# Impact Analysis Report

**Section 1: Meta-data**

|  |  |
| --- | --- |
| **RFC ID** | RFC\_NCTS\_0179(60124) |
| **Related Incident ID** | IM449575, IM450169 |
| **RFC Initiator / Organization** | CUSTDEV3 |
| **CI** | NCTS-P5 (DDNTA-5.14.1-v1.00 Appendices) |
| **Type of Change** | **Standard** **Emergency** |
| **Nature of Change** | Justification for Evolutive   |  | | --- | | NCTS-P5: Changes in the XSD patterns of alphanumeric data items, to ensure automatic validation of text fields’ length, triggered by **RFC\_DDCOM\_0027** (RTC-55494) | |
| **RFC Source** | |  |  | | --- | --- | | **Legal & Policy Change**  **Organisational Changes** | **Business Change**  **IT Change** | |
| **Review by Business User recommended?** | **Yes No** |

***Change Summary***

|  |
| --- |
| **DDNTA 5.14.1-v1.00: Appendix X - change the <xs:token> base type in all alphanumeric fields** |
| Based on RFC\_DDCOM\_0027 (RTC-55494), the following changes will take place in order to correctly manage the whitespaces within the text fields of an IE:   * Base type **<xs:normalizedString>** for alphanumeric data items (apart from those that are controlled by codelists or other specific strict pattern e.g. MRN) * On top of (a), the **<xs:pattern value="\P{Z}(.\*\P{Z})?"/>** only to ED IEs that are initiated from traders.   The affected External Domain IEs for NCTS-P5 that are initiated by the traders are the following: IE007, IE013, IE014, IE015, IE017, IE026, IE034, IE044, IE054, IE141, IE170, IE224. |

**Section 2: Problem Statement**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Currently, all alphanumeric data fields are based on <xs:token> base type according to which the leading, trailing and intermediate white spaces are “skipped” during the XSD validation.  As a results, there were cases where the total number of characters was erroneously being calculated.  For example, on CC115C message, the value of the Data Items <CONSIGNMENT-CONSIGNOR.Identification Number> there can be the following cases:   |  |  | | --- | --- | | Case | Example | | Case #1  No violation will occur.  Total number of characters is 17 | <CONSIGNOR>  <Identification number>*DE269854655123422*</Identification number>  </CONSIGNOR> | | Case #2  Violation will occur.  Total number of character is 19 which exceeds the format an..17 | <CONSIGNOR>  <Identification number>*DE269854655123422AB*</Identification number>  </CONSIGNOR> | | Case #3  No violation will occur.  Total number of characters is 22 (2 leading, 2 trailing and 2 intermediate spaces) but the leading and trailing spaces are skipped and the intermediate spaces are considered as a single space. | <CONSIGNOR>  <Identification number> *DE269854 55123422* </Identification number>  </CONSIGNOR> |   Table 1 - xs:token violation examples  The following alphanumeric data items were identified and categorized based on their format:   |  |  | | --- | --- | | Data Item | Format | | House number | an..17 | | Access code | an..4 | | Identification number | an..17 | | Postcode | an..17 | | Additional identifier | an..4 | | Authorisation number | an..35 | | Authority | an..35 | | Cancel enquiry notification text | an..512 | | City | an..35 | | Combined nomenclature code | an2 | | Complement of information | an..35 | | Controlled by | an..35 | | Declaration type | an..5 | | Description | an..512 | | Description of goods | an..512 | | Identifier | an..20 | | Discrepancies notification text | an..512 | | Error pointer | an..512 | | Error reason | an..7 | | Error text | an..512 | | Code | an..17 | | Container identification number | an..17 | | Guarantor notification text | an..512 | | LRN | an..22 | | Identification number | an..35 | | Conveyance reference number | an..17 | | Conveyance reference number | an..35 | | E-mail address | an..256 | | Invalidity reason text | an..512 | | Justification | an..512 | | Location | an..35 | | LRN | an..35 | | Reference number UCR | an..35 | | Reference number UCR | an..70 | | TIR holder identification number | an..17 | | MRN | an..35 | | Name | an..70 | | Other guarantee reference | an..35 | | Phone number | an..35 | | Reference number | an..35 | | Place | an..35 | | Access code current | an..4 | | Access code new | an..4 | | No release motivation text | an..512 | | Original attribute value | an..512 | | Master access code | an..4 | | Other things to report | an..512 | | Shipping marks | an..512 | | Text | an..512 | | Remark | an..512 | | Unloading remark | an..512 | | Recovery notification text | an..512 | | Reference number | an..70 | | Rejection reason | an..512 | | Status | an3 | | Street and number | an..70 | | Warning pointer | an..512 |   Table 2 – NCTS-P5 Identified alphanumeric data items  **NOTE:** The data items which contain codelists or a specific pattern for a specific message will not be updated as they are not causing violation whatsoever. For the extensive view of the identified items, please refer to the attached .xlsx file    Following the above analysis, the following XSD simple types are proposed to be applied for each identified category which will respect both the format and the applicability of white spaces.  These are the following:   |  |  | | --- | --- | | Simple Types | Description | | AlphaNumeric\_2 | Simple type that will be applied on alphanumeric fields with format an2 | | AlphaNumeric\_2\_Nospaces | Simple type that will be applied on alphanumeric fields with format an2 and with pattern that will not allow leading and trailing spaces, on external domain messages that are generated by traders | | AlphaNumeric\_3 | Simple type that will be applied on alphanumeric fields with format an3 | | AlphaNumeric\_MAX17 | Simple type that will be applied on alphanumeric fields with format an..17 | | AlphaNumeric\_MAX17\_NoSpaces | Simple type that will be applied on alphanumeric fields with format an..17 and with pattern that will not allow leading and trailing spaces, on external domain messages that are generated by traders | | AlphaNumeric\_MAX20 | Simple type that will be applied on alphanumeric fields with format an..20 | | AlphaNumeric\_MAX20\_Nospaces | Simple type that will be applied on alphanumeric fields with format an..20 and with pattern that will not allow leading and trailing spaces, on external domain messages that are generated by traders | | AlphaNumeric\_MAX22 | Simple type that will be applied on alphanumeric fields with format an..22 | | AlphaNumeric\_MAX22\_Nospaces | Simple type that will be applied on alphanumeric fields with format an..22 and with pattern that will not allow leading and trailing spaces, on external domain messages that are generated by traders | | AlphaNumeric\_MAX35 | Simple type that will be applied on alphanumeric fields with format an..35 | | AlphaNumeric\_MAX35\_Nospaces | Simple type that will be applied on alphanumeric fields with format an..35 and with pattern that will not allow leading and trailing spaces, on external domain messages that are generated by traders | | AlphaNumeric\_MAX4 | Simple type that will be applied on alphanumeric fields with format an..4 | | AlphaNumeric\_MAX4\_NoSpaces | Simple type that will be applied on alphanumeric fields with format an..4 and with pattern that will not allow leading and trailing spaces, on external domain messages that are generated by traders | | AlphaNumeric\_MAX5 | Simple type that will be applied on alphanumeric fields with format an..5 | | AlphaNumeric\_MAX512 | Simple type that will be applied on alphanumeric fields with format an..512 | | AlphaNumeric\_MAX512\_Nospaces | Simple type that will be applied on alphanumeric fields with format an..512 and with pattern that will not allow leading and trailing spaces, on external domain messages that are generated by traders | | AlphaNumeric\_MAX7 | Simple type that will be applied on alphanumeric fields with format an..7 | | AlphaNumeric\_MAX70 | Simple type that will be applied on alphanumeric fields with format an..70 | | AlphaNumeric\_MAX70\_Nospaces | Simple type that will be applied on alphanumeric fields with format an..70 and with pattern that will not allow leading and trailing spaces, on external domain messages that are generated by traders | | ContainerIdentificationNumber\_NoSpaces | Simple type that will be applied on container identification number with format an..17 and with pattern that will not allow leading or trailing spaces, on external messages that are generated by traders | | ConveyanceReferenceNumber17\_NoSpaces | Simple type that will be applied on conveyance reference number with format an..17 and with pattern that will not allow leading or trailing spaces, on external messages that are generated by traders | | EmailAddressType256\_NoSpaces | Simple type that will be applied on email addresses with format an..256 and with pattern that will not allow leading or trailing spaces, on external messages that are generated by traders | | ~~IdentificationNumberContentType02\_NoSpaces~~ | ~~Simple type that will be applied on identification number with format an..17 and with pattern that will not allow leading or trailing spaces, on external messages that are generated by traders~~ | | ~~TIRHolderIdentificationNumber\_NoSpaces~~ | ~~Simple type that will be applied on TIR holder identification number with format an..17 and with pattern that will not allow leading or trailing spaces, on external messages that are generated by traders~~ | | UCRReferenceNumber35\_NoSpaces | Simple type that will be applied on UCR Reference number with format an..35 and with pattern that will not allow leading or trailing spaces, on external messages that are generated by traders |   Table 3 – NCTS-P5 New Simple Types  **Further to the above, a number of existing simple types will be updated with <xs:normalizedString> base. These are the following:**   |  | | --- | | Simple Types | | ~~IdentificationNumberContentType02~~ | | ~~IdentificationNumberContentType03~~ | | ~~TIRHolderIdentificationNumberContentType~~ | | AlphaNumType | | AlphaType | | StringLatin1 |   Table 4: NCTS-P5 Update on existing Simple Types  **Impacted IEs:**  IE013,IE015,IE029,IE034,IE200,IE203,IE224,IE026,IE007,IE170,IE190,IE003,IE038,IE043,IE115,IE165,IE180,IE181,  IE182,IE059,IE001,IE004,IE009,IE012,IE017,IE019,IE023,IE028,IE035,IE037,IE045,IE050,IE054,IE055,IE056,IE060,  IE140,IE141,IE142,IE150,IE151,IE160,IE228,IE229,IE928,IE018,IE025,IE044,IE042,IE051,IE143,IE144,IE145,IE152,  IE022,IE057,IE906,IE917,IE014,IE201,IE205,IE225,IE231,IE191,IE903,IE975,IE070,IE071,IE971  **Impacted R/C/G/T/TRT/BRT:** N/A  **Impacted CIs Artefacts**:   * **DDNTA-5.14.1-v1.00 (only Appendix X): Yes** * **CSE-v51.6.0: Yes** * **TRP-5.8.0-v1.00: Yes** * **CRP-5.6.0-v1.0: Yes** * **CTS-5.6.2-v1.10: Yes** * CTP-5.8.0-v1.00: No * DMP Package-5.6.0-v1.00 (incl. update of file Rules and Conditions\_v0.43): No * DDNXA-5.14.1-v1.00 (Main Document): No * DDCOM 20.3.0-v1.00: No * ACS: 5.6.0-v1.00 & ACS-Annex-AES/NCTS: 5.6.0-v1.00: No * Functional Specifications (FSS/BPM): No * UCC IA/DA Annex B: No * ieCA 1.0.2.1: No * CS/MIS2\_DATA: No * CS/RD2\_DATA: No * AES-P1 and NCTS-P5 Long-Lived “Legacy” (L3) Movements Study v1.40: No |

**Section 3: Description of the proposed solution**

|  |
| --- |
| In the forthcoming release of DDNΤA Appendices the following updates shall take place (addition of **text highlighted in yellow**, removal of ~~text with strikethrough~~):   * The following simple type will be updated as follows:   For alphanumeric data items:  <xs:simpleType name="AlphaNumType">  <xs:annotation>  <xs:documentation>Base class for all anN and an..N types  </xs:documentation>  </xs:annotation>  <xs:restriction base="~~xs:token~~ xs:normalizedString"/>  </xs:simpleType>  For alphabetic data items:  <xs:simpleType name="AlphaType">  <xs:annotation>  <xs:documentation>Base class for all aN and a..N types  </xs:documentation>  </xs:annotation>  <xs:restriction base=" ~~xs:token~~ xs:normalizedString"/>  </xs:simpleType>  For string latin data items:  <xs:simpleType name="StringLatin1">  <xs:restriction base=" ~~xs:token~~ xs:normalizedString ">  <xs:pattern value="\p{IsBasicLatin}+" />  </xs:restriction>  </xs:simpleType>    ~~<xs:simpleType name="IdentificationNumberContentType02">~~  ~~<xs:restriction base=" xs:token xs:normalizedString ">~~  ~~<xs:pattern value=".{1,35}" />~~  ~~</xs:restriction>~~  ~~</xs:simpleType>~~    ~~<xs:simpleType name="IdentificationNumberContentType03">~~  ~~<xs:restriction base=" xs:token xs:normalizedString ">~~  ~~<xs:pattern value=".{1,17}" />~~  ~~</xs:restriction>~~  ~~</xs:simpleType>~~    ~~<xs:simpleType name="TIRHolderIdentificationNumberContentType">~~  ~~<xs:restriction base=" xs:token xs:normalizedString ">~~  ~~<xs:pattern value=".{1,17}" />~~  ~~</xs:restriction>~~  ~~</xs:simpleType>~~    **Please note that the simple type that were mentioned on the “Table 3 – NCTS-P5 Simple types” and not listed below are covered by the above mentioned updates**.   * The new simple types will be created as follows:   <xs:simpleType name="AlphaNumeric\_MAX7">  <xs:restriction base="AlphaNumType">  <xs:maxLength value="7" />  <xs:pattern value=".{1,7}" />  </xs:restriction>  </xs:simpleType>  <xs:simpleType name="AlphaNumeric\_MAX12">  <xs:restriction base="AlphaNumType">  <xs:maxLength value="12" />  <xs:pattern value=".{1,12}" />  </xs:restriction>  </xs:simpleType>  <xs:simpleType name="AlphaNumeric\_MAX256">  <xs:restriction base="AlphaNumType">  <xs:maxLength value="256" />  <xs:pattern value=".{1,256}" />  </xs:restriction>  </xs:simpleType>  <xs:simpleType name="AlphaNumeric\_MAX512">  <xs:restriction base="AlphaNumType">  <xs:maxLength value="512" />  <xs:pattern value=".{1,512}" />  </xs:restriction>  </xs:simpleType>  <xs:simpleType name="AlphaNumeric\_2\_NoSpaces">  <xs:restriction base="AlphaNumType">  <xs:length value="2" />  <xs:pattern value="\P{Z}(.\*\P{Z})?"/>  </xs:restriction>  </xs:simpleType>  <xs:simpleType name="AlphaNumeric\_4\_NoSpaces">  <xs:restriction base="AlphaNumType">  <xs:length value="4" />  <xs:pattern value="\P{Z}(.\*\P{Z})?"/>  </xs:restriction>  </xs:simpleType>  <xs:simpleType name="AlphaNumeric\_MAX4\_NoSpaces">  <xs:restriction base="AlphaNumType">  <xs:maxLength value="4" />  <xs:pattern value="\P{Z}(.\*\P{Z})?"/>  </xs:restriction>  </xs:simpleType>  <xs:simpleType name="AlphaNumeric\_MAX17\_NoSpaces">  <xs:restriction base="AlphaNumType">  <xs:maxLength value="17" />  <xs:pattern value="\P{Z}(.\*\P{Z})?"/>  </xs:restriction>  </xs:simpleType>  <xs:simpleType name="AlphaNumeric\_MAX20\_NoSpaces">  <xs:restriction base="AlphaNumType">  <xs:maxLength value="20" />  <xs:pattern value="\P{Z}(.\*\P{Z})?"/>  </xs:restriction>  </xs:simpleType>  <xs:simpleType name="AlphaNumeric\_MAX22\_NoSpaces">  <xs:restriction base="AlphaNumType">  <xs:maxLength value="22" />  <xs:pattern value="\P{Z}(.\*\P{Z})?"/>  </xs:restriction>  </xs:simpleType>  <xs:simpleType name="AlphaNumeric\_MAX35\_NoSpaces">  <xs:restriction base="AlphaNumType">  <xs:maxLength value="35" />  <xs:pattern value="\P{Z}(.\*\P{Z})?"/>  </xs:restriction>  </xs:simpleType>  <xs:simpleType name="AlphaNumeric\_MAX70\_NoSpaces">  <xs:restriction base="AlphaNumType">  <xs:maxLength value="70" />  <xs:pattern value="\P{Z}(.\*\P{Z})?"/>  </xs:restriction>  </xs:simpleType>  <xs:simpleType name="AlphaNumeric\_MAX256\_NoSpaces">  <xs:restriction base="AlphaNumType">  <xs:maxLength value="256" />  <xs:pattern value="\P{Z}(.\*\P{Z})?"/>  </xs:restriction>  </xs:simpleType>  <xs:simpleType name="AlphaNumeric\_MAX512\_NoSpaces">  <xs:restriction base="AlphaNumType">  <xs:maxLength value="512" />  <xs:pattern value="\P{Z}(.\*\P{Z})?"/>  </xs:restriction>  </xs:simpleType>  <xs:simpleType name="AlphaNumeric\_MAX12\_NoSpaces">  <xs:restriction base="AlphaNumType">  <xs:maxLength value="12" />  <xs:pattern value="\P{Z}(.\*\P{Z})?"/>  </xs:restriction>  </xs:simpleType>  <xs:simpleType name="ContainerIdentificationNumber\_NoSpaces">  <xs:restriction base="AlphaNumType">  <xs:minLength value="1" />  <xs:maxLength value="17" />  <xs:pattern value="[!-~][ -~]{1,15}[!-~]|[!-~]{1,2}" />  </xs:restriction>  </xs:simpleType>  <xs:simpleType name="ConveyanceReferenceNumber17\_NoSpaces">  <xs:restriction base="AlphaNumType">  <xs:minLength value="1" />  <xs:maxLength value="17" />  <xs:pattern value="[!-~][ -~]{1,15}[!-~]|[!-~]{1,2}" />  </xs:restriction>  </xs:simpleType>  <xs:simpleType name="EmailAddressType256\_NoSpaces">  <xs:restriction base="StringLatin1">  <xs:maxLength value="256" />  <xs:pattern value="\P{Z}[^@]\*@[^\.]+\..\*\P{Z}" />  </xs:restriction>  </xs:simpleType>  ~~<xs:simpleType name="IdentificationNumberContentType02\_NoSpaces">~~  ~~<xs:restriction base="xs:normalizedString ">~~  ~~<xs:pattern value="\P{Z}.{1,15}\P{Z}|\P{Z}{1,2}" />~~  ~~</xs:restriction>~~  ~~</xs:simpleType>~~  ~~<xs:simpleType name="TIRHolderIdentificationNumber\_NoSpaces">~~  ~~<xs:restriction base="xs:normalizedString ">~~  ~~<xs:pattern value="\P{Z}.{1,15}\P{Z}|\P{Z}{1,2}" />~~  ~~</xs:restriction>~~  ~~</xs:simpleType>~~  <xs:simpleType name="UCRReferenceNumber35\_NoSpaces">  <xs:restriction base="AlphaNumType">  <xs:minLength value="1" />  <xs:maxLength value="35" />  <xs:pattern value="[!-~][ -~]{1,33}[!-~]|[!-~]{1,2}" />  </xs:restriction>  </xs:simpleType>  **Impact on External Domain – Medium**  This RFC-Proposal concerns changes at syntactic level in all (Common and External Domain) messages.    It is considered that the change proposed via the current IAR has not impact on business continuity, since it transfers the principles of DDCOM 20.3.0-v1.00 (that all NAs should conform to) to the syntactic validation. Therefore, it can be deployed in a ***flexible way***.  More specifically:    **Changes at semantic level**   N/A    **Movement initiated under the previous DDNXA (5.14.1) release which continues its flow under the new DDNXA (5.15.0) release (open movement):**  No issues are expected to occur in open movement case.    **Changes at syntactic level**  This IAR concerns changes at syntactic level, as it describes the addition of patterns to the decimal data items. That practically means that:   * If the sender is aligned with the proposed changes, and the recipient is not, then no syntactic rejection will be caused, since the sender ensures the quality of data according to DDCOM 20.3.0-v1.00 principles.   + In case the recipient is not aligned to the DDCOM 20.3.0-v1.00 principles, then no rejection will be caused. * If the sender is not aligned with the proposed changes and the recipient is, then based on DDCOM, the sender should also ensure the quality of data, even if there is no syntactic check to verify this alignment to DDCOM 20.3.0-v1.00 “V.2.1.1.2 Text Fields” section. In this case no syntactic rejection shall be caused.   + In case the sender is not aligned to the DDCOM 20.3.0-v1.00 principles, then a syntactic rejection will be caused by the recipient.     **Impact in case of no Implementation**  In case of not implementing this change, the validation of DDCOM principles regarding alphanumerical fields should be performed by the NAs by implementing extra rules/checks to verify the quality of the data that are exchanged.  **Proposed** date of applicability in Operations **(T-ops)**: As soon as possible, before 1.12.2023 **(*flexible*)**  **Proposed** date of applicability in CT **(T-CT)**: July 2022  **Expected** date of approval by ECCG **T-CAB)**: January 2022  **Impact on transition NCTS-P4-NCTS-P5**: None  **Consequence of not approving the RFC-Proposal:** as stated above, NAs shall validate the DDCOM principles regarding numerical fields  **Risk of not implementing the change: No** |

**Impact on CI Artefacts**

|  |  |  |
| --- | --- | --- |
| DDNXA-5.14.1-v1.00 (only Appendix X) | Cosmetic  Low  Medium  High  Very High  Short description   |  | | --- | | Updates in DDNXA Appendix X (ctypes.xsd and stypes.xsd) as described in section 3 | |
| CSE-v51.6.0 | Cosmetic  Low  Medium  High  Very High  Short description   |  | | --- | | Updates as described in section 3. | |
| CTP-5.8.0-v1.00 | Cosmetic  Low  Medium  High  Very High  Short description   |  | | --- | | Alignment of messages according to the updates of specifications. | |
| TRP-5.8.0-v1.00 | Cosmetic  Low  Medium  High  Very High  Short description   |  | | --- | | Alignment of messages according to the updates of specifications. | |
| CRP-5.6.0-v1.0 | Cosmetic  Low  Medium  High  Very High  Short description   |  | | --- | | A new version of CRP will be published due to the updates of its components. | |

**Estimated impact on National Projects**

|  |  |
| --- | --- |
| Cosmetic  Low  Medium  High  Very High  Short description   |  | | --- | |  | |

**Document History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Document History** | | |  |
| **Version** | **Status** | **Date** | ***Comment*** |
| v0.10 | Draft by CUSTDEV | 17/02/2022 | *First draft prepared by CUSTDEV.* |
| v1.00 | SfA to NPMs | 01/03/2022 | *Updates in turquoise with implementation details.* |