**Machine Readable Table**

**(MRT)**

**Definitions**

November 19, 2015

Version 1.9

Contents

[Introduction 2](#_Toc435710117)

[What are Machine Readable Tables (MRTs)? 2](#_Toc435710118)

[What is in this Document? 2](#_Toc435710119)

[General Guidelines for Tables 3](#_Toc435710120)

[Current Status 3](#_Toc435710121)

[MRTs and Related Documents 4](#_Toc435710122)

[FieldDefinition 6](#_Toc435710123)

[LookupCodes 19](#_Toc435710124)

[TOT Records 21](#_Toc435710125)

[TOTfields 22](#_Toc435710126)

[TOTFieldDetail 23](#_Toc435710127)

[RecordTypeDesc 25](#_Toc435710128)

[Appendix A: MRTVerify Error Codes 26](#_Toc435710129)

# Introduction

## What are Machine Readable Tables (MRTs)?

The Machine Readable Tables are a set of files which which take table-based definitions for fields and transactions as defined in the ANSI/NIST-ITL standard and associated application profiles (e.g. EBTS, INT-I) and convert them into a machine-readable format. The purpose is to have standard tables that are hosted as a standard machine-readable definition of all the content in the various field tables in ANSI/NIST-ITL 1-2011 and 1-2013, with separate files for FBI EBTS (identical format for ANSI/NIST tables, but expanded to cover transactions). The set of files is designed for any system needing to parse, edit, create, and/or check ANSI/NIST-derived transactions. The master format of the MRTs is XML (EXtensible Markup Language), which can be easily manipulated into various other formats.

## What is in this Document?

This document describes the format of the MRTs. It explains the information contained in each MRT, defines the MRT column headers, and lists additional discussions, examples, and issues. The following types of MRT documents are explained and defined:

[FieldDefinition 6](#_Toc435710123)

[LookupCodes 19](#_Toc435710124)

[TOT Records 21](#_Toc435710125)

[TOTfields 22](#_Toc435710126)

[TOTFieldDetail 23](#_Toc435710127)

[RecordTypeDesc 25](#_Toc435710128)

## General Guidelines for Tables

* Each MRT is named with the standard/application profile name followed by one of the MRT types mentioned above. For example, ANSI/NIST-ITL 1-2011 contains the *AN2011-FieldDefinition, AN2013e (2011+)-FieldDefinition, AN2011-RecordTypeDesc,* and *AN2011-LookupCodes* MRT, and EBTS v10.0.5 contains the *FBIEBTS10.0.5-FieldDef*, *FBIEBTS10.0.5-LookupCodes*, *FBIEBTS10.0.5-TOTrecord*, *FBIEBTS10.0.5-TOTfields*, *FBIEBTS10.0.5-TOTFieldDetail,* and *FBIEBTS10.0.5-RecordTypeDesc* MRT.
* In general, entries in all tables are limited to 7-bit ASCII characters: characters represented on standard US keyboards (from 0x20 (space) to 0x7E (~)).
* Quotation marks shall always use the straight marks from the keyboard, not the “curly” quotation marks substituted by some editing software. Likewise, dashes shall always use the minus sign from the keyboard, not the em-dash and en-dash substituted by some editing software.
* The field definition rules are additive and a field must pass all of them to be in compliance.  Therefore, in the *FieldDefinition* MRTs, a field must comply with each of the columns: ValueRange, CodeTable, and RegExpression. For example, if a CodeTable XX contains the values 1, 2, and 3, and the ValueRange states “{1,2}” (meaning that either value ‘1’ or ‘2’ is allowed), this means that the field is only allowed to contain either value ‘1’ or ‘2’ from the CodeTable XX. The field would have to comply with both the CodeTable and ValueRange in this case.

## Current Status

The following table shows the status of the standards/application profiles:

|  |  |
| --- | --- |
| Standards/Application Profile | Status |
| ANSI/NIST 2011 & 2011plus (includes error corrections in 2013 – “2013e”) | **Complete** |
| ANSI/NIST 2013 | **Complete** |
| ANSI/NIST 2015 | **To be completed when final version is published** |
| EBTS v9.3 | **Complete** |
| EBTS v10.0.5 (includes TOU # 1, 2, 3, 4 and 5) | **Complete** |
| Iris Pilot EBTS v10.0.5 (includes requirements from the EBTS Iris Pilot) | **Complete** |

## MRTs and Related Documents

1. The **ANSI/NIST MRTs** include\*:
   1. *AN2011-FieldDefinition* – Lists all fields for ANSI/NIST 2011 without any error corrections
   2. *AN2011-LookupCodes* – Lists table codes in ANSI/NIST 2011
   3. *AN2013e (2011+)-FieldDefinition*– Corrects errors (based on ANSI/NIST 2013 error corrections) of fields in *AN2011-FieldDefinition* ; also known as “ANSI/NIST 2011**+**”
   4. *AN2011-RecordTypeDesc* – Describes Record Types in ANSI/NIST 2011
   5. *AN2013na-FieldDefinition –* Lists new information/fields for ANSI/NIST 2013 (lists only fields/subfields/information items which are different from AN2011)
   6. *AN2013-LookupCodes* – Lists table codes in ANSI/NIST 2013
   7. *AN2013-RecordTypeDesc* – Describes Record Types in ANSI/NIST 2013
2. The **EBTS v10.0.5 MRTs** include:
   1. *FBIEBTS10.0.5-FieldDefinition* – Lists all fields for EBTS v10.0.5 (EBTS v10.0.5 includes TOU #1, 2, 3, 4 and 5)
   2. *FBIEBTS10.0.5-LookupCodes* – Lists table codes in EBTS v10.0.5
   3. *FBIEBTS10.0.5-TOTrecords* – Data from EBTS v10.0.5: Tables L-3 and L-4
   4. *FBIEBTS10.0.5-TOTfields* - Data from EBTS v10.0.5: Tables D-1, E-1, and I-1
   5. *EBTS10.0.5TOTFieldDetail* – Information from EBTS v10.0.5 Appendix D Footnotes
   6. *FBIEBTS10.0.5-RecordTypeDesc* – Describes Record Types in EBTS v10.0.5
3. The **Iris Pilot EBTS v10.0.5 MRTs\*\* (Iris Pilot requirements that are in addition to or different from EBTS v10.0.5)** include:
   1. *IRIS\_EBTS10.0.5-FieldDefinition* – Lists fields for EBTS v10.0.5 (EBTS v10.0.5 includes TOU #1, 2, 3, 4 and 5 and the Iris Pilot Reqs)
   2. *IRIS\_EBTS10.0.5-LookupCodes* – Lists table codes in EBTS v10.0.5
   3. *IRIS\_EBTS10.0.5-TOTrecords* – Data from EBTS v10.0.5: Tables L-3 and L-4
   4. *IRIS\_EBTS10.0.5-TOTfields* - Data from EBTS v10.0.5: Tables D-1, E-1, and I-1
   5. *IRIS\_EBTS10.0.5TOTFieldDetail* – Information from EBTS v10.0.5 Appendix D Footnotes
   6. *IRIS\_FBIEBTS10.0.5-RecordTypeDesc* – Describes Record Types in EBTS v10.0.5
4. The **EBTS 9.3 MRTs** include:
   1. *FBIEBTS9.3-FieldDefinition* – Lists all fields for EBTS v9.3
   2. *FBIEBTS9.3-LookupCodes* – Lists table codes in EBTS v9.3
   3. *FBIEBTS9.3-TOTrecords* – Data from EBTS v9.3: Tables L-3 and L-4
   4. *FBIEBTS9.3-TOTfields* - Data from EBTS v9.3: Tables D-1, E-1, and I-1
   5. *FBIEBTS9.3TOTFieldDetail* – Information from EBTS v9.3 Appendix D Footnotes
   6. *FBIEBTS9.3-RecordTypeDesc* – Describes Record Types in EBTS v9.3
5. ***MRT Definitions\_v1.9***  (this document)
6. **Xxxx-KnownErrors.xml**
   1. For each version of the specification/standard, there is an XML file called “KnownErrors”. This file contains the errors found by the MRT Verify program which checks the content, structure, and syntax of each MRT. The errors shown in these files are exceptions to the normal syntax of the MRTs and do not need correction. Use Appendix A as a reference for MRT Verify codes.
7. **Configurations.xml**
   1. This file lists standard/specification versions and their parent standard/specifications, and the MRTs which correspond to that version.
8. **All\_Elements\_xxxx.xml**
   1. All the possible XML elements allowed in a certain version of the standard or specification.
9. **[MRT].xsd**
   1. The XSD file for the MRT
10. **[MRT].xslt**
    1. The XSLT file for the MRT; used to format the XML information
11. **mrt.css**
    1. CSS for all MRTs; adds background color and table borders

\*Note 1:

* ANSI/NIST 2011 consists of the following MRTs: *AN2011-FieldDefinition*, *AN2011-LookupCodes*, *AN2011-RecordTypeDesc,* and *AN2013e (2011+)-FieldDefinition*.
* ANSI/NIST 2013 consists of the following MRTs: *AN2011-FieldDefinition*, *AN2013e-FieldDefinition*, *AN2013na-FieldDefinition*, and *AN2013-LookupCodes*. The AN2013 FieldDefinition MRT (contains new and changed fields to AN2011) layers on top of the AN2011 Field Definition MRTs.

\*\*Note 2:

* The EBTS Iris Pilot MRTs layer on top of the EBTS v10.0.5. Therefore, to include the Iris Pilot requirements in the MRT, use the Iris Pilot EBTS MRTs as well as the EBTS v10.0.5 MRTs.

# FieldDefinition

The *FieldDefinition* MRT defines columns for a general-purpose table-based definition for each field in the record types, as defined in application profiles and ANSI/NIST-ITL. For ANSI/NIST, it contains all fields found in the standard. For application profiles, it includes fields which exist in the application profile only (not defined in ANSI/NIST) and ANSI/NIST fields which were edited by the application profile. Unless otherwise stated, all examples are derived from the ANSI/NIST-ITL 2011 MRT.

There are a few open issues which have yet to be resolved:

* CodeTable: How do we want to address CodeTables not maintained by NIST (e.g., a. such as GENC (NIEM maintains), DNA, CODECS, IBIA quality algorithms, SMT (FBI maintains)? Possible suggestions include
  + NIST houses the AN201x MRTs and contains the code tables that NIST is comfortable that they can keep up to date
  + For tables that other organizations maintain that NIST is concerned may get out of sync, caveat with text such as “Table X contains content maintained by <organization>, and includes the most recent updates received by NIST as of <date>. If more recent updates are available, please contact…” and house separate MRTs for each of those tables which are included in the AN201x MRTs (if the threshold for updating isn’t a major burden)
* Summary: Should there be a maximum length and if so how long?
* XMLPath: Should this start at *itl:NISTBiometricInformationExchangePackage* or at the record level (e.g. *itl:PackageInformationRecord*)? The XML parent currently starts at the record level.

| Column Name | Column Definition | Required in MRT | Example |
| --- | --- | --- | --- |
| TradXML | States whether the field/row is formatted as XML only (“X”), Traditional only (“T”), or both XML and Traditional (“TX”).  Note: In application profiles, there are rows that contain “REMOVE: FieldRef = XX/XXX:X”. These designate a field in the parent profile (i.e., ANSI/NIST), that must be removed for the application profile to be processed correctly. E.g., EBTS v10.0 has “REMOVE: FieldRef=10/SMD:T” before field 10.042. This means that when using EBTS v10.0 as an MRT, field 10/SMD:T in the ANSI/NIST MRT must be removed and replaced by field 10.042 in EBTS v10.0. | Required | TX  (for 1.003)  X  (for 1.002:X2) |
| FieldRef | FieldRef is a unique reference to a row in this table and equals the string [RecordNum/FieldMnemonic/InfoItemMnemonic]. A unique reference to this field or info item is necessary because field mnemonics are only unique within their record, and infoitem mnemonics are only unique within their field.  Special cases:   * When a single field/info item is represented with more than one row (ContentType=Data\_T or Data\_X), the row for the traditional field is suffixed ":T" and the row for the XML field is suffixed with ":X"(e.g., 01.001:X) * If one traditional field corresponds to multiple XML elements (ContentType= Data\_NX-1T, or Set\_X), the constituent XML elements are suffixed with ":X1", ":X2", etc. * If an XML path applies to multiple fields or infoitems, the format shall be “[lowest field]\_[highest field]”. Note that the XML path does not necessarily apply to all fields within that range. This is used to represent an XML parent which encompasses many different fields or an XML element and path which applies to multiple fields. | Required | 9/MIN  (for 9.331, which contains multiple information items)  9/MIN/MRU  (fifth information item in 9.331)  1/VER:X1  (first XML element for 1.002)  1/LEN:T  (for field 1.001:T)  4/IMP\_4/DATA  (XML set which applies to fields 4.003 through 4.009) |
| FieldID | FieldID is a unique reference to a row in this table. It uses the field numbers and field order traditionally used in ANSI/NIST, formatted as [Record.Field number-InfoItem]. Three decimal places are always used (e.g. 01.004 or 14.016), with the exception of record two in which up to 4 decimal places are allowed (e.g. FBI EBTS 02.2033).  Special cases:   * If an XML path encompasses multiple fields and/or infoItems and contains no data, the format shall be “[lowest field]\_[highest field]” * If the FieldID is not otherwise unique, suffixes are added after a colon as described in FieldRef (e.g. “:X”, “:T” “:X1”) | Required | [01.004]  [01.015-A]  [01.003-2A\_01.003-2B]  [01.011:T]  [01.011:X]  [09.300-E:X2] |
| InfoItem | Information items are lettered explicitly, starting with "A". "SET" indicates the field contains one or more sets of information items.  Note: "Information Item" is used as defined in ANSI/NIST; note that EBTS uses the term "subfield" to refer to these entities. For this reason, we do not use the term "subfield" here, but use the term "occurrence" to refer to repeated instances of a field (as used in EBTS). | Optional | A  (=1st infoitem in RecordNum.FieldNum)  SET  (= header row for a field that has InfoItems) |
| InfoItemCount | If InfoItem="SET", this indicates the number of unique information items included.   Special cases:   * If InfoItemsVary="TYPE", this indicates the number of information items that can be contained in a given occurrence of a field * If InfoItemsVary="NUM" this is omitted | Optional | 7  (for 9.331, the field contains 7 InfoItems, lettered A..F)  2  (for 1.003, which has InfoItemsVary="TYPE" and 2 info items per occurrence)  Omitted  (for 14.025, which has InfoItemsVary="NUM") |
| InfoItemsVary | Information item exception (rare: this exists solely to address a few ANSI/NIST and EBTS special-case fields). In general, a field with information items contains a defined list of information items that are repeated for each occurrence. This deals with the exceptions.   * InfoItemsVary = "NUM" to indicate a field in which the number of information items varies (e.g., 14.025), or to indicate the information items that vary in such a field (e.g., 14.025-C and 14.025-D, but not 14.025-A and 14.025-B) * InfoItemsVary = "TYPE" to indicate a field where the type of information items differs by a field’s occurrence (e.g., 1.003). For information items that vary by occurrence, this indicates the field occurrence number (e.g., "1" would represent the first occurrence of the field), or ">=" followed by an occurrence number (e.g., ">=2" which means that the information item is only allowed to occur in the second, third, etc. occurrence of the field – this information item would not be allowed in the first occurrence of the field). | Optional | NUM  (for 14.025, which contains a variable number of infoitems (2+ # of vertices)  NUM  (for FBI EBTS 2.074)  NUM  (for 14.025-C, which is a repeating infoitem)  TYPE  (for 1.003, which has different information items for the first occurrence than the other occurrences)  1  (for 1.003-A or 1.003-B, which are only used in the first occurrence)  >=2  (for 1.003-C or 1.003-D, which are used for the 2nd and greater occurrences) |
| Mnemonic | Three-letter Field Identifier as stated in ANIS/NIST-ITL or application profile. May be repeated within a field or InfoItem that consists of more than one row in the MRT (e.g., 1.002), but is unique for each field in a record type (e.g., mnemonic VER is used for field 1.002 in record 1 and for no other record 1 field).  Note: XRCC was created as a mnemonic for xx.001 fields since the original mnemonic RCC is not unique in T9. | Required, except for record XML headers (e.g., [01]) and XML parents encompassing multiple fields (e.g., 1/VER\_1/ANM) | VER  (for 1.002)  NSR  (for 1.011:X) |
| Description | Definition of field, identical to Content Description in ANSI/NIST, Field Name in EBTS, or other description in an application profile.  For XML elements that do not correspond directly to traditional fields/info items (and therefore have no definition), copy the XML element name or traditional field description and suffix “(XML)” | Required | Destination Agency Identifier  (for field 1.007) |
| CondCode | M = Mandatory O = Optional M^ = Mandatory InfoItem (or XML element) within the field O^ = Optional InfoItem (or XML element) within the field D = presence is dependent upon other fields/information items, or other conditions, detailed in one of the following: DependPresenceReq, DependPresenceOpt, DependAbsence, DependValue, DependValueReq, DependValueOpt, DependOther, Exception  Note:   * The caret (^) symbol is used here rather than the up arrow used in ANSI/NIST so that the contents of this file are always parseable as 7-bit ASCII. * The caret (M^ or O^) is only used for XML elements that are enclosed in a field (i.e. are groups or subdivisions of InfoItems within a field). | Required, except for Type-2 fields in EBTS | M  (for 1.003)  M^  (for 1.003-A)  D  (for 9.126:T) |
| DataType | Derived from the ANSI/NIST Character Type, but with additional delineation of data types. Omitted if ContentType = "Set\_X", “Set\_T” or “Set. See ANSI/NIST section 5.5. For dates, see ANSI/NIST sec 7.7.2.  Data Types defined in ANSI/NIST 2011 section 5.5:   * A = Alphabetic: 26 English letters, both upper and lower case) or spaces * N = Numeric (1 2 3 4 5 6 7 8 9 0) * AN = Alphanumeric: Alphabetic and numeric * S = used in combination with A, N, or AN representation to indicate Special characters. The specific characters permitted are listed in SpecialChar and SpecialNonPrintingChar * B = Binary for Traditional or Base64 for XML * Base64 = Base-64 encoded (exclusively), for both traditional and XML * H = Hexadecimal (0..9, A..F) * U = Unicode characters   Data Type unique to MRTs:   * Y = yes/no, Boolean: field indicated as Y|N in traditional, true|false or 1|0 in XML | Required, except if ContentType = "Set\_X", “Set\_T” or “Set” | ANS  (for 1.007)  D  (for 1.005:X)  B  (for 10.999)  GMT  (for 1.014:T)  Base64  (for 9.372)  U  (for 9.127-B)  H  (for 9.126-C) |
| MinLength | Minimum number of characters. Does not include separators for traditional encoding. | Required, except if ContentType = "Set\_X", “Set\_T” or “Set | 5  (for 1.011:T) |
| MaxLength | Maximum number of characters. Does not include separators for traditional encoding. Set to "\*" if there is no maximum. | Required, except if ContentType = "Set\_X", “Set\_T” or “Set | 20  (for 1.014:X)  \*  (for 1.008) |
| MinOccur | Minimum number of occurrences (cardinality) of the field. Set to 1 if the field/info item is required, 0 if optional. | Required, except for EBTS Type-2 fields (required for Type-2 Info Items) | 1  (for 1.009)  0  (for 1.010) |
| MaxOccur | Maximum number of occurrences (cardinality) of the field. Set to "\*" if there is no maximum.  A field with >1 maximum occurrences will have multiple subfields (using the ANSI/NIST definition); these subfields repeat sets of information items (if InfoItem="SET"). For example, in AN 2011, field 1.016 has a min and max occurrence of 0..1 for the field and 1..99 for repeating subfields. Therefore, in the MRT, the min and max occurrence is set to 99. | Required | 1  (for 1.015-B)  \*  (for 9.320) |
| SpecialChar | For fields with DataType “S”, this explicitly lists the permissible printable characters in brackets to prevent parsing issues in spreadsheets (which otherwise would have issues with quotes). Characters listed here must be in the ASCII range chr(33) through chr(126) (! through ~ - See AN2013 Table 116)  Special Case:   * If all printable 7-bit ASCII characters are permitted, set SpecialChar="ALL". Note that "ALL" is not put in brackets and does not include blanks, which must be explicitly specified in NonPrintChar. | SpecialChar and/or NonPrintChar required for fields with DataType “S” (i.e., DataType = AS, ANS, NS, S) | ALL  (for field 1.007)  [,-]  (for field 9.357-B:T) |
| NonPrintChar | For fields with DataType “S” that permit non-printing characters, this can be set to one or more of the following:   * \s (space) * \t (tab) * \n (newline)   Multiple values are comma separated. It is assumed that \n permits CR and/or LF (chr(13) and chr(10)) | SpecialChar and/or NonPrintChar required for fields with DataType “S” (i.e., DataType = AS, ANS, NS, S) | \s  (spaces are permitted, for field 1.017)  \s,\t,\n  (spaces, tabs and newlines, are permitted, for field 2.2102 in EBTS) |
| ContentType | Denotes how the data of a unique FieldID/FieldRef is formatted.  If TradXML=”TX”, then the following values are allowed:   * “Set” – Encloses infoitems and XML elements that contain data * “Data” – Field represented by in traditional and XML format; contains data   For those fields which there is not a 1:1 correspondence (TradXML = “X” or “T”) between the traditional fields/infoItems and XML tags, there are codes are used to indicate categories of differences:   * "Data\_1T-NX"- single traditional field/infoItem that is represented as multiple XML tags (XML path includes all such elements; counterpart to Data\_NX-1T) * "Data\_NX-1T"- XML tag that represents a portion of a single traditional field/infoItem (counterpart to Data\_1T-NX) * "Data\_1X-NT"- single XML field that corresponds to multiple traditional fields/infoitems (counterpart to Data\_NT-1X) * "Data\_NT-1X"- traditional field/infoitem that represents a portion of an XML element (counterpart to Data\_1X-NT) * "Data\_T" - Traditional field not represented in XML (e.g. record length fields) * "Data\_X" - XML field not represented in traditional format (e.g. record category code) * "Set\_T" - traditional field header (InfoItem="SET") for which there is no XML equivalent, because XML refers to the constituent info items using a different structure; encloses infoitems that contain data. * "Set\_X" - XML group element for which there is no traditional equivalent; encloses XML elements that contain data. * "Xreuse" - XML path/element that corresponds to more than one traditional field/infoitem (e.g., in EBTS when an old field corresponds to a new field). Xreuse may be indicated in combination with another XML exception (commas delimited, e.g. "Data\_NX-1T,Xreuse"); the other values are all mutually exclusive. | Required | "Data\_1T-NX"  (for 9.300-E)  "Data\_NX-1T"  (for 09.300-E:X3, which is a part 9.300-E)  "Data\_1X-NT"  (for 10.043-B\_10.043-F)  "DATA\_NT-1X"  (for 10.043-B:T)  "Data\_T"  (for 1.001:T)  "Data\_X"  (for 1.001:X)  "Set\_T"  (for 1.017:T)  "Xreuse"  (for ebts:CandidateMatchScoreValue, which corresponds to both 2.089 and 2.2033-G) |
| XMLPath | The XML path for a field. Includes the XML parent and element.   * The XML path is relative to the record type. In the first example shown, the actual XML path for itl:PackageInformationRecord/biom:RecordCategoryCode is itl:NISTBiometricInformationExchangePackage/itl:PackageInformationRecord/biom:RecordCategoryCode. itl:NISTBiometricInformationExchangePackage is not listed in the XML path since it precludes all record types. * There are cases when more than one XML path is allowed depending on the values of a specified field. For these exceptions, use a conditional statement in the format:   if [FieldRef]in{ValueRange} then XMLPath1  elseif [FieldRef]in{ValueRange} then XMLPath2  else XMLPath3}   * In place of a value range, a code table may be referenced by using <CodeTable>. For example, if an XML path is dependent on a value in the TMC CodeTable in ANSI/NIST, the user would use “<TMC>” (see example). * One "if" statement is required, followed by at least one "elseif" or "else". * There may be any number of "elseif" statements, and zero or one "else" statements. | Required for TradXML fields with X or TX | itl:PackageInformationRecord/biom:RecordCategoryCode  (for 1.001:X)  if [20/AQS/AQT] IN {7..10, 19..21} then itl:PackageSourceRepresentationRecord/biom:SourceVideo/biom:ImageSegment  else itl:PackageSourceRepresentationRecord/biom:SourceImage/biom:ImageSegment  endif  (for 20.016)  if [10/TMC] IN {<TMC>} then itl:PackageFacialAndSMTImageRecord/biom:FaceImage/biom:FaceImageFeaturePointTierCode  else itl:PackageFacialAndSMTImageRecord/biom:FaceImage/biom:FaceImageFeaturePointTierNumeric  endif  (for 10.031) |
|  | If the permissible values are restricted beyond all combinations defined by the data type and the min-max length, the permissible values are defined by one or more of these methods: CodeTable, CodeTableOpt, ValueRange, RegExpression, the Depends\*methods, or the InterField\* methods. |  |  |
| CodeTable | For fields limited to a fixed set of values, CodeTable contains text to point to the specific lookup table - see the LookupCodes table. ValueRange may be used to refer to a subset of CodeTable (e.g., to refer to finger positions, code table can be set to FGP and value range can be set to [1..10]).  Notes:   * In determining whether to use CodeTable or ValueRange, use CodeTable if each value has a specific definition. * In the unusual case in which a single field points to two tables (e.g., 10.026), the code table names are separated by a semicolon (";"). A semicolon indicates that the field can use codes from one or more of the code tables listed. Semicolons shall not be used in code table names. | Optional | SAP  (field 10.013 would set CodeTable to "SAP", and the LookupCodes table would contain the contents of Table 10 in ANSI/NIST under the CodeTable value "SAP")  SXS;SMT  (for field 10.026, which allows facial description codes and/*or* SMT codes) |
| CodeTableOpt | CodeTable Options:   * Set to "Opt" if the CodeTable is optional (defining commonly-used or special-purpose values) rather than definitive (listing all possible values). * Set to "Part" if the code table applies only to some of the characters of the field (example: EBTS state ID (2/SID) has 2-character state abbreviations, followed by an alphanumeric, so the code table includes entries for e.g., AL, AK, etc., which only apply to the initial characters of the field). | Optional | Opt  (for 1.015-A)  Part  (for 9.308) |
| ValueRange | Defines the permissible values of a field, specified as one or more of the following:   * {>=value} represents minimum value * {<=value} represents maximum value * {value} represents a single value * {value1,value2,value3} represents multiple values * {value1..value2} represents a range   Notes:   * Values to be compared as strings are enclosed in quotes (e.g., {"A".."Z"}). * A given range cannot contain both numeric and non-numeric data - e.g., these are not acceptable ranges: {1.."A"} or {0.."32UTF"} * Numbers can be negative or contain decimals (e.g., {-22.5..22.5}) * Numbers that are not enclosed in quotes are compared as numerics (e.g. 7 is less than 10), not strings (e.g., "7" is greater than "10"). * For ranges of strings enclosed in a single bracket (e.g., {"ABC".."BDE"}), the values must be of the same length and only may include uppercase letters and digits, and the second string must be greater than the first in ASCII order. (e.g., these are NOT permissible ValueRanges: {"abc".."B1","Z".."A"} * When multiple attributes restrict values (e.g., ValueRange and InterFieldValue and CodeTable), the valid fields must satisfy ALL of the restrictions. For example a field with ValueRange={0..49999} and InterFieldValue<9/ROI/EWI must satisfy both. | Optional | {0500}  (for 1.002:T)  {2,4,7..10,13..21,98,99}  (for 1.003-2A)  {"PRX","DST","MED","UNK"}  (for 9.302-B)  {"U"}  (for 9.004:T)  {>=1}  (for 9.137-B)  {1..99,"L","U"}  (for 9.322-A) |
| RegExpression | Includes a regular expression. For character-level value constraints that cannot be defined precisely in the combination of DataType, CodeTable, SpecialChar, and ValueRange.  Format TBD: will most likely be a defined subset of Perl/Java regular expression syntax (most modern regular expression formats are derived from Perl). | Optional | (M[0-9A-F]{12})|([P]\S{12,15})  (for 9.903 DUI: M followed by 12 hexadecimal characters, or P followed by 12 to 15 non-whitespace characters) |
|  | For fields where CondCode="D", one of the following is selected: DependPresenceReq, DependPresenceOpt, DependAbsence, DependValue, DependValueReq, DependValueOpt, or DependOther. |  |  |
| DependPresenceReq | This field is required if another field is present (and is not permitted otherwise). One or more FieldRefs are included, separated by commas. All fields listed must be present for this field to be required. | Optional | 14/DATA  (for 14.010, which is mandatory if an image is present in 14.999 DATA) |
| DependPresenceOpt | This field is optional but is only permitted if another field is present. One or more FieldRefs are included, separated by commas. All fields listed must be present for this field to be permitted. | Optional | 18/DGD  (Field 18.021 is optional and only permitted if field 18/DGD, 18.020, is present) |
| DependAbsence | This field is only permitted if another field is absent. One or more FieldRefs are included, separated by commas. All fields listed must be absent for this field to be permitted. | Optional | 9/MIN  (for 9.334, which is not included if 9.331 MIN is used) |
| DependValue | This field is required if another field is set to a specific value (and otherwise shall not be entered).   * One or more FieldRefs are included followed by a comparison sign * Multiple FieldRefs are separated by a comma * Value is in quotes if alphabetic   The following formats are allowed:   * For a single string of characters (alphabetic): [FieldRef] = “XX” * For a single integer: [FieldRef] = ## * If value is one of several values: [FieldRef]IN{“X”, “Y”, “Z”} or [FieldRef]IN{1,2,3} * If value is one in a range: [FieldRef]IN{1..19,40..50} * For single value inequalities: [FieldRef]>##   Note: comparison sign can be "<", ">","<=", ">=" ,"!="   * For multiple value inequalities: ##<[FieldRef]<## * Note: comparison sign can be "<", ">","<=", ">=" ,"!=" | Optional | [10/POS]="D"  (for 10.025, which shall be required when 10.020 POS = "D"; otherwise should not be present)  [8/SRT]<2  (for 08.008\_08.008:X1)  [9/MFD/FME]IN{"AUTO","REV","EDIT"}  (for 9.350-C)  [18/DLS/UTY]IN{1,2}  (for 18.003-C:T) |
| DependValueReq | This field is required if another field is set to a specific value (but is otherwise optional). See DependValue for more information on formatting. | Optional | [10/SAP]>=40  (for 10.023, which shall be required if 10.013 is greater than or equal to 40; otherwise 10.023 is optional)  [14/FGP]=19  (for 14.014) |
| DependValueOpt | This field is optional if another field is set to a specific value (but otherwise shall not be entered). See DependValue for more information on formatting. | Optional | [9/FPP/FGP]IN{0..10,16..17,20..38,81..84}  (for 9.342-E, which is optional only if 9/FPP/FGP, or 9.302-A, contains the listed values; otherwise 9.342 shall not be present)  [9/CBI/CFT]IN{514,515}  (for 9.139)  [10/IMT]="FACE",[10/SAP]IN{0..20} (for 10.014) |
| DependOther | Set to "Y" to indicate that this field is only permitted based on other dependencies not addressed in DependPresence or DependAbsence. A compliance checker must incorporate special logic for such fields. An explanation must be included in TechNotes. | Optional | Y  (for 8.005) |
|  | For fields where infoitems are dependent on other infoitems or fields, one of the following is selected: InterFieldOccur, InterFieldCount. InterFieldValue, InterFieldOther, Exception. Note that if a FieldRef listed in one of these does not exist in a particular transaction, then ignore the inter field reference in that instance (e.g., in EBTS v10.0.5, the EQRR TOT contains field 2.079, but not 2.2010. Therefore, the InterFieldValue in field 2.079 would be ignored in this case). |  |  |
| InterFieldOccur | For fields where the number of occurrences of the field must be equal to the number of occurrences of another field. One or more FieldRefs are included, separated by commas. If InterFieldOccur contains a FieldRef which does not exist in the transaction, then then InterFieldOccur does not apply to the field in that particular instance. | Optional | 2/ASL/AOL  (for 2.047-A) |
| InterFieldCount | For fields where the number of occurrences of the field must be equal to the numeric value of another field. One FieldRef is included, indicating the field that contains the value of the occurrence count. If InterFieldCount contains a FieldRef which does not exist in the transaction, then then InterFieldCount does not apply to the field in that particular instance.  Notes:   * InterFieldCount is NOT defined for the field that is referenced. * This may only be used when the referenced field is limited to a single occurrence. * This can apply to information items, e.g. in 10.015. | Optional | 9/NOM  (for 9.137, the number of occurrences is stored in 9.136 NOM) |
| InterFieldValue | For Fields where the permissible values are defined by the values of other fields, this is set to one or more comparative statements, of the form <operator><[FieldRef]>. Multiple statements are comma-separated, and imply a logical AND (all statements must be satisfied). Operators include "=", "<", ">","<=", ">=" and “!=”. If InterFieldValue contains a FieldRef which does not exist in the transaction, then then InterFieldValue does not apply to the field in that particular instance.  Notes:   * It is permissible to use negative values,   e.g. ">[-9/EHO]"   * This may only be used when the referenced field is limited to a single occurrence. * May be used in combination with ValueRange. When multiple attributes restrict values (e.g. ValueRange and InterFieldValue and CodeTable), the valid fields must satisfy ALL of the restrictions. For example a field with ValueRange of [0..49999] and InterFieldValue of <[9/ROI/EWI] must satisfy both. | Optional | <[9/ROI/EWI]  (For 9.331-A, which must be less than 9.300-A EWI)  <=[9/NOM];!=[9/RCI/CMI]  (for 9.138-2B) |
| InterFieldOther | Set to "Y" to indicate that the permissible values are restricted by the presence, values, or counts of other fields in ways that cannot be defined by InterFieldValue. A compliance checker must incorporate special logic for such fields. An explanation must be included in TechNotes. | Optional | Y  (for 9.126-B) |
| Exception | Set to "Y" or “true” to indicate that the values for this field are restricted in ways that cannot be completely and explicitly defined addressed through the use of other attributes; otherwise set to “N” or “false”. An explanation must be included in TechNotes. A compliance checker must incorporate special logic for such fields. | Optional | Y  (for 10.998-J) |
| Summary | Short summary of field/information item, appropriate for tooltip or other user interface display. Must be written to be appropriate for end user. Limited to alphanumeric, symbols, and space (no CRs or tabs).  The summary for both ANSI/NIST and EBTS consists of wording from the text descriptions and XML mapping documents. | Required for all traditional fields | An identification for an organization to which the transaction is being sent (for 1.007) |
| TechNotes | Technical free text explanation of specific requirements or dependencies of the field. Only used if they cannot otherwise be explicitly defined in this table.  This text is intended for technical information related to format or content, as opposed to Summary, for end users. Limited to alphanumeric, symbols, and space (no CRs or tabs). | Required if InterFieldOther="Y", DependOther="Y", or Exception="Y”; | UTM zone number ("0".."60") followed by band letter ("C".."H","J".."N","P".."X") (for 10.998-J) |
| Example | Sample value for that field |  | false  (for 9.004:X) |
| Changes\_to\_AN | Changes made to ANSI/NIST 2011 fields in subsequent ANSI/NIST versions (e.g., AN2011plus which lists corrections made to ANSI/NIST 2011 errors as described in comments <2013e> in ANSI/NIST 2013). Applies to ANSI/NIST only and not application profiles. | Optional | Type changed from AN to ANS; added SpecialChar and NonPrintChar (for 1.009 in AN2013e-FieldDef) |
| Notes | Additional annotations | Optional |  |
| OpenIssues | Unresolved issues | Optional |  |

# LookupCodes

The *LookupCodes* MRT defines the legal values for those fields which specify a value in the CodeTable column of the *FieldDefinitions* MRT.

|  |  |  |  |
| --- | --- | --- | --- |
| Column Name | Column Definition | Required in MRT | Example |
| CodeTable | A set of valid values for a field that are grouped together by sharing a common CodeTable. If a field requires use of a code table, its value is correct if and only if that value is listed in its code table.  Notes:   * The code table names for application profiles start with the application profile abbreviation and a colon (e.g. FBIEBTS:PATCL) (to avoid conflicts, since an implementation may have more than one application profile in use at the same time). * Code table names without a colon are for the base ANSI/NIST 2011 standard. * Semicolons shall not be used in code table names. | Required | AMPCD |
| Code | The value saved in the field.  Notes:   * It is permissible for a single code to appear more than once in the table with a different description. In such cases, the code groups must be different and/or the PrimaryDescription is indicated. * It is permissible to have blank codes. | Required | XX  (code in code table AMPCD) |
| Description | Text displayed to the user which describes the code. Not saved in the transaction. | Required | Partial print due to amputation  (description for code XX) |
| CodeGroup | For long lists, can assist in selection of values. Not saved in the ANSI/NIST transaction. | Optional | Tattoo  (for SMT code table) |
| PrimaryDescription | If (and only if) one Code is listed twice in the same CodeTable, Primary may be set to "Y" to indicate the primary Description and/or CodeGroup that should be used.  Note: One Code may be listed twice in the same CodeTable to aid in usability in selecting the correct code, either by having it listed with different descriptions (e.g. NF=Newfoundland (includes Labrador); NF=Labrador (included in Newfoundland)), or to include one code in more than one CodeGroup. | Optional | Y  (for code SPORT in SMD/TSC code table) |
| OpenIssues | Unresolved issues | Optional |  |
| Notes | Additional annotations | Optional |  |

# TOT Records

The *TOTrecords* MRT defines columns for a general-purpose table-based definition for transactions' record set requirements, as defined for application profiles derived from ANSI/NIST-ITL (e.g. EBTS, INT-I). Transactions are not defined for the base ANSI/NIST-ITL standards themselves. Each row represents one type of transaction. The EBTS MRTs are equivalent to the “Record Set Requirements Summary by Type of Transaction” and “Record Set Requirements Summary by Type of Response” tables in FBI EBTS Appendix L.

|  |  |  |  |
| --- | --- | --- | --- |
| Column Name | Column Definition | Required in MRT | Example |
| TransactionFullName | Full name for type of transaction | Required | AMNESIA VICTIM |
| TOT | Abbreviation for Type of Transaction | Required | AMN |
| TOT\_Description | Provides a brief overview and describes the purpose of the Type of Transaction | Required | The Biometric Delete Response (BDELR) is returned when a successful Biometric Delete Request is completed. If any permission or processing errors are encountered, an error transaction (ERRA) is returned.  (for BDELR TOT) |
| RecordNum | Record Type Number in application profile. | Required | 2  (equals Record Type-2) |
| Record\_Min | Minimum number of records. Records not included in the transaction are not listed in TOTrecords. | Required | 1  (for Type-1) |
| Record\_Max | Maximum number of records. Records not included in the transaction are not listed in TOTrecords. | Required | 14  (for AMN, Type-4) |
| Notes | Additional annotations | Optional |  |

# TOTfields

The *TOTfields* MRT defines columns for a general-purpose table-based definition for transactions' record set requirements, as defined for application profiles derived from ANSI/NIST-ITL (e.g. EBTS, INT-I). In table format, each row represents a field and TOT combination. Transactions are not defined for the base ANSI/NIST-ITL standards themselves and the EBTS TOTfields MRT is equivalent to Tables D-1, E-1, and I-1 in FBI EBTS.

|  |  |  |  |
| --- | --- | --- | --- |
| Column Name | Column Definition | Required in MRT | Example |
| RecordNum | Record Type Number in application profile. | Required | 2  (equals Record Type-2) |
| FieldNum | Record Field number | Required | 12  (equals field 12 within RecordNum) |
| Mnemonic | Three-letter Field Identifier | Required | LCN  (mnemonic for field 2.012) |
| TOT | Abbreviation for Type of Transaction | Required | AMN |
| Max | Max number of field occurrences | Required | 9  (for field 2.007 of AMN TOT and means that up to 9 occurrences of field 2.007 are allowed for an AMN transaction) |
| Req | Set to "Y" if field is required for a TOT; otherwise omitted.  If a field is optional in ANSI/NIST, it can be made required in the application profile - the reverse is not true. | Optional | Y  (for field 2.018 of AMN TOT) |
| Exc | Indicates additional field requirements for fields which are transaction specific and not covered in this table.   * In EBTS, one way the Exc (exception) columns are used is if a table value contains a superscript pointing to an endnote. Endnotes are described in the TOTFieldDetail MRT. * An entry is included in the TOTfieldDetail MRT if and only if TOT-Exc is set to "Y". | Optional | Y  (for field 2.018 of AMN TOT) |
| Notes | Additional annotations | Optional |  |

# TOTFieldDetail

The *TOTFieldDetail* MRT defines columns for transaction-specific field requirements, as defined for application profiles derived from ANSI/NIST-ITL (e.g. EBTS, INT-I). Transactions are not defined for the base ANSI/NIST-ITL standards themselves. In a table format, each row represents a combination of one TOT and one field. An entry in this table is only specified if the TOTfields MRT has a "Y" or “true” in the -Exc (exception) column. In EBTS, most information in this MRT is taken from table endnotes (e.g., Appendix D –Reference Notes in EBTS v10.0).

|  |  |  |  |
| --- | --- | --- | --- |
| Column Name | Column Definition | Required in MRT | Example |
| TradXML | States whether the field/row is formatted as XML only (“X”), Traditional only (“T”), or both XML and Traditional (“TX”). | Required | TX  (for RPIS, Field 2.084)  T  (for CNA, Field 2.005) |
| TOT | Abbreviation for Type of Transaction | Required | CAR |
| RecordNum | Record Type Number in application profile. | Required | 2  (equals Record Type-2) |
| FieldNum | Record Field number | Required | 14  (equals field 14 within RecordNum) |
| Mnemonic | Three-letter Field Identifier | Required | FBI  (mnemonic for CAR and field 2.014) |
| Notes | Text notes on additional field details and/or requirements (e.g., FBI EBTS Appendix D endnotes) | Required | App D Note 5: FBI Number must be present if known for inquiry prints.  (notes for CAR and field 2.014) |
| Data Type | See description in FieldDefinition | Optional | AN  (for 2.2029 field and BDEL TOT) |
| ValueRange | See description in FieldDefinition | Optional | [“Y”] |
| RegExpression | See description in FieldDefinition | Optional | (M[0-9A-F]{12})|([P]\S{12,15}) (for 9.903 DUI: M followed by 12 hexadecimal characters, or P followed by 12 to 15 non-whitespace characters) |
| FieldReq | States a field which is mandatory if the field and TOT combination described in under TOT, RecNum, FieldNum, and Mnemonic is present. |  | 2/CSL  (From App D Note 9: CSL must be included where submission includes SLE) |
| DependPresenceReq | See description in FieldDefinition | Optional | 14/DATA  (for 14.010 TVPS, which is mandatory if an image is present in 14.999 DATA) |
| DependPresenceOpt | See description in FieldDefinition | Optional | 18/DGD  (Field 18.021 is optional and only permitted if field 18/DGD, 18.020, is present) |
| DependAbsence | See description in FieldDefinition | Optional | 9/MIN  (for 9.334 NMIN, which is not included if 9.331 MIN is used) |
| DependValueReq | See description in FieldDefinition | Optional | 2/RET="Y"  (from App D Note 2) |
| InterFieldCount | See description in FieldDefinition | Optional | 10/FEC/NOP  (for HPO 10/FEC/HPO:T, the number of occurrences of 2.074 FGP and 2.064 CAN must be the same) |
| InterFieldValue | See description in FieldDefinition | Optional | <[9/ROI/EWI]  (For 9.331A MXC, which must be less than 9.300A EWI) |

# RecordTypeDesc

The *RecordTypeDesc* MRT defines columns for a general-purpose table-based definition for Record Types, as defined in application profiles and ANSI/NIST-ITL. The full name and summary for each record are included.

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Column Definition** | **Required in MRT** | **Example** |
| RecordType | Numeric record representation | Required | 1  (for Record Type 1) |
| RecordName | Full record type name | Required | Transaction Information Record  (for Record Type 1) |
| RecordDescription | Summary of record type. | Required | The Type-1 record provides information describing the type and use or purpose for the transaction involved, a listing of each record included in the transaction, the originator or source of the physical record, and other useful and required information items. Record Type-1 is mandatory…  (sample text for Record Type 1) |
| Notes | Additional annotations | Optional |  |

# Appendix A: MRTVerify Error Codes

Numbers in braces (e.g. {0}) indicate parts of the error message that are filled in by MRTVerify when the error is reported. Use this table as reference when looking at KnownErrors.xml files.

| **Error Code** | **Format String** | **Column(s)** | **Description** |
| --- | --- | --- | --- |
| 0 | Warning/Error/Fatal Error at file {0}, line {1}, char {2}  Message: {3} | N/A | DOMError. Error reported by the Xerces-C DOM xml reader. |
| 1 | Warning/Error/Fatal Error at file {0}, line {1}, char {2}  Message: {3} | N/A | SAXError. Error reported by the Xerces-C SAX2 xml reader. |
| 2 | Warning/Error/Fatal Error at file {0}, line {1}, char {2}  Message: {3} | N/A | MRT specific read error. Can occur when two or more rows in the table have the same key or when a schema has an invalid id. |
| 3 | Schema {0} is not recognized. | N/A | Unknown schema id. |
| 4 | FieldRef suffix and FieldID suffix do not match | FieldID | Example: 1/LEN:T and 01.001:X, “:T” != “:X” |
| 5 | FieldID is not unique and is also used for FieldRef '{0}' | FieldID | FieldIDs must be unique in a table. |
| 6 | Row does not have FieldNum or XMLPath defined | FieldID | If the FieldID does not have a field number, it is an XML only field and should then have an XMLPath defined. |
| 7 | InfoItem must be blank if FieldNum is blank | InfoItem |  |
| 8 | InfoItem does not match info item specifier in FieldID column | InfoItem | Example: InfoItem is “B” but FieldID is “01.003-A”, “A” != “B” |
| 9 | InfoItemCount must match the number of info items when InfoItem is 'SET' | InfoItemCount |  |
| 10 | If a CondCode contains a caret, it can only be used for info items. | CondCode |  |
| 11 | If there exists a value in InfoItem, there must be a value in DataType unless the InfoItem = 'SET' or ContentType = 'Set', 'Set\_X', or 'Set\_T' | DataType |  |
| 12 | If there exists a value in InfoItem, there must be a value in MinLength unless the InfoItem = 'SET' or ContentType = 'Set', 'Set\_X', or 'Set\_T' | MinLength |  |
| 13 | If there exists a value in InfoItem, there must be a value in MaxLength unless the InfoItem = 'SET' or ContentType = 'Set', 'Set\_X', or 'Set\_T' | MaxLength |  |
| 14 | Mnemonics are required for traditional field rows | Mnemonic |  |
| 15 | Traditional mnemonics must be unique for field rows within a record. '{0}' is already defined for row '{1}'. | Mnemonic |  |
| 16 | Traditional mnemonics must be unique for info item rows within a field. '{0}' is already defined for row '{1}'. | Mnemonic |  |
| 17 | SpecialChar should only include 'special characters' i.e. printable characters that aren't letters or numbers (ascill codes 33 thru 126) | SpecialChar |  |
| 18 | SpecialChar or NonPrintChar required when DataType has an 'S' | SpecialChar, NonPrintChar |  |
| 19 | SpecialChar and NonPrintChar should be omitted if DataType has no 'S' | SpecialChar, NonPrintChar |  |
| 20 | XMLPath required for XML fields (TradXML = X or TX) | XMLPath |  |
| 21 | XMLPath should be omitted for traditional fields (TradXML = T) | XMLPath |  |
| 22 | XML path '{0}' does not exist in a transaction file | XMLPath | This is checked using data generated from the official xml schemas for a particular application profile. |
| 23 | XML path '{0}' does not have a 'set' content type and should be a leaf node | XMLPath |  |
| 24 | Code Table '{0}' does not exist | CodeTable, XMLPath |  |
| 25 | CodeTableOpt should be omitted if there are no Code Tables specified | CodeTableOpt |  |
| 26 | Regular expression compilation error: {0} | RegExpression |  |
| 27 | If there exists an InfoItem that is not equal to 'SET', then there should be a CondCode | CondCode |  |
| 28 | If CondCode does not have a caret there should not be an InfoItem or it should be 'SET' | CondCode |  |
| 29 | Primary description should be set to true for one occurrence of code '{0}' in code table '{1}' | PrimaryDescription |  |
| 30 | Code '{0}' occurs more than twice in code table '{1}' | Code |  |
| 31 | TOT '{0}' does not exist in the FBIEBTS:TOT code table | TOT |  |
| 32 | Record.field combination '{0}.{1}' does not exist in the field definition table | RecordNum, FieldNum |  |
| 33 | Mnemonic '{0}' does not match the mnemonic for record.field combination '{1}.{2}' in the field definition table | Mnemonic |  |
| 34 | For fields where CondCode = D, there needs to be a value in one of these fields: DependPresenceReq, DependPresenceOpt, DependAbsence, DependValue, DependValueReq, DependValueOpt, DependOther, or the Exception column must be true. | CondCode |  |
| 35 | For fields where CondCode = O or O^, MinOccur shall be 0 | CondCode |  |
| 36 | For fields where CondCode = M or M^, MinOccur shall be greater than or equal to 1 | CondCode |  |
| 37 | If FieldID is present, Description is required | Description |  |
| 38 | Record number '{0}' specified in column '{1}' does not exist in the Field Definition Table | TOTRecords columns with the format T\d+\_((Min)|(Max)) |  |
| 39 | Summary field is required for traditional fields, optional for XML fields | Summary |  |
| 40 | The string '{0}' has invalid syntax | XMLPath, ValueRange, DependValue, DependValueReq, DependValueOpt, InterFieldValue |  |
| 41 | The minimum must be less than or equal to the maximum in range {0}..{1} | XMLPath, ValueRange, DependValue, DependValueReq, DependValueOpt |  |
| 42 | FieldRef '{0}' does not exist | Any column that uses FieldRefs |  |
| 43 | Only {0} field reference(s) are allowed for this column, found {1} | InterFieldCount |  |
| 44 | Referenced field '{0}' cannot have InterFieldCount defined | InterFieldCount |  |
| 45 | Referenced field '{0}' must be limited to a single occurrence | InterFieldCount |  |
| 46 | If DependOther or InterFieldOther is set to true, TechNotes must be populated | TechNotes |  |
| 47 | Parser error at location ({0}), near '{1}': {2} | XMLPath, ValueRange, DependValue, DependValueReq, DependValueOpt, InterFieldValue |  |
| 48 | Value '{0}' must be less than '{1}' in double comparison. | DependValue, DependValueReq, DependValueOpy |  |
| 49 | Value '{0}' must be greater than '{1}' in double comparison. | DependValue, DependValueReq, DependValueOpy |  |
| 50 | Value '{0}' must be less than or equal to '{1}' in double comparison. | DependValue, DependValueReq, DependValueOpy |  |
| 51 | Value '{0}' must be greater than or equal to '{1}' in double comparison. | DependValue, DependValueReq, DependValueOpy |  |
| 52 | Field reference '{0}' has an invalid format | FieldRef |  |
| 53 | FieldID '{0}' has an invalid format | FieldID |  |
| 54 | XML path '{0}' is repeated. Also used for field '{1}'. | XMLPath |  |
| 55 | Record type {0} does not exist in the Field Definition Table. | RecordType |  |