the decimal point “.”. No other symbols are permitted as decimal separator.  (…)  With footnote 12 deleted:  ~~12~~ ~~The numeric value zero (0) is not considered as positive integer or as positive decimal value. The only exception is time numerical fields where the numeric value zero (0) may be used.~~  **Impacted CI artefacts**:   1. DDCOM 20.3.0-v1.00: **Yes**   **IMPACT ASSESSMENT:**  This RFC-Proposal *enables other changes* in DDNTA (NCTCS-P5) and DDNXA (AES) (e.g. replacing R0021 by XSD pattern).  This change has no direct and immediate impact on the NCTS, ECS/AES, ICS systems. It aims to be backward compatible.  **Proposed** date of applicability in Operations (T-Ops): as soon as published (**backward compatible**)  **Proposed** date of applicability in CT (T-CT): N/A  **Expected** date of approval by ECCG (T-CAB): ~~January~~ February 2022  **Impact on transition Legacy/To-Be:** None  **Consequence of not approving the RFC-Proposal**: The changes in DDNTA and DDNXA are not possible (for replacing R0021 by XSD patterns) |

**Impact on CI artefacts**

|  |  |  |
| --- | --- | --- |
| **DDCOM-20.3.0-v1.00** | Cosmetic  Low  Medium  High  Very High  Short description   |  | | --- | | Updates as described in section 3. | |
| **DDNTA-5.14.1-v1.00** | Cosmetic  Low  Medium  High  Very High  Short description   |  | | --- | | Will be documented in a separate RFC-Proposal for NCTS-P5 | |
| **DDNXA-5.14.1-v1.00** | Cosmetic  Low  Medium  High  Very High  Short description   |  | | --- | | Will be documented in a separate RFC-Proposal for AES-P1 | |

**Estimated impact on National Project**

|  |  |
| --- | --- |
| Cosmetic  Low  Medium  High  Very High  Short description   |  | | --- | | Impact will be assessed on DDNxA. | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Document History** | | |  |
| **Version** | **Status** | **Date** | ***Comment*** |
| v0.10 | Draft by CUSTDEV | 10/11/2021 |  |
| v0.11 | SfR to NPMs | 06/12/2021 |  |
| v1.00 | SfA to NPMs | 03/02/2022 | *Implementing comments #18 from ES.* |
| v1.10 | SfA to NPMs | 03/03/2022 | *SfA with implementation details* |
| v1.20 | SfA to NPMs | 06/04/2022 | *Implementing comment #56 from DDCOM 20.4.0-SfR Review cycle* |

1. Otherwise, validation could be successful but, if spaces are not removed before inserting in the database an error could occur. Database cannot be prepared to insert whatever length. [↑](#footnote-ref-1)
2. The affected External Domain IEs for NCTS-P5 are: IE007, IE013, IE014, IE015, IE017, IE026, IE034, IE044, IE054, IE141, IE170, IE224. The affected External Domain IEs for AES-P1 are: IE507, IE511, IE513, IE514, IE515, IE547, IE570, IE573, IE583, IE613, IE614, IE615. [↑](#footnote-ref-2)