



XML Naming and Design Rules

Draft 1.0, 3 August 2004

This *UN/CEFACT – XML Naming and Design Rules* has been developed in accordance with the UN/CEFACT/TRADE/22 Open Development Process (ODP) for Technical Specifications. It has been approved by the United Nations Centre for Trade Facilitation and Electronic Business (UN/CEFACT) Applied Techniques Group (ATG) for promulgation for public review in accordance with Step 5 of the ODP.

Distribution of this document is unlimited.

The document formatting is based on the Internet Society's Standard RFC format.

This version:

**UN/CEFACT – XML Naming and Design Rules for Core Components, Version 1.0 of 3.
August 2004**

Previous version:

NamingAndDesignRules_0p4r1.doc

Document identifier:

NamingAndDesignRules_1.0.doc

Location:

<http://www.disa.org/cefact-groups/atg/downloads/index.cfm>

UN/CEFACT – XML Naming and Design Rules Project Team Participants

We would like to recognise the following for their significant participation to the development of this Technical Specification.

Project Team Leader:

Mark Crawford LMI

Editors:

Gunther Stuhec SAP AG

Paula Heilig Worldspan

Margaret Pemberton Diskray

Contributors:

Hisanao Sugamata ECOM-Japan

Frank Lin GCOM

K.K. Suen EAN Hong Kong

Luc Mouchot CNAM-TS

Thomas Bikeev EAN.UCC

Jostein Frømyr EDISYS

Sue Probert SEPIA eb

Acknowledgements

The UN/CEFACT - XML Naming and Design Rules were developed in close coordination with other XML standards efforts. In particular, the OASIS Universal Business Language Technical Committee Naming and Design Rules were instrumental in developing this document. Additionally, contributions were also received from:

SWIFT
U.S. Department of the Navy
U.S. Environmental Protection Agency
U.S. Federal CIO Council XML Working Group
OpenTravel Alliance
Australia Electricity & Gas Industry
CIDX
EAN/UCC
European Transmission System Operators
PIDX

37 **Table of Contents**

38	1	INTRODUCTION	7
39	1.1	Scope and Focus.....	7
40	1.2	Audiences.....	7
41	1.3	Structure of this Specification	7
42	1.4	Terminology and Notation	7
43	1.5	Related Documents	8
44	1.6	Conformance	8
45	1.7	Guiding Principles	8
46	2	GENERAL XML CONSTRUCT	9
47	2.1	Overall Schema Structure.....	9
48	2.2	Relationship to the CCTS	9
49	2.3	Naming and Modeling Constraints	12
50	2.3.1	Naming Constraints	12
51	2.3.2	Modelling Constraints	14
52	2.4	Reusability Scheme.....	14
53	2.4.1	Element Naming Conventions	14
54	2.5	Modularity Model.....	15
55	2.5.1	Root Schema.....	15
56	2.5.2	Internal Schema	15
57	2.5.3	External Schema	16
58	2.5.3.1	Core Component Type Schema Module.....	16
59	2.5.3.2	Unqualified Data Type Schema Module.....	17
60	2.5.3.3	Qualified Data Type Schema Module.....	17
61	2.5.3.4	Reusable Aggregate Business Information Entity Schema Module	17
62	2.5.3.5	Code List Schema Modules	17
63	2.5.3.6	Identifier List Schema Modules.....	18
64	2.5.3.7	External Standards Body Aggregate Business Information Entity Schema Modules.....	18
65	2.6	Namespace Scheme.....	18
66	2.6.1	UN/CEFACT Namespace Scheme	19
67	2.6.2	Declaring Namespace	20
68	2.6.3	Namespace Persistence	21
69	2.6.4	Namespace Uniform Resource Identifiers	21
70	2.6.5	Namespace Constraint	22
71	2.6.6	UN/CEFACT Schema Namespace Tokens.....	22
72	2.7	Schema Location.....	22
73	2.8	Versioning.....	23

74	2.8.1	Major Versions	23
75	2.8.2	Minor Versions	24
76	3	GENERAL XML SCHEMA LANGUAGE CONVENTIONS	26
77	3.1	Schema Construct	26
78	3.1.1	Constraints on Schema Construction	26
79	3.2	Attribute and Element Declarations	27
80	3.2.1	Attributes	27
81	3.2.1.1	Usage of Attributes	27
82	3.2.1.2	Constraints on Attribute Declarations	27
83	3.2.2	Elements	27
84	3.2.2.1	Usage of Elements	27
85	3.2.2.2	Element Declaration	27
86	3.2.2.3	Constraints on Element Declarations	28
87	3.3	Type Definitions	28
88	3.3.1	Usage of Types	28
89	3.3.2	Simple Type Definitions	28
90	3.3.3	Complex Type Definitions	29
91	3.4	Use of XSD Extension and Restriction	29
92	3.4.1	XSD Extension	29
93	3.4.2	Restriction	29
94	3.5	Annotation	30
95	3.5.1	Documentation	30
96	4	XML SCHEMA MODULES	32
97	4.1	Root Schema	32
98	4.1.1	Schema Construct	32
99	4.1.2	Namespace Scheme	32
100	4.1.3	Imports and Includes	33
101	4.1.4	Root Element Declaration	33
102	4.1.5	Type Definitions	34
103	4.1.6	Annotations	34
104	4.2	Internal Schema	35
105	4.2.1	Schema Construct	35
106	4.2.2	Namespace Scheme	35
107	4.2.3	Imports and Includes	35
108	4.3	Reusable Aggregate Business Information Entities	35
109	4.3.1	Schema Construct	35
110	4.3.2	Namespace Scheme	36
111	4.3.3	Imports and Includes	36
112	4.3.4	Type Definitions	37
113	4.3.5	Element Declarations	39
114	4.3.6	Annotation	40
115	4.4	Core Component Type	44
116	4.4.1	Use of Core Component Type Module	44
117	4.4.2	Schema Construct	44
118	4.4.3	Namespace Scheme	45
119	4.4.4	Imports and Includes	45
120	4.4.5	Type Definitions	45
121	4.4.6	Attribute Declarations	46

122	4.4.7	Extension and Restriction	46
123	4.4.8	Annotation	46
124	4.5	Unqualified Data Type	48
125	4.5.1	Use of Unqualified Data Type Module.....	48
126	4.5.2	Schema Construct	48
127	4.5.3	Namespace Scheme	49
128	4.5.4	Imports and Includes.....	49
129	4.5.5	Type Definitions	49
130	4.5.6	Attribute Declarations.....	51
131	4.5.7	Restriction.....	51
132	4.5.8	Annotation	51
133	4.6	Qualified Data Type	53
134	4.6.1	Use of Qualified Data Type Module.....	53
135	4.6.2	Schema Construct	54
136	4.6.3	Namespace Scheme	54
137	4.6.4	Imports and Includes.....	55
138	4.6.5	Type Definitions	55
139	4.6.6	Attribute and Element Declarations.....	56
140	4.6.7	Extension and Restriction	56
141	4.6.8	Annotation	57
142	4.7	Code Lists	59
143	4.7.1	Schema Construct	59
144	4.7.2	Namespace Scheme	60
145	4.7.3	Imports and Includes.....	61
146	4.7.4	Type Definitions	61
147	4.7.5	Element Declarations.....	62
148	4.7.6	Extension and Restriction	62
149	4.7.7	Annotation	63
150	4.8	Identifier Schemas	64
151	4.8.1	Schema Construct	64
152	4.8.2	Namespace Scheme	65
153	4.8.3	Imports and Includes.....	65
154	4.8.4	Type Definitions	65
155	4.8.5	Attribute and Element Declarations.....	66
156	4.8.6	Extension and Restriction	66
157	4.8.7	Annotation	67
158	5	XML INSTANCE DOCUMENTS	68
159	5.1	Character Encoding.....	68
160	5.2	Empty Content	68
161	5.3	xsi:type.....	68
162		APPENDIX A. OVERALL STRUCTURE	69
163	1.	XML DECLARATION	69
164	2.	SCHEMA MODULE IDENTIFICATION AND COPYRIGHT INFORMATION.....	69
165	3.	SCHEMA START-TAG	70

166	4. INCLUDES.....	71
167	5. IMPORTS.....	72
168	6. ROOT ELEMENT	73
169	7. TYPE DEFINITIONS.....	74
170	APPENDIX B. ANNOTATION TEMPLATES.....	78
171	APPENDIX C. NAMING & DESIGN RULES LIST	81
172	APPENDIX D. DEFINITION OF TERMS	93
173	APPENDIX E. RELATED DOCUMENTS	96
174	APPENDIX F. CORE COMPONENT SCHEMA MODULE.....	98
175	APPENDIX G. UNQUALIFIED DATA TYPE SCHEMA MODULE.....	110
176		

177 **1 Introduction**

178 This *UN/CEFACT – XML Naming and Design Rules* describes and specifies the rules and guidelines that
179 will be applied by UN/CEFACT when developing XML schema specifications.

180 This *UN/CEFACT – XML Naming and Design Rules* provides a way to identify, capture and maximize the
181 re-use of business information expressed as XML schema components to support and enhance
182 information interoperability across multiple business situations.

183 **1.1 Scope and Focus**

184 This *UN/CEFACT – XML Naming and Design Rules* can be employed wherever business information is
185 being shared or exchanged amongst and between enterprises, governmental agencies, and/or other
186 organisations in an open and worldwide environment using the XML schema language for defining the
187 content of the information exchange.

188 This specification will form the basis for standards development work of technical experts developing XML
189 schema specifications based on information models developed in accordance with the *UN/CEFACT –*
190 *Core Components Technical Specification*.

191 **1.2 Audiences**

192 The primary audience for this *UN/CEFACT – XML Naming and Design Rules* are member of the
193 UN/CEFACT Applied Technologies Group, working group on XML who are responsible for development
194 and maintenance of the UN/CEFACT XML schema specifications and the wider membership of the other
195 UN/CEFACT Groups who will participate in the process of creating and maintaining the UN/CEFACT XML
196 schema specifications.

197 Additional audiences are designers of tools who need to specify the conversion of user input in the tool
198 into the XML schema representation adhering to the roles defined in this document as well as designers
199 of XML schema specifications outside of the UN/CEFACT Forum community.

200 **1.3 Structure of this Specification**

201 These *UN/CEFACT – XML Naming and Design Rules* have been divided in to 5 main sections.

202 Section 1 provides general information about the document itself.

203 Section 2 provides information on the guiding principles applied in developing this specification as well as
204 its dependency and relationship to the *UN/CEFACT – Core Components Technical Specification*.

205 Furthermore this section describes the approach taken to modularity in order to maximize the re-use of
206 business information expressed as XML schema components and the general naming conventions
207 applied.

208 Section 3 provides the general conventions applied with respect to the use of the XML schema language.

209 Section 4 provides detailed rules applicable to each of the schema modules defined by the modularity
210 approach.

211 Section 5 provides guidelines and rules related to XML instance documents.

212 **1.4 Terminology and Notation**

213 The keywords MUST, MUST NOT, REQUIRED, SHALL, SHALL NOT, SHOULD, SHOULD NOT,
214 RECOMMENDED, MAY, and OPTIONAL, when they appear in this document, are to be interpreted as
215 described in Internet Engineering Task Force (IETF) Request For Comments (RFC) 2119.1
216

217 [Definition] – A formal definition of a term. Definitions are normative.

218 [Example] – A representation of a definition or a rule. Examples are informative.

Key words for use in RFCs to Indicate Requirement Levels - Internet Engineering Task Force, Request
For Comments 2119, March 1997,

<http://www.ietf.org/rfc/rfc2119.txt?number=2119>

219 [Note] – Explanatory information. Notes are informative.
220 [Rn] – Identification of a rule that requires conformance. Rules are normative. In order to ensure
221 continuity across versions of the specification, rule numbers that are deleted will not be re-issued, and
222 any new rules will be assigned the next higher number - regardless of location in the text.
223 **Courier** – All words appearing in **bolded courier font** are values or objects or keywords.
224 [] = optional
225 < > = Variable
226

227 1.5 Related Documents

228 The documents referenced in this document are listed in informative Appendix E.

229 1.6 Conformance

230 Applications will be considered to be in full conformance with this technical specification if they comply
231 with the content of normative sections, rules and definitions.

232 [R 1] Conformance shall be determined through adherence to the content of normative sections, rules
233 and definitions.

234 1.7 Guiding Principles

235 The following guiding principles were used as the basis for all design rules contained in this document:

- 236 • Relationship to UMM – UN/CEFACT XML Schemas will be based on UMM metamodel adherant
237 Business Process Models.
- 238 • Relationship to Information Models – UN/CEFACT XML Schemas will be based on UML-based
239 Information Models.
- 240 • Schema Creation– UN/CEFACT XML design rules will support schema creation through handcrafting
241 as well as automatic generation from UML profiled objects and XML profiled objects.
- 242 • ebXML Use – UN/CEFACT XML Schemas and instance documents shall be straightforwardly usable
243 within the ebXML framework and compatible with other frameworks to the maximum extent
244 practicable.
- 245 • Interchange and Application Use – UN/CEFACT XML Schemas and instance documents are intended
246 for business-to-business and application-to-application use.
- 247 • Tool Use and Support - The design of UN/CEFACT XML Schemas will not make any assumptions
248 about sophisticated tools for creation, management, storage, or presentation being available.
- 249 • Time Constraints–Urgency is a key item in the development of UN/CEFACT XML design rules.
- 250 • Legibility - UN/CEFACT XML instance documents should be intuitive and reasonably clear in the
251 context for which they are designed.
- 252 • Schema Features - The design of UN/CEFACT XML Schemas should use the most commonly
253 supported features of W3C XSD Schema.
- 254 • Technical Specifications – UN/CEFACT XML design rules will be based on Technical Specifications
255 holding the equivalent of W3C recommended status.
- 256 • Schema Specification – UN/CEFACT XML design rules will be fully conformant with W3C XML
257 Schema Definition Language.
- 258 • Interoperability - The number of ways to express the same information in a UN/CEFACT XML Schema
259 and UN/CEFACT XML instance document is to be kept as close to one as possible.
- 260 • Customization - The design of UN/CEFACT XML Schemas must facilitate customization.
- 261 • Maintenance – The design of UN/CEFACT XML Schemas must facilitate maintenance.
- 262 • Context Sensitivity - The design of UN/CEFACT XML Schemas must ensure that context-sensitive
263 document types aren't precluded.
- 264 • Relationship to Other Namespaces - UN/CEFACT XML design rules will be cautious about making
265 dependencies on other namespaces.
- 266 • Legacy formats - UN/CEFACT XML design rules are not responsible for sustaining legacy formats.
- 267 • Messages must express semantics fully in schemas and not rely on well-formedness.

268 2 General XML Construct

269 This section defines rules related to general XML constructs to include:

- 270 • Overall Schema Structure
- 271 • Relationship to CCTS
- 272 • Naming and Modeling Constraints
- 273 • Reusability Scheme
- 274 • Modularity Strategy
- 275 • Namespace Scheme
- 276 • Versioning Scheme
- 277 • Schema Documentation Requirements

278 2.1 Overall Schema Structure

279 To maintain consistency in lexical form, all UN/CEFACT schema need to use a standard structure for all
280 content. This standard structure is contained in Appendix A.

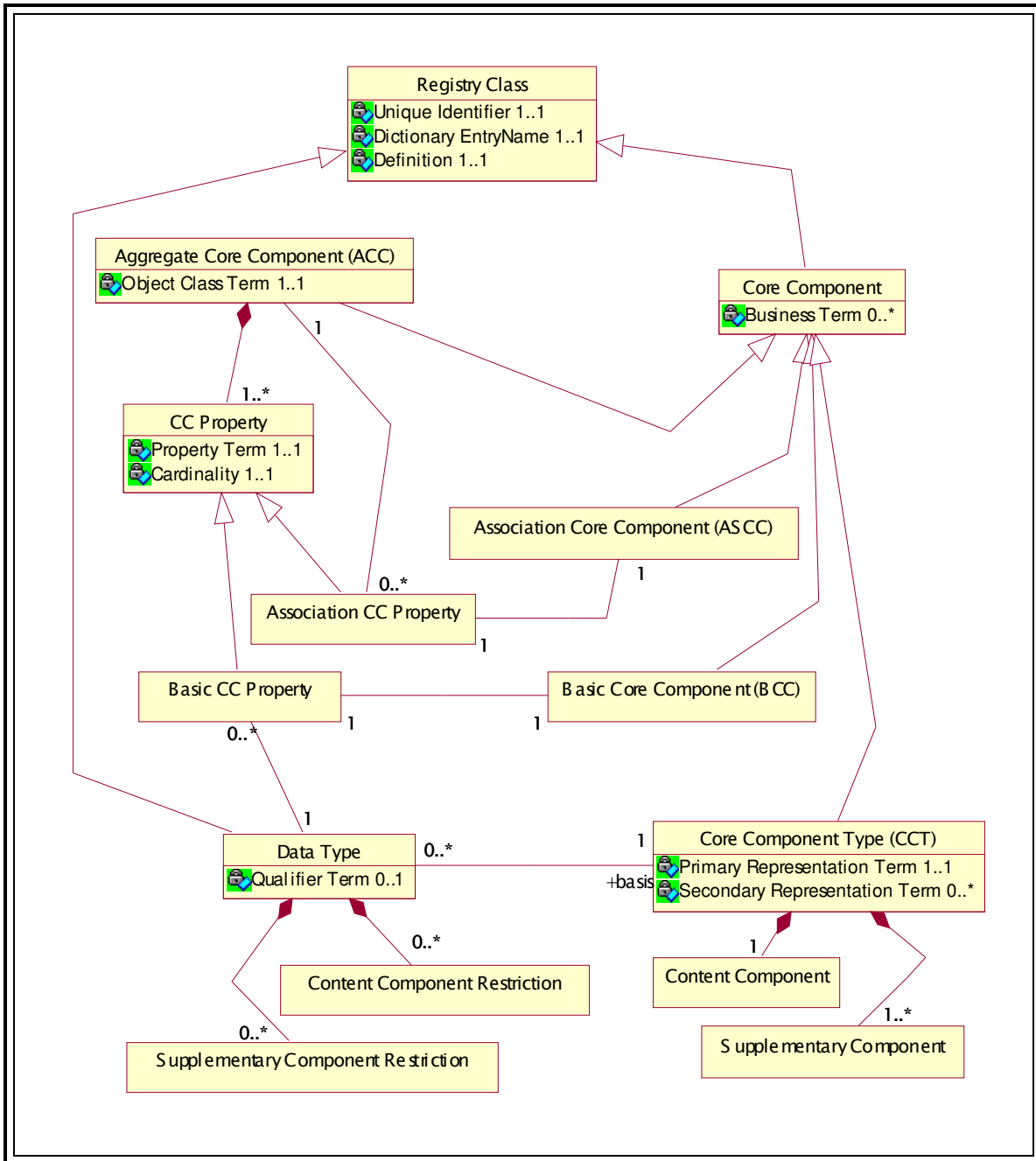
281 [R 2] UN/CEFACT Schema MUST follow the standard structure defined in Appendix A.

282 2.2 Relationship to the CCTS

283 All UN/CEFACT business modelling and business process definition employs the methodology and model
284 described in *Core Components Technical Specification, Part 8 of the ebXML Technical Framework*
285 (CCTS). The CCTS is a continuation of work that originated in, and remains a part of, the ebXML
286 initiative. CCTS defines a new paradigm in data definition and use.

287 CCTS defines context neutral and context specific building blocks. Context neutral components are
288 defined as Core Components (**ccts:CoreComponents**). Context neutral **ccts:CoreComponents** are
289 defined in CCTS as “A building block for the creation of a semantically correct and meaningful information
290 exchange package. It contains only the information pieces necessary to describe a specific concept.”²
291 Figure 2-1 illustrates the various pieces of the overall **ccts:CoreComponents** metamodel.

² Core Components Technical Specification, Part 8 of the ebXML Technical Framework Version 2.0 (Second Edition), UN/CEFACT, 15 November 2003



292

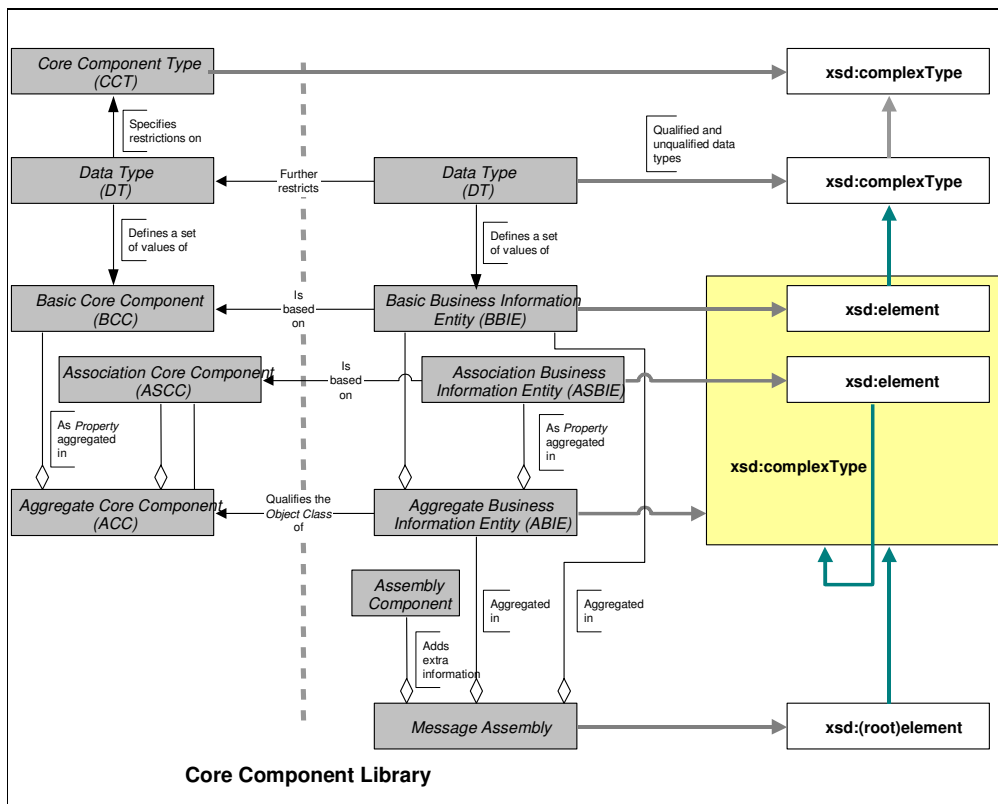
293 **Figure 2-1 Core Component MetaModel**

294

295 The context specific components are defined as Business Information Entities
 296 (**ccts:BusinessInformationEntities**).³ Context specific **ccts:Business**
 297 **InformationEntities** are defined in CCTS as “A piece of business data or a group of pieces of
 298 business data with a unique *Business Semantic* definition.”⁴ Figure 2-2 illustrates the various pieces of

³ See CCTS Section 6.2 for a detailed discussion of the ebXML context mechanism.

⁴ Core Components Technical Specification, Part 8 of the ebXML Technical Framework Version 2.0 (Second Edition), UN/CEFACT, 15 November 2003



311

312

Figure 2-3 Relationship between CCTS and XSD Artifacts in UN/CEFACT Schema

313

- The message assembly is represented as a complex type designated as the root element of the XML message.

314

- An ABIE is defined as a complex type.

315

- An ASBIE is declared as a local element within the complex type representing the associated ABIE. The ASBIE element is in itself based on (is of type) complex type of the associated ABIE.

316

- A BBIE is declared as a local element within the complex type representing the parent ABIE. The BBIE is based on a (is of type) DT.

317

- A DT is declared as a complex type or simple type. `xsd:built-in` data types will be used whenever the facets of the built in data type are equivalent to the built in supplementary components for that data type.

318

- A CCT is represented as a complex type. Supplementary components are declared as attributes for the CCT type.

319

320

321

322

323

324

325

2.3 Naming and Modeling Constraints

326

UN/CEFACT XML is derived from CCTS and UN/CEFACT Modelling Methodology (UMM) process modelling and data analysis as precursors to developing the UN/CEFACT XML component library. In

327

determining how best to affect this work, several constraints have been identified that directly impact on both the process modelling and data analysis, as well as on the resultant UN/CEFACT XML Schema.

328

329

330

2.3.1 Naming Constraints

331

The UN/CEFACT library contains fully conformant CCTS dictionary entry names as well as truncated XML element names developed in conformance with the naming constraint rules specified below. The XML fully qualified XPath ties the information to its standardized semantics as described in the underlying

332

CCTS construct and CCTS Dictionary Entry Name, while the XML element or attribute name is a truncation that reflects the hierarchy inherent in the XML construct. There are difference in the rules for

333

naming of elements, attributes, and types.

334

335

336

337

[R 3] Each element or attribute XML name MUST have *one* and only *one* fully qualified XPath (FQXP).

338
339 This rule and the other rules on element naming imply that the fully qualified Xpath will always represent
340 the CCTS dictionary entry name of the corresponding ABIE, BBIE, ASBIE or DT.

341 **Example 2-1:**

342 Address/Coordinate/Latitude Measure
343 Organisation/Location/Name

344

345 [R 4] Element, attribute and type names MUST be in the English language, using the primary English
346 spellings provided in the *Oxford English Dictionary*.

347 The official language for UN/CEFACT XML is English. All official XML constructs as published by
348 UN/CEFACT will be in English. XML development work may very well occur in other languages, however
349 official submissions for inclusion in the UN/CEFACT XML library must be in English. Other language
350 translations of UN/CEFACT published XML components are at the discretion of users. Strict literal
351 translations for implementation and run-time use are considered fully compliant with the published XML
352 component.

353 [R 5] Lower-camel-case (LCC) MUST be used for naming attributes.

354 **Example 2-2: Attribute**

355 `<xsd:attribute name="unitCode" .../>`

356

357 [R 6] Upper-camel-case (UCC) MUST be used for naming elements and types.

358 **Example 2-3: Element**

359 `<xsd:element name="LanguageCode" ...>`

360 **Example 2-4: Type**

361 `<xsd:complexType name="DespatchAdviceCodeType">`

362

363 [R 7] Names MUST be in singular form unless the concept itself is plural.

364 **Example 2-5: Singular and Plural Concept Form**

365 **Singular:**

366 `<xsd:element name="ItemQuantity" ...>`

367 **Plural:**

368 `<xsd:element name="GoodsQuantity" ...>`

369

370 [R 8] Names MUST NOT contain non-letter characters, unless required by language-specific rules.

371 **Example 2-6: Non-Letter Characters**

372 **Not Allowed**

373 `<xsd:element name="LanguageCode8" ...>`

374

375 [R 9] XML names constructed from dictionary entry names MUST NOT include periods, spaces, or other
376 separators; or characters not allowed by W3C XML 1.0 for XML names.

377 **Not Allowed**

378 `<xsd:element name="Customized_ Language. Code:8" ...>`

379

380 [R 10] Element and `xsd:simple` and `xsd:complexType` names MUST NOT use acronyms,
381 abbreviations, or other word truncations, except those included in the UN/CEFACT controlled
382 vocabulary.

383 2.3.2 Modelling Constraints

384 UN/CEFACT has developed a robust modelling methodology (UMM). This modelling methodology forms
385 the basis for all UN/CEFACT business process analysis conducted by, and process models developed
386 by, UN/CEFACT TBG. In keeping with our guiding principles, modelling constraints are limited to those
387 necessary to ensure consistency in development.

388 2.4 Reusability Scheme

389 UN/CEFACT is committed to transitioning to an object based approach for its process models and core
390 components implementation efforts as supported in both UMM and CCTS. UN/CEFACT deliberated
391 adopting a type based approach (named types), a type and element based approach, and an element
392 based approach. An `xsd:type` based approach for XML management provides the closest alignment
393 with the process modelling methodology described in UMM. Type information is beginning to be
394 accessible when processing XML instance documents. Post schema-validation infoset (PSVI)
395 capabilities are beginning to emerge that support this approach, such as “data-binding” software that
396 compiles schema into ready-to-use object classes and is capable of manipulating XML data based on
397 their types. The most significant drawback to a type based approach is the risk of developing an
398 inconsistent element vocabulary where elements are declared locally and allowed to be reused without
399 regard to semantic clarity and consistency across types. UN/CEFACT manages this risk by carefully
400 controlling the creation of BBIEs and ASBIEs with fully defined semantic clarity that are only usable within
401 the ABIE in which they appear. This is accomplished through the relationship between BBIEs, ASBIEs
402 and their parent ABIE and the strict controls put in place for harmonization and approval of the semantic
403 constructs prior to their XSD instantiation.

404 [R 11] All element declarations for BBIEs and ASBIEs MUST be locally declared within the parent ABIE
405 type.

406 2.4.1 Element Naming Conventions

407 The fully qualified path anchors the use of that construct to a particular location in a business message.
408 The dictionary definition identifies any semantic dependencies that the FQXP has on other elements and
409 attributes within the UN/CEFACT library that are not otherwise enforced or made explicit in its structural
410 definition. The dictionary serves as a traditional data dictionary, and also serves *some* of the functions of
411 traditional implementation guides. As discussed in Section 2.4 above, the dictionary must be carefully
412 controlled to overcome the limitations in control inherent in a local element approach.

413 [R 12] Each element name declaration MUST be based on the property term and qualifiers and the
414 representation term of the BBIE or the property term and object class of the ASBIE. If there are
415 successive duplicate words in the property term and representation terms of the source dictionary
416 entry name which represent exactly the same semantics, then the duplicate words MUST be
417 removed.

418 2.5 Modularity Model

419 Modularity in schema design promotes reuse and provides significant management capabilities. Modules
420 can be either unique in their functionality, or represent splitting of larger schema files for performance or
421 manageability enhancement. A modularity model provides an efficient and effective mechanism for
422 importing components as needed rather than dealing with complex, multi-focused schema.

423 2.5.1 Root Schema

424 UN/CEFACT incorporates a modularity concept that leverages the benefits previously described. In the
425 UN/CEFACT XML repository, there are a number of UN/CEFACT root schemas, each of which expresses
426 a separate business function. Each root schema will consist of a single named complex type that defines
427 associations to xsd constructs contained in internal (same namespace) or external (different namespace)
428 schema modules.

429 [R 13] A root schema **MUST** be created for each unique business information exchange.

430
431 The UN/CEFACT modularity approach enables the reuse of individual root schemas without having to
432 import the entire UN/CEFACT root schema library. Additionally, a root schema can import individual
433 modules without having to import all UN/CEFACT schema modules. Each root schema will define its own
434 dependencies. A root schema should not duplicate reusable xml constructs contained in other schema,
435 rather it should reuse existing constructs available elsewhere. Specifically, root schema will import or
436 include other schema modules to maximize reuse through `xsd:include` or `xsd:import` as appropriate.

437 [R 14] A root schema **MUST NOT** replicate reusable constructs available in schema modules capable of
438 being referenced through `xsd:include` or `xsd:import`.

439 Schema modules used by the root schema need to be treated as either internal or external schema
440 modules so correct namespace decisions can be made.

441 [R 15] UN/CEFACT schema modules **MUST** either be treated as external schema modules or as internal
442 schema modules of the root schema.

443 2.5.2 Internal Schema

444 For larger namespaces, schema modules – internal schema modules – may be defined. UN/CEFACT
445 schemas may have zero or more internal modules that they include. Root schemas are limited to defining
446 a single complex type that fully describes the business information exchange.

447 [R 16] Root schemas **MUST** only define a single complex type that fully describes the business
448 information exchange.

449 If additional complex types are needed that don't properly belong in the reusable ABIE module (See
450 Section 2.5.5 below), then those complex types will be defined in an internal schema module.
451 Internal schema modules will reside in the same namespace as their parent root schema. Since the
452 Internal schema reside in the same namespace as the root, the root schema uses `xsd:include` to
453 incorporate these internal modules. The UN/CEFACT schema modularity model ensures that logical
454 associations exist between root and internal schema modules and that individual modules can be reused
455 to the maximum extent possible.

456 [R 17] All UN/CEFACT internal schema modules **MUST** be in the same namespace as their
457 corresponding `rsm:RootSchema`.

458 UN/CEFACT internal schema modules will necessarily have semantically meaningful names. Internal
459 schema module names will identify the parent root schema module, the internal schema module function,
460 and the schema module itself.

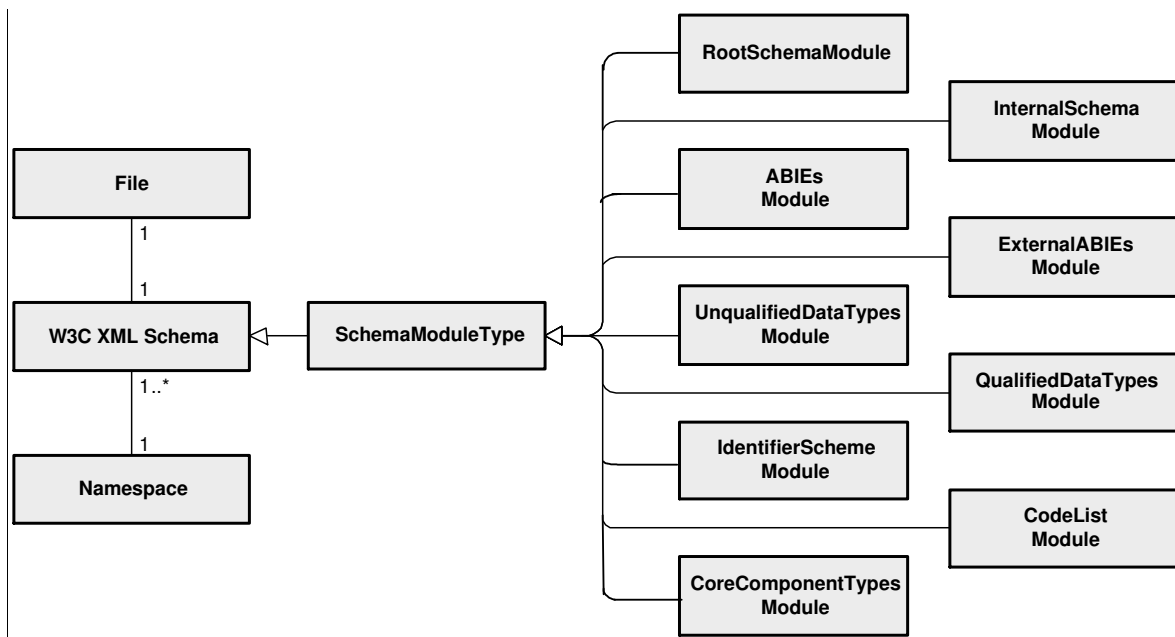
461 [R 18] Each UN/CEFACT internal schema module **MUST** be named
462 {ParentRootSchemaModuleName} {InternalSchemaModuleFunction} {Schema
463 Module}

464 **2.5.3 External Schema**

465 To adhere to the principles and rules contained in Section 2.5.1, schema modules will be created for
 466 reusable components. External schema modules will be used by several root schemas. The root schema
 467 may import one or more of these external schema modules. UN/CEFACT has identified the need for the
 468 following external schema modules:

- 469 • Core Component Types
- 470 • Unqualified Data Types⁵
- 471 • Qualified Data Types
- 472 • Reusable ABIEs
- 473 • Code Lists
- 474 • Identifier Lists (when the functionality mimics that of a code list)
- 475 • Other External Standard Body ABIE modules

476 These external schema modules are reflected in Figure 2-4.
 477
 478



479
 480

481 **Figure 2-4 UN/CEFACT Schema Modules**

482 **2.5.3.1 Core Component Type Schema Module**

483 A schema module is required to represent the normative form for CCTs from CCTS. This schema
 484 module will be used as the normative reference for all CCTS based XML instantiations.

485 [R 19] A core component type schema module **MUST** be created

486 The Core Component schema module will have a standardized name that uniquely differentiates it from
 487 other UN/CEFACT schema modules.

488 [R 20] The `cct:CoreComponentType` schema module **MUST** be named "CCTS CCT Schema Module"

⁵ The terms “unqualified data type” and “qualified data type” refer to the ISO 11179 concept of qualifiers for name constructs, not to the xml namespace concept of qualified and unqualified.

489 2.5.3.2 Unqualified Data Type Schema Module

490 A single unqualified data type schema module is required. This schema module will contain a type
491 definition for each data type expressed by approved CCTS primary and secondary representation terms.

492 [R 21] An unqualified data type schema module MUST be created

493 The unqualified data type schema module will have a standardized name that uniquely differentiates it
494 from other UN/CEFACT schema modules.

495 [R 22] The `udt:UnqualifiedDataType` schema module MUST be named "UN/CEFACT Unqualified
496 Data Type Schema Module"

497 2.5.3.3 Qualified Data Type Schema Module

498 A single qualified data type schema module is required. This schema module will contain a type definition
499 for each qualified data type as defined in CCTS.

500 [R 23] A qualified data type schema module MUST be created

501 The qualified data type schema module will have a standardized name that uniquely differentiates it from
502 other UN/CEFACT schema modules.

503 [R 24] The `qdt:QualifiedDataType` schema module MUST be named "UN/CEFACT Qualified Data
504 Type Schema Module"

505 2.5.3.4 Reusable Aggregate Business Information Entity Schema Module

506 A single reusable aggregate business information entity schema module is required. This schema
507 module will contain a type definition for every reusable ABIE in the UN/CEFACT Core Component
508 Library.

509 [R 25] An aggregate business information entity schema module MUST be created

510 The reusable aggregate business information entity schema module will have a standardized name that
511 uniquely differentiates it from other UN/CEFACT schema modules.

512 [R 26] The `ram:ReusableAggregateBusinessInformationEntity` schema module MUST be
513 named "UN/CEFACT Aggregate Business Information Entity Schema Module"

514 2.5.3.5 Code List Schema Modules

515 In cases where a code list is required or used, reusable code list schema modules will be created to
516 minimize the impact of code list changes on document and other reusable schema. Each reusable code
517 list schema module will contain enumeration values for codes and code values.

518 [R 27] Reusable code list schema modules MUST be created to convey code list enumerations

519 Code list schema modules will have a standardized name that uniquely differentiates it from other
520 UN/CEFACT schema modules and external organization generated code list modules.

521 [R 28] The name of each `clt:CodeList` schema module MUST be of the form: <Code List Agency
522 Name><Code List Name> - Code List Schema Module
523 Where:
524 Code List Agency Name = Agency that maintains the code list
525 Code List Name = The name of the code list as assigned by the agency that maintains the code
526 list

527 Example 2-7:

528 `UN/CEFACT Account Type Code - Code List Schema Module`

529 **2.5.3.6 Identifier List Schema Modules**

530 In those cases where identifier schemes mimic the functionality of code lists, reusable identifier lists
531 schema modules will be created to minimize the impact of identifier list changes on root and other
532 reusable schema. Each reusable identifier list schema module will contain enumeration values for codes
533 and code values.

534 [R 29] An Identifier List schema module **MUST** be created for each identifier list that mimics code list
535 functionality to convey enumerations of the identifier list value and token for that value

536 Identifier list schema modules will have a standardized name that uniquely differentiates it from other
537 UN/CEFACT schema modules or external organization generated schema modules.

538 [R 30] The name of each UN/CEFACT identifier list schema module **MUST** be of the form: <Identification
539 Scheme Agency Name><Identification Scheme Name> - Identifier List Schema Module

540 Where:

541 Identification Scheme Agency Name = Agency that maintains the identifier list

542 Identification Scheme Name = Name as assigned by the agency that maintains the identifier list

543 **Example 2-8:**

544 `ISO Country Identifier - Identifier List Schema Module`

545

546 **2.5.3.7 External Standards Body Aggregate Business Information Entity Schema** 547 **Modules**

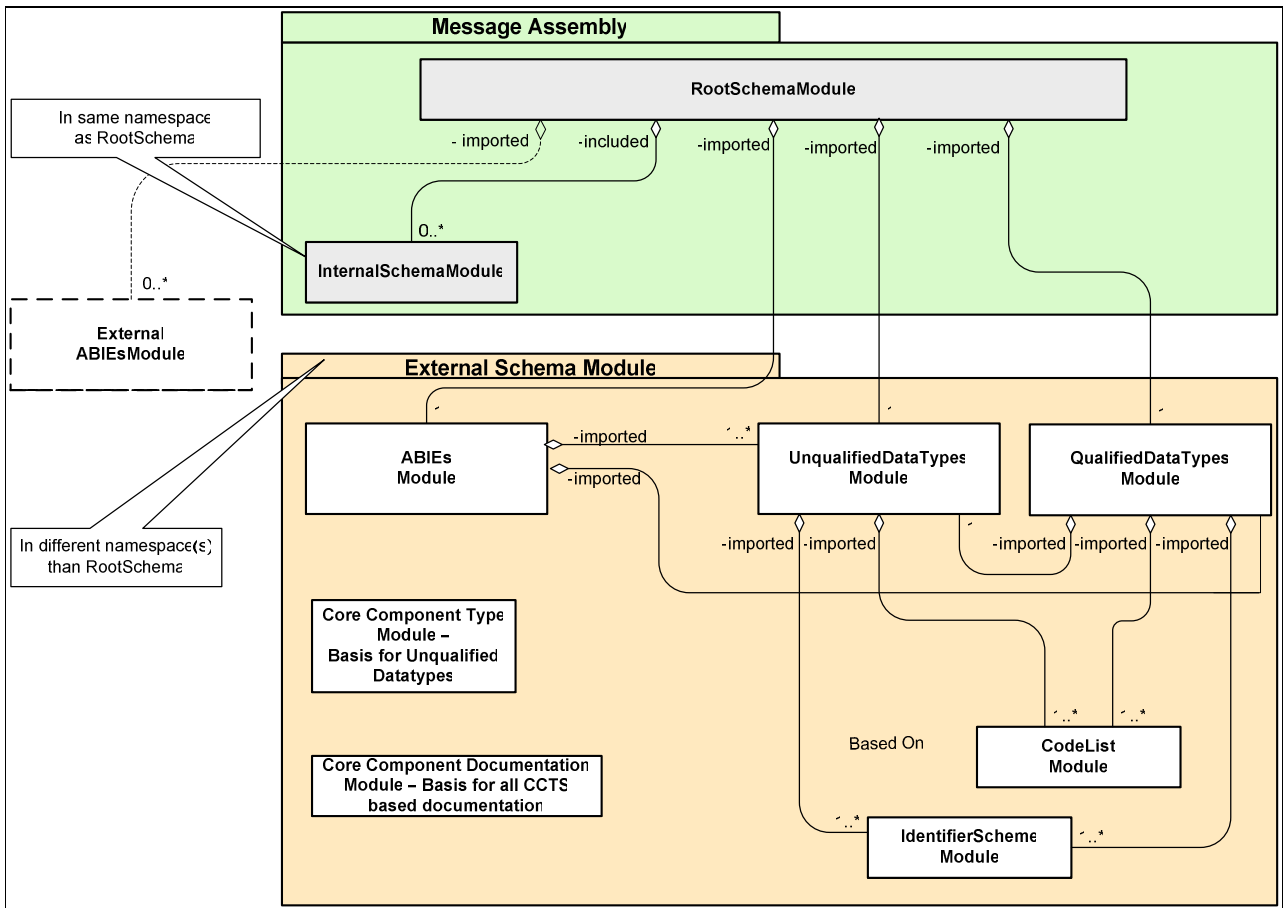
548 The external Standards Body ABIE modules are those reusable XML constructs created by other
549 standards bodies and made available for reuse. UN/CEFACT will only import external ABIE modules
550 when their contents are in strict conformance to the requirements of the CCTS and this document.

551 [R 31] Imported schema modules **MUST** be fully conformant with UN/CEFACT naming and design rules
552 and the Core Components Technical Specification.

553 **2.6 Namespace Scheme**

554 As defined in the W3C specification, "XML namespaces provide a simple method for qualifying element
555 and attribute names used in Extensible Markup Language documents by associating them with
556 namespaces identified by URI references."⁶ This enables interoperability and consistency in the XML
557 artifacts for the extensive library of reusable types and schema modules. The UN/CEFACT reusability
558 approach that maximizes reuse of defined named types and locally declared elements and attributes
559 within those types (See Section 2.4) and modularity approach of multiple reusable schema modules (See
560 Section 2.5) proscribe just such an approach. There exists specific relationships between the various
561 internal and external schema modules identified in Section 2.8.3 with respect to their namespaces.
562 These relationships are defined in Figure 2.5. Accordingly, a sufficiently robust namespace scheme is
563 essential.
564

⁶ World Wide Web Consortium, Namespaces in XML, 14 January 1999

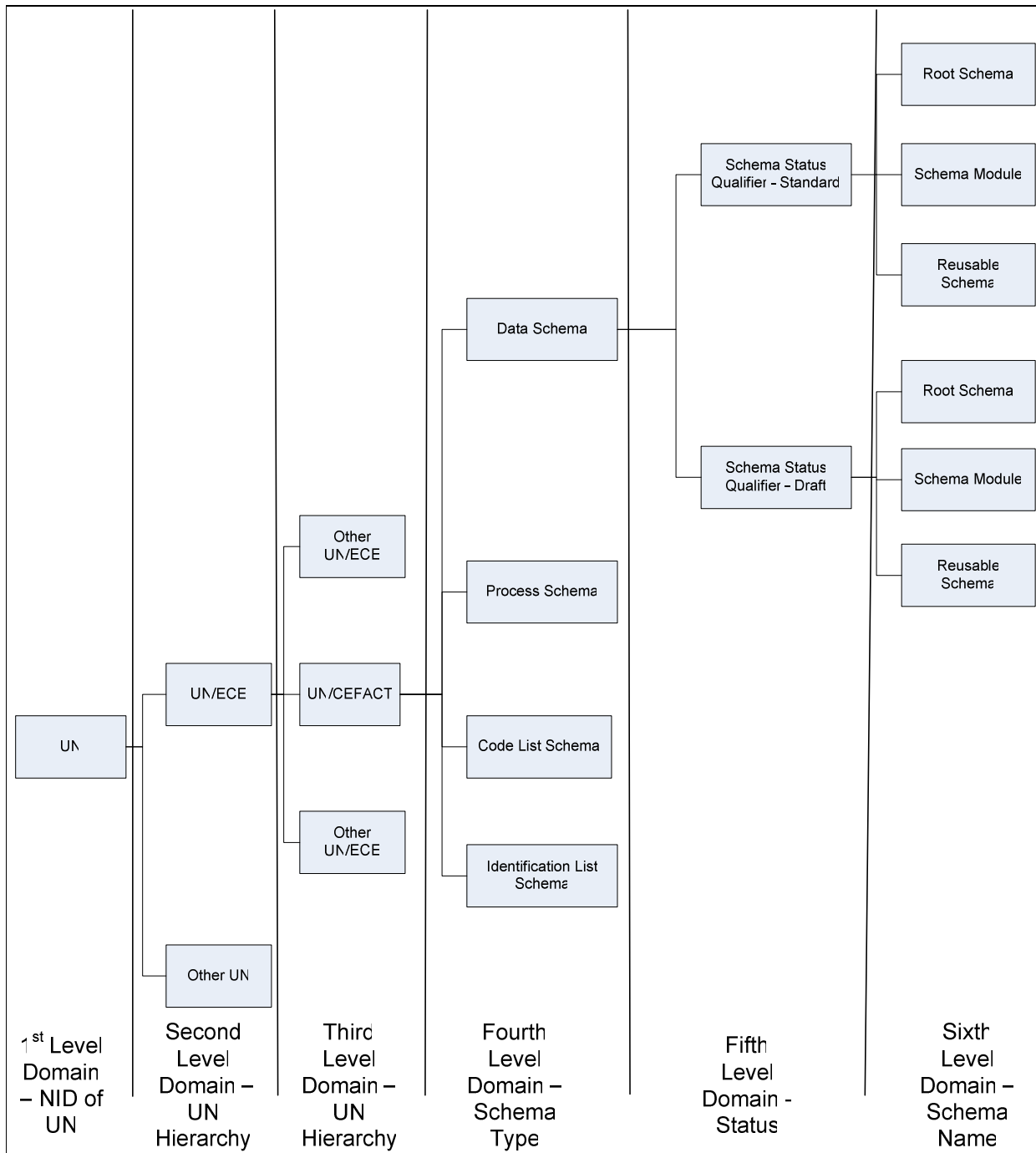


565
566

567 **Figure 2-5: UN/CEFACT Schema Modularity Scheme**

568 **2.6.1 UN/CEFACT Namespace Scheme**

569 In establishing a UN/CEFACT approach to namespaces, it is important to recognize that in addition to
 570 XML requirements, many other requirements exist for a standardized namespace approach. Accordingly,
 571 an overarching UN/CEFACT namespace scheme must be sufficiently flexible and robust to accommodate
 572 both XML and other syntax requirements. Figure 2-6 reflects such an approach and will be used as the
 573 basis for determining the namespace structure and rules that follow.



574
575

576 **Figure 2-6: UN/CEFACT Namespace Scheme**

577 **2.6.2 Declaring Namespace**

578 Best practice dictates that every schema module have its own namespace with the exception that internal
579 schema modules will be in the same namespace as the root schema.

580 [R 32] Every UN/CEFACT defined or used schema module MUST have a namespace declared, using the
581 `xsd:targetNamespace` attribute.

582 2.6.3 Namespace Persistence

583 Namespaces also provide a means for achieving consistency and harmonization between schema
584 versions. UN/CEFACT has chosen to align namespace versioning with schema versioning and
585 modularity. The UN/CEFACT modularity approach provides for grouping of reusable schemas by a root
586 schema. Many of these schema are intended to be reused across multiple schema. Others are unique to
587 a particular root schema. The root schema and those schema modules that are unique to it are
588 considered a **schema set**. The contents of a **schema set** are so interrelated that proper management
589 dictates that both versioning and namespace of all members of the set be in synchronization. Schema
590 sets are therefore assigned to a single, versioned namespace. Other schema modules are also best
591 managed by being assigned to their own unique versioned namespaces. Accordingly, with the exception
592 of internal schema modules, each UN/CEFACT schema module will have its own namespace and each
593 namespace will be versioned.

594 [R 33] Every defined or used schema module version other than internal schema modules MUST have its
595 own unique namespace.

596 Once a namespace declaration is published, any change would result in an inability to validate instance
597 documents citing the namespace. Accordingly, a change in the construct or contents of the namespace
598 should not be allowed.

599 [R 34] UN/CEFACT published namespace declarations or contents MUST never be changed.

600 2.6.4 Namespace Uniform Resource Identifiers

601 Namespaces must be persistent. Namespaces should be resolvable. Uniform Resource Indicators
602 (URIs) are used for identifying a namespace. Within the URI space, options include Uniform Resource
603 Locators (URLs) and Uniform Resource Names (URNs). URNs have an advantage in that they are
604 persistent. URLs have an advantage in that they are resolvable. After careful consideration,
605 UN/CEFACT has determined that URNs are most appropriate as persistence is of a higher priority, and
606 efforts are underway to make URN's resolvable.

607 [R 35] UN/CEFACT namespaces MUST be defined as URNs

608
609 To ensure consistency, each UN/CEFACT namespace will have the same general structure. This
610 namespace structure will follow the provisions of Internet Engineering Task Force (IETF) Request For
611 Comments (RFC) 2141 – URN Syntax. That specification calls for a standardized URN syntax structure
612 as follows: (phrases enclosed in quotes are REQUIRED):

613
614
$$\langle \text{URN} \rangle ::= \text{"urn:"} \langle \text{NID} \rangle \text{"."} \langle \text{NSS} \rangle$$

615
616 where :

617 $\langle \text{NID} \rangle$ = the Namespace Identifier

618 $\langle \text{NSS} \rangle$ = the Namespace Specific String.

619 The leading "urn:" sequence is case-insensitive.

620 The Namespace ID determines the syntactic interpretation of the Namespace Specific String

621

622 Following this pattern, the UN/CEFACT namespace general structure for a namespace name should be:

623 urn:un:unece:uncefact:<schematype>:<status>:<name>:<major>:<minor>[:<revision>]

624

625 Where:

626 • Namespace Identifier (NID) = UN

627 • Namespace Specific String =

628 unece:uncefact:<schematype>:<status>:<name>:<major>:<minor>[:<revision>] with unece and
629 uncefact as fixed value second and third level domains within the NID of un

630 • schematype = a token identifying the type of schema module: `data|process|codelist`

631 • status = the status of the schema as: `draft|standard`

632 • name = the name of the module (using underscore as separator)

633 • major = The major version number. Sequentially assigned, first release starting with the number 1.

- 634 • minor = The minor version number within a major release. Sequentially assigned, first release starting
635 with the number 0. Not applicable for codelist schema.
636 • revision = Sequentially assigned alphanumeric character for each revision of a minor release. Only
637 applicable where status = draft. Not applicable for codelist schema.
638

639 [R 36] The names for namespaces MUST have the following structure while the schemas are at draft
640 status:
641 urn:un:unece:uncefact:<schematype>:draft:<name>:<major>:[<minor>]:[<rev
642 ision>]
643 Where:
644 schematype = a token identifying the type of schema module: data|process|codelist
645 name = the name of the module (using underscore as separator)
646 major = the major version number. Sequentially assigned, first release starting with the number 1.
647 minor = the minor version number within a major release. Sequentially assigned, first release
648 starting with the number 0. Not applicable for codelist schema.
649 revision = sequentially assigned alphanumeric character for each revision of a minor release.
650 Only applicable where status = draft and schema type does not equal codelist.

651 **Example 2-9: Namespace Name at Draft Status**

652 `"urn:un:unece:uncefact:data:draft:unqualifieddatatypesschemamodule:0:3:5"`

653

654 [R 37] The namespace names for schemas holding specification status MUST be of the form:
655 urn:un:unece:uncefact:<schematype>:standard:<name>:<major>:[<minor>]
656 Where:
657 schematype = a token identifying the type of schema module: data|process|codelist
658 name = the name of the module (using underscore as separator)
659 major = the major version number, sequentially assigned, first release starting with the number 1.
660 minor = the minor version number within a major release, sequentially assigned, first release
661 starting with the number 0. Not applicable for codelist schema.
662

663 **Example 2-10: Namespace Name at Specification Status**

664 `"urn:un:unece:uncefact:data:standard:unqualifieddatatypesschemamodule:1:0"`

665

666 **2.6.5 Namespace Constraint**

667 To ensure consistency in declaring namespaces, a namespace should only be declared for an XML
668 construct by the owner of that namespace – unless specifically designed as a generic namespace such
669 as xsi. Accordingly, UN/CEFACT namespaces will only contain XML constructs created and assigned by
670 UN/CEFACT.

671 [R 38] UN/CEFACT namespaces MUST only contain UN/CEFACT developed schema modules.

672 **2.6.6 UN/CEFACT Schema Namespace Tokens**

673 A unique token will be defined for each namespace. The exact token for each type of namespace will be
674 defined by the applicable schema module subsection in Section 4.

675 **2.7 Schema Location**

676 Schema locations are required to be in the form of a URI scheme. Schema locations are typically the
677 same as their namespaces. Schema locations are typically defined as URL based URI schemes because
678 of resolvability limitations of URN based URI schemes. However, UN/CEFACT schema use a URN

679 based URI scheme for namespace declarations because persistence is considered more important than
680 resolvability. In recognition of the need for resolvability of schema location, until such time as URNs
681 become fully resolvable, UN/CEFACT will store schema in locations identified using a URL based URI
682 scheme aligned with the URN based URI scheme used for the namespace declaration as follows:
683 urn:un:unece:uncefact:<schematype>:<status>:<name>:<major>:<minor>:[<revision>]

684 [R 39] The general structure for schema location MUST be:
685 `http://www.unece.org/uncefact/<schematype>/<name>_<major>.<minor>`
686 `[<revision>][<status>].xsd`
687 Where:
688 `schematype` = a token identifying the type of schema module: data|process|odelist
689 `name` = the name of the module (using underscore as separator)
690 `major` = the major version number, sequentially assigned, first release starting with the number 1.
691 `minor` = the minor version number within a major release, sequentially assigned, first release
692 starting with the number 0.
693 `revision` = sequentially assigned alphanumeric character for each revision of a minor release.
694 Only applicable where `status` = draft.
695 `status` = the status of the schema as: draft|standard

696 [R 40] Each `xsd:schemaLocation` attribute declaration MUST contain a persistent and resolvable
697 URL.

698 [R 41] Each `xsd:schemaLocation` attribute declaration URL MUST contain an absolute path.

699 [R 42] Schema modules MUST be located under the directory:
700 `http://www.unece.org/uncefact/cc/schema/<schema-mod-name>.xsd`

701

702 2.8 Versioning

703 [R 43] A UN/CEFACT namespace URN is divided into three parts. First, is the standard UN/CEFACT
704 namespace information. Second, is the description of the purpose of the namespace. Third, is the
705 version information. The version information will in turn be divided into major (or incompatible)
706 and minor (or compatible) fields. The minor field has an optional revision extension.

707 2.8.1 Major Versions

708 A major version of a UN/CEFACT schema module constitutes significant and/or non-backwards
709 compatible changes. If any XML instance based on such older UN/CEFACT schema attempts validation
710 against the newer version, it will experience validation errors. A new major version will be produced when
711 significant and/or non-backwards compatible changes occur, i.e.

- 712 • Removing or changing values in enumerations
- 713 • Changing of element names, type names and attribute names
- 714 • Changing the structures so as to break polymorphic processing capabilities
- 715 • Deleting or adding mandatory elements or attributes
- 716 • Changing cardinality from mandatory to optional

717 Major version numbers are reflected in the namespace declaration as follows:

718 urn:un:unece:uncefact:<schematype>:<status>:<name>:<major>:0

719 Where:

- 720 • major = the first release starts with the number 1.
- 721 • minor = always 0 for major release numbers.

722 [R 44] Every schema major version MUST have the URI of:
723 urn:un:unece:uncefact:<schematype>:<status>:<name>:<major>:0:[<revision>]

724 Major version numbers should be based on logical progressions to ensure semantic understanding of the
725 approach and guarantee consistency in representation. Non-negative, sequentially assigned incremental
726 integers satisfy this requirement.

727 [R 45] Every UN/CEFACT schema and schema module major version number MUST be a sequentially
728 assigned incremental integer greater than zero.

729 2.8.2 Minor Versions

730 Within a major version of an UN/CEFACT schema module there can be a series of minor, or compatible,
731 changes. The minor versioning of an UN/CEFACT schema module determines its compatibility with
732 UN/CEFACT schema modules with preceding and subsequent minor versions within the same major
733 version. The minor versioning scheme thus helps to establish backward and forward compatibility. Minor
734 versions will only be increased when compatible changes occur, i.e

- 735 • Restrictions of types, but not to required elements
- 736 • Adding values to enumerations
- 737 • Optional extensions
- 738 • Restrictions on facets
- 739 • Delete or add optional elements
- 740 • Change cardinality from optional to mandatory

741 [R 46] Minor versioning MUST be limited to declaring new optional xsd constructs, extending existing xsd
742 constructs and refinements of an optional nature.

743 Minor version numbers are reflected in the namespace declaration as follows:

744 urn:un:unece:uncefact:<schematype>:<status>:<name>:<major-number>:non-zero:[<revision>]

745 Where:

- 746 • major = the major version number, sequentially assigned, first release starting with the number 1
- 747 • minor = always positive integer

748 [R 47] Every UN/CEFACT schema minor version MUST have the URI of:

749 urn:un:unece:uncefact:cc:schema:<name>:<major-number>:<non-zero integer>:[<revision>]

750 Just like major version numbers, minor version numbers should be based on logical progressions to
751 ensure semantic understanding of the approach and guarantee consistency in representation. Non-
752 negative, sequentially assigned incremental integers satisfy this requirement.

753

754 Minor version changes are not allowed to break compatibility with previous minor versions. Compatibility
755 includes consistency in naming of the schema constructs. UN/CEFACT minor version changes will not
756 include renaming the XML construct.

757 [R 48] For UN/CEFACT minor version changes, the name of the schema construct MUST NOT change

758 Semantic compatibility across minor versions is essential.

759 [R 49] Changes in minor versions MUST NOT break semantic compatibility with prior versions.

760 For a particular namespace, the parent major version and subsequent minor versions of a major version
761 establish a linearly linked relationship. Since each minor version is assigned its own namespace, for
762 conformance purposes, the first minor version must incorporate all XML constructs present in the parent
763 major version, and each new minor version needs to incorporate all XML constructs present in the
764 immediately preceding minor version.

765 [R 50] UN/CEFACT minor version schema MUST incorporate all XML constructs from the immediately
766 preceding major or minor version schema.

767

768 [Note]

769 There has been much discussion surrounding the issue of namespaces and versioning.
770 ATG solicits input from interested parties on the pro's and con's of assigning a unique
771 namespace for each minor version as opposed to assigning a new namespace for only
772 major versions and having all minor versions have the same namespace as its major
773 version.

775

3 General XML Schema Language Conventions

776
777

[R 51] All UN/CEFACT schema design rules MUST be based on the W3C XML Schema Recommendations: XML Schema Part 1: Structures and XML Schema Part 2: DataTypes.

778
779

[R 52] All schemata and instances MUST be based on the W3C suite of technical specifications holding recommendation status.

780

3.1 Schema Construct

781

[R 53] Element form default MUST be declared as qualified.

782

[R 54] Attribute form default MUST be declared as unqualified.

783

Example 3-1: Element and Attribute Form Default

784
785
786

```
<xsd:schema targetNamespace=" ... see namespace ...  
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"  
  elementFormDefault="qualified" attributeFormDefault="unqualified">
```

787

788

3.1.1 Constraints on Schema Construction

789
790

[R 55] The “xsd” prefix MUST be used.
 xmlns:xsd=http://www.w3.org/2001/xmlSchema

791

[R 56] The XSI prefix SHALL be used where appropriate.

792

[R 57] Processing Instructions MUST NOT be used.

793

[R 58] Notations MUST NOT be used.

794

[R 59] Wildcards MUST NOT be used.

795

[R 60] The **xsd:any** element MUST NOT be used.

796

[R 61] The **xsd:any** attribute MUST NOT be used.

797

[R 62] Mixed content MUST NOT be used (excluding documentation).

798

[R 63] Substitution groups MUST NOT be used.

799

[R 64] ID/IDREF MUST NOT be used.

800

[R 65] Key/KeyRef MUST be used for information association.

801

[R 66] The absence of a construct or data MUST NOT carry meaning.

802

803 3.2 Attribute and Element Declarations

804 3.2.1 Attributes

805 3.2.1.1 Usage of Attributes

806 User defined attributes are only used to convey the supplementary components of core component types.
807 However, `xsd:built-in` attributes will be used as described elsewhere in this document.

808 [R 67] User defined attributes MUST only be used to convey core component type (CCT) supplementary
809 component information.

810 3.2.1.2 Constraints on Attribute Declarations

811 In general, the absence of an element in an XML schema does not have any particular meaning - it may
812 indicate that the information is unknown, or not applicable, or the element may be absent for some other
813 reason. The XML schema specification does however provide a feature, the nillable attribute, whereby an
814 element may be transferred with no content, but still use its attributes and thus carry semantic meaning.
815 In order to respect the principles of the CCTS and to retain semantic clarity the nillability feature of `xsd`
816 will not be used.

817 [R 68] The `nillable` attribute MUST NOT be used.

818 3.2.2 Elements

819 3.2.2.1 Usage of Elements

820 Elements are used to define the top-level complex type that represents business process, all attributes of
821 an object class (BBIE and ASBIE), and the object class itself (ABIE).

822 3.2.2.2 Element Declaration

823 [R 69] All element declarations MUST be local except for a root element that must be declared globally.

824 [R 70] Empty elements MUST NOT be used.

825 The `xsd:enumeration` element may be used within reusable or internal schema modules if the list of
826 enumerated values is less than 10, are not represented by a token, and are considered by TBG to be
827 static and particular to the business processes.

828 [R 71] The `xsd:type` of each leaf element declaration MUST be of the data type of its source business
829 information entity (BBIE) or complex type of its source association business information entity
830 (ASBIE).

831 Example 3-2:

```
832 <xsd:complexType name="AccountType">
833 <xsd:annotation>
834   ...see annotation...
835 </xsd:annotation>
836 <xsd:sequence>
837   <xsd:element name="ID" type="udt:IdentifierType"
838     minOccurs="0" maxOccurs="unbounded">
839     <xsd:annotation>
840       ...see annotation...
841     </xsd:annotation>
842   </xsd:element>
843   <xsd:element name="Status" type="ram:StatusType"
844     minOccurs="0" maxOccurs="unbounded">
845     <xsd:annotation>
846       ...see annotation...
```

```
847     </xsd:annotation>
848   </xsd:element>
849   <xsd:element name="Name" type="udt:NameType"
850     minOccurs="0" maxOccurs="unbounded">
851     <xsd:annotation>
852       ... see annotation ...
853     </xsd:annotation>
854   </xsd:element>
855   ...
856 </xsd:sequence>
857 </xsd:complexType>
858
```

859 3.2.2.3 Constraints on Element Declarations

860 [R 72] The element declaration of `xsi:nil` MUST NOT appear in any conforming instance.

861 [R 73] The `xsd:a11` element MUST NOT be used.

862 [R 74] Mixed-content elements MUST NOT be used.

863

864 3.3 Type Definitions

865 3.3.1 Usage of Types

866 [R 75] All type definitions MUST be named.

867 Example 3-3:

```
868 <xsd:complexType name="AccountType">
869 <xsd:annotation>
870   ... see annotation ...
871 </xsd:annotation>
872 <xsd:sequence>
873   ... see element declaration ...
874 </xsd:sequence>
875 </xsd:complexType>
```

876

877 [R 76] Type definitions MUST NOT duplicate the functionality of existing `xsd:built-in` simple types.

878

879 3.3.2 Simple Type Definitions

880 [R 77] User defined simple types definitions MUST NOT be used.

881 Built-in simple types must always be used where they satisfy the business requirements. Where the
882 business requirements cannot be satisfied, user defined complex type definitions will be used.

883 Example 3-4: Simple Types in Unqualified Data Types Schema Module

```
884 <xsd:simpleType name="DateTimeType">
885 <xsd:annotation>
886   ... see annotation ...
887 </xsd:annotation>
888 <xsd:restriction base="xsd:dateTime"/>
889 </xsd:simpleType>
```

890

891 **Example 3-5: Simple Types in Code Lists Module**

```
892 <xsd:simpleType name="CurrencyCodeContentType">
893 <xsd:restriction base="xsd:token">
894 <xsd:enumeration value="ADP">
895 ...see enumeration of code lists ...
896 </xsd:enumeration>
897 <xsd:annotation>
898 ... see annotation ...
899 </xsd:annotation>
900 </xsd:restriction>
901 </xsd:simpleType>
```

902

903 **3.3.3 Complex Type Definitions**

904 User defined complex types may be used when built-in simple types do not satisfy the business
905 requirements or when an aggregate business information entity (ABIE) must be defined.

906 **Example 3-6: Complex Type of Object Class "AccountType"**

```
907 <xsd:complexType name="AccountType">
908 <xsd:annotation>
909 ... see annotation ...
910 </xsd:annotation>
911 <xsd:sequence>
912 ... see element declaration ...
913 </xsd:sequence>
914 </xsd:complexType>
```

915

916 **3.4 Use of XSD Extension and Restriction**

917 The general philosophy is that all UN/CEFACT schema constructs will follow the model defined in Figure
918 2.1. These schema constructs are based on the concept that the underlying semantic structures of the
919 core components and business information entities are normative forms of standards that developers are
920 not allowed to alter without coordination of appropriate TBG groups (including TBG17 - Harmonization)
921 and ICG. Accordingly, as business requirements dictate, new schema constructs will be created and new
922 types defined and elements declared as appropriate and the concept of derivation through the use of xsd
923 extension and restriction will only be used in limited circumstances as described below.

924 **3.4.1 XSD Extension**

925 [R 78] Extension MUST only be used in the **cct:CoreComponentTypes** schema module and the
926 **udt:UnqualifiedDataType** schema module. When used it MUST only extend a built-in
927 **xsd:datatype**.

928 **3.4.2 Restriction**

929 The CCTS specification employs the concept of semantic restriction in creating specific instantiations of
930 core components. Accordingly, **xsd:restriction** will be used as appropriate to define types that are
931 derived from the existing types. Where used, the derived types must always be renamed. Simple and
932 complex type restrictions may be used.

933 Example 3-7: Restriction of Simple Type

```
934 <xsd:simpleType name="IndicatorType">
935 <xsd:annotation>
936   ... see annotation ...
937 </xsd:annotation>
938 <xsd:restriction base="xsd:boolean">
939   <xsd:pattern value="false"/>
940   <xsd:pattern value="true"/>
941 </xsd:restriction>
942 </xsd:simpleType>
```

943 3.5 Annotation

944 In the UN/CEFACT schema modules the `xml:annotations` will only be used to provide documentation.

945 3.5.1 Documentation

946 The annotation documentation will be used to convey all metadata as specified in the CCTS, i.e., to
947 convey the semantic content carried in the XML construct. The following sets of annotations are required
948 as defined in section 4 in type definitions and element declarations (the representation of each item in
949 XML code is shown in parens):

- 950
- 951 • **Unique Identifier:** The unique identifier assigned to the artefact in the library. (**UniqueID**)
- 952 • **Category Code:** The category to which the artefact belongs. (**CategoryCode**)
- 953 • **Dictionary Entry Name:** The complete name (not the tag name) of the artefact in the library.
954 (**DictionaryEntryName**)
- 955 • **Version:** The version of the artefact as assigned by the registry. (**VersionID**)
- 956 • **Cardinality:** An indication of whether the property represents a not-applicable, optional, mandatory
957 and/or repetitive characteristic of the object. (**CardinalityText**)
- 958 • **Definition:** The semantic meaning of the artefact. (**DefinitionText**)
- 959 • **Object Class:** The Object Class represented by the artefact. (**ObjectClassTermName**)
- 960 • **Property Term:** The Property Term represented by the artefact. (**PropertyTermName**)
- 961 • **Representation Term:** The Representation Term represented by the artefact.
962 (**RepresentationTermName**)
- 963 • **Associated Object Class Term:** The Associated Object Class Term represented by the artefact.
964 (**AssociatedObjectClassTermName**)
- 965 • **Qualifier Term:** A term(s) that qualifies the Object Class and/or Property. (**QualifierTermName**)
- 966 • **Business Process Context:** A valid value describing the Business Process contexts for which this
967 construct has been designed. Default is "In All Contexts". (**BusinessProcessContext**)
- 968 • **Geopolitical/Region Context:** A valid value describing the Geopolitical/Region contexts for which this
969 construct has been designed. Default is "In All Contexts". (**GeopoliticalOrRegionContext**)
- 970 • **Official Constraints Context:** A valid value describing the Official Constraints contexts for which this
971 construct has been designed. Default is "None". (**OfficialConstraintContext**)
- 972 • **Product Context:** A valid value describing the Product contexts for which this construct has been
973 designed. Default is "In All Contexts". (**ProductContext**)
- 974 • **Industry Context:** A valid value describing the Industry contexts for which this construct has been
975 designed. Default is "In All Contexts". (**IndustryContext**)
- 976 • **Role Context:** A valid value describing the Role contexts for which this construct has been designed.
977 Default is "In All Contexts". (**BusinessProcessRoleContext**)
- 978 • **Supporting Role Context:** A valid value describing the Supporting Role contexts for which this
979 construct has been designed. Default is "In All Contexts". (**SupportingRoleContext**)
- 980 • **System Capabilities Context:** A valid value describing the Systems Capabilities contexts for which
981 this construct has been designed. Default is "In All Contexts". (**SystemCapabilitiesContext**)
- 982 • **Usage Rule:** A constraint that describes specific conditions which are applicable to the artefact.
983 (**UsageRuleText**)
- 984 • **Business Term:** A synonym term under which the artefact is commonly known and used in business.
985 (**BusinessTermName**)
- 986

987 Appendix B specifies normative information on the specific annotation required for each of the artefacts.

988 **Example 3-8: Example of annotation**

```
989 <xsd:annotation>
990 <xsd:documentation xml:lang="en">
991 <ccts:UniqueID>UN00000002</ccts:UniqueID>
992 <ccts:CategoryCode>BBIE</ccts:CategoryCode>
993 <ccts:DictionaryEntryName>Account.
994 Identifier</ccts:DictionaryEntryName>
995 <ccts:Definition>The identification of a specific
996 account.</ccts:Definition>
997 <ccts:VersionID>1.0</ccts:VersionID>
998 <ccts:ObjectClassTermName>Account</ccts:ObjectClassTermName>
999 <ccts:PropertyTermName>Identifier</ccts:PropertyTermName>
1000 <ccts:RepresentationTermName>Identifier</ccts:RepresentationTermName>
1001 <ccts:BusinessTermName>Account Number</ccts:BusinessTermName>
1002 </xsd:documentation>
1003 </xsd:annotation>
```

1004
1005 Each UN/CEFACT construct containing a code should include documentation that will identify the code
1006 list(s) that must be minimally supported when the construct is used.

1007 4 XML Schema Modules

1008 This section describes the requirements of the various XML schema modules that will be incorporated
1009 within the UN/CEFACT library.

1010 4.1 Root Schema

1011 The root schema serves as the container for all other schema content that is required to fulfill a business
1012 process. The root schema resides in its own namespace and imports external schema modules as
1013 needed. It may also include internal schema modules that reside in its namespace.

1014 4.1.1 Schema Construct

1015 Each root schema will be constructed in a standardized format in order to ensure consistency and ease of
1016 use. The specific format is shown in the example below and must adhere to the format of the relevant
1017 sections as detailed in Appendix A.

1018 Example 4-1: Structure of RootSchema Module

```
1019 <?xml version="1.0" encoding="UTF-8"?>
1020 <!-- edited with XMLSPY v5 rel. 4 U (http://www.xmlspy.com) by Gunther Stuhec (UBL) --
1021 >
1022 <!-- ===== -->
1023 <!-- ===== [MODULENAME] Schema Module; [VERSION] ===== -->
1024 <!-- ===== -->
1025 <!--
1026     Module of [MODULENAME],
1027     Agency: UN/CEFACT,
1028     Version: 0.3 Rev. 6
1029     Last change: 25. June 2004
1030
1031     Copyright (C) UN/CEFACT (2004). All Rights Reserved.
1032
1033     ... see copyright information ...
1034 -->
1035 <xsd:schema
1036     targetNamespace="urn:un:unece:uncefact:data:draft:[MODULENAME]:0:3:6"
1037     .. see namespaces ...
1038     xmlns:xsd="http://www.w3.org/2001/XMLSchema"
1039     elementFormDefault="qualified" attributeFormDefault="unqualified">
1040 <!-- ===== Includes ===== -->
1041 <!-- ===== -->
1042 <!-- ===== Include of [MODULENAME] ===== -->
1043     ... see includes ...
1044 <!-- ===== Imports ===== -->
1045 <!-- ===== -->
1046 <!-- ===== Import of [MODULENAME] ===== -->
1047     ... see imports ...
1048
1049 <!-- ===== Root Element ===== -->
1050 <!-- ===== -->
1051     ... see root element declaration ...
1052 <!-- ===== Type Definitions ===== -->
1053 <!-- ===== -->
1054 <!-- ===== Type Definition: [TYPE] ===== -->
1055 <!-- ===== -->
1056 <xsd:complexType name="[TYPENAME]">
1057 <xsd:restriction base="xsd:token">
1058     ... see type definition ...
1059 </xsd:restriction>
1060 </xsd:complexType>
1061 </xsd:schema>
```

1063 4.1.2 Namespace Scheme

1064 [R 79] The root schema module MUST be represented by the token "rsm".

1065 **Example 4-2: Structure of RootSchema Module**

1066 `xmlns:rsm="urn:un:unece:uncefact:data:draft:ExamplesSchemaModule:0:3:6"`

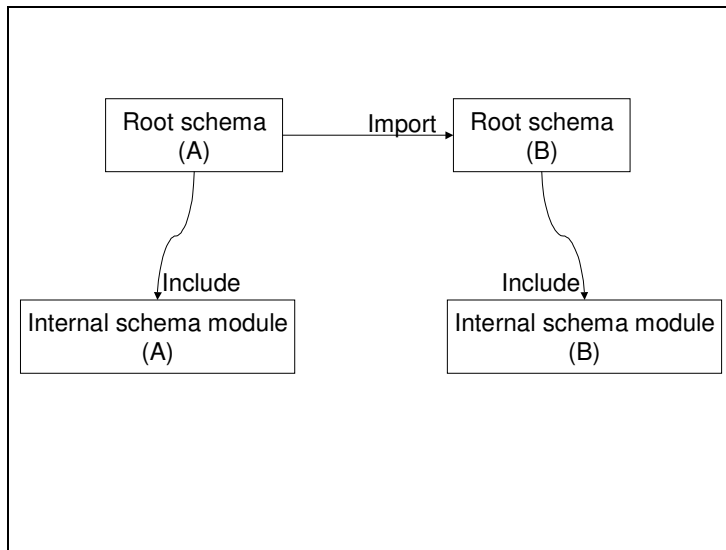
1067 **4.1.3 Imports and Includes**

1068 [R 80] The `rsm:RootSchema` MUST import the following schema modules:

- 1069 - `ram:ReusableABIE` Schema Module
- 1070 - `udt:UnqualifiedDataTypes` Module
- 1071 - `qdt:QualifiedDataTypes` Module

1072 The root schema may import other external schema modules as necessary provided they conform to
1073 UN/CEFACT naming and design rules.

1074 One root schema (root schema A) may also make use of ABIEs defined as part of another root schema
1075 (root schema B) or that root schemas internal schema module, i.e. type definitions and element
1076 declarations defined in another namespace. An example may be that the root schema for an Order
1077 Response message (root schema A) makes use of ABIEs defined as part of the schema definition for an
1078 Orders message (root schema B). If that is the case then such type definitions and element declarations
1079 should be imported in to the root schema (root schema A). To achieve this only the root schema (root
1080 schema B) in the namespace containing the type definitions and element declarations needed should be
1081 imported as this in itself included the subordinate internal schema modules.
1082



1083

1084 [R 81] A `rsm:RootSchema` in one UN/CEFACT namespace that is dependent upon type definitions or
1085 element declaration defined in another namespace MUST import the `rsm:RootSchema` from
1086 that namespace.

1087 [R 82] A `rsm:RootSchema` in one UN/CEFACT namespace that is dependant upon type definitions or
1088 element declarations defined in another namespace MUST NOT import Schema Modules from
1089 that namespace other then the `rsm:RootSchema`.

1090 [R 83] The `rsm:RootSchema` MUST include any internal schema modules that reside in the root schema
1091 namespace.

1092 **4.1.4 Root Element Declaration**

1093 Each UN/CEFACT business message has a single root element that is globally declared in the root
1094 schema representing the business information exchange. The root element is named according to the
1095 business information exchange that it represents and references the message assembly that contains the
1096 actual business information.

-
- 1097 [R 84] A single global element known as the root element MUST be globally declared in a
1098 **rsm:RootSchema**.
- 1099 [R 85] The root element MUST reflect the Message Assembly that is defined for the actual content of the
1100 business information.
- 1101 [R 86] The name of the root element MUST be the name of the Message Assembly with separators and
1102 spaces removed.
-

1103 **Example 4-3:**

```
1104 <!-- ===== Root Element ===== -->  
1105 <!-- ===== -->  
1106 <xsd:element name="PurchaseOrder" type="exp:PurchaseOrderType">  
1107 <xsd:annotation>  
1108 ... see annotation ...  
1109 </xsd:annotation>  
1110 </xsd:element>
```

1111

1112 4.1.5 Type Definitions

-
- 1113 [R 87] For the root element a corresponding complex type that represents the Message Assembly MUST
1114 be defined.
- 1115 [R 88] The name of the top-level complex type MUST be the name of the root element with the word
1116 "type" appended.
-

1117 4.1.6 Annotations

-
- 1118 [R 89] For every **rsm:RootSchema** root element declaration a structured set of annotations MUST be
1119 present in the following pattern:
-

- 1120 ■ **UniqueID** (mandatory): The identifier that references the Message Assembly
1121 instance in a unique and unambiguous way.
- 1122 ■ **CategoryCode** (mandatory): The category to which the object belongs. In this
1123 case the value will always be RSM.
- 1124 ■ **Name** (mandatory): The name of the Message Assembly
- 1125 ■ **VersionID** (mandatory): An indication of the evolution over time of a
1126 Message Assembly.
- 1127 ■ **Description** (mandatory): A brief description of the business information
1128 exchange.
- 1129 ■ **BusinessDomain** (mandatory, repetitive): The TBG group(s) that developed
1130 this Message Assembly.
- 1131 ■ **BusinessProcessContext** (mandatory, repetitive): The business process with
1132 which this Message Assembly is associated.
- 1133 ■ **GeopoliticalorRegionContext** (optional, repetitive): The geopolitical/region
1134 contexts for this Message Assembly.
- 1135 ■ **OfficialConstraintContext** (optional, repetitive): The official constraint
1136 context for this Message Assembly.

1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147

- **ProductContext** (optional, repetitive): The product context for this Message Assembly.
- **IndustryContext** (optional, repetitive): The industry context for this Message Assembly.
- **BusinessProcessRoleContext** (optional, repetitive): The role context for this Message Assembly.
- **SupportingRoleContext** (optional, repetitive): The supporting role context for this Message Assembly.
- **SystemCapabilitiesContext** (optional, repetitive): The system capabilities context for this Message Assembly.

1148 **4.2 Internal Schema**

1149 A UN/CEFACT internal schema module is a schema instance that contains schema constructs specific to
1150 a given root schema and that therefore is part of a schema set within a specific namespace.
1151 The internal schema module will contain schema constructs representing ABIEs that are specific to a
1152 given root schema. These constructs are subject to the same rules as those for reusable ABIEs as
1153 provided in sections 4.3.4, 4.3.5, and 4.3.6.

1154 **4.2.1 Schema Construct**

1155 Each internal schema will be constructed in a standardized format in order to ensure consistency and
1156 ease of use. The specific format is shown below and must adhere to the format of the relevant sections
1157 as detailed in Appendix A.

1158 **4.2.2 Namespace Scheme**

1159 [R 90] All UN/CEFACT internal schema modules MUST be in the same namespace as their
1160 corresponding **rsm:RootSchema**.

1161 The UN/CEFACT internal schema modules do not declare a target namespace, but instead reside in the
1162 namespace of their parent root schema. All internal schema modules are accessed from the root schema
1163 using **xsd:include**.

1164 [R 91] The internal schema module MUST be represented by the same token as its **rsm:RootSchema**.

1165 **4.2.3 Imports and Includes**

1166 The internal schema module does not import or include any other schema module as this is done within
1167 the root schema in the same namespace.

1168 **4.3 Reusable Aggregate Business Information Entities**

1169 The UN/CEFACT ABIE schema module is a schema instance that contains all of the reusable ABIEs.
1170 This schema module may thus be used (imported in to) in conjunction with any if the UN/CEFACT root
1171 schemas.

1172 **4.3.1 Schema Construct**

1173 The reusable ABIE schema will be constructed in a standardized format in order to ensure consistency
1174 and ease of use. The specific format is shown below and must adhere to the format of the relevant
1175 sections as detailed in Appendix A.

1176 **Example 4-4: Structure of Reusable ABIEs Schema Module**

```
1177 <?xml version="1.0" encoding="UTF-8"?>
1178 <!-- ===== -->
1179 <!-- ===== RAM Reusable ABIEs Schema Module ===== -->
1180 <!-- ===== -->
1181 <!--
1182     Module of Reusable ABIEs (Aggregate Business Information Entities),
1183     Agency: UN/CEFACT,
1184     Version: 0.3 Rev. 6,
1185     Last change: 25. June 2004
1186
1187     Copyright (C) UN/CEFACT (2004). All Rights Reserved.
1188
1189     ... see copyright information ...
1190 -->
1191 <xsd:schema
1192     targetNamespace=
1193     ... see namespace declaration ...
1194     xmlns:xsd="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
1195     attributeFormDefault="unqualified">
1196     <!-- ===== Imports ===== -->
1197     ... see imports ...
1198     <!-- ===== Type Definitions ===== -->
1199     <!-- ===== -->
1200     ... see type defintions ...
1201 </xsd:schema>
```

1202 **4.3.2 Namespace Scheme**

1203 [R 92] The schema module MUST be represented by the token “ram”.

1204 **Example 4-5: Namespace of Reusable Aggregate Business Information Entity Schema Module**

```
1205 "urn:un:unece:uncefact:data:draft:ReusableAggregateBusinessInformationEntitySchemaModu
1206 le:0:3:6"
```

1207 **Example 4-6: Schema-Element of Reusable ABIEs Schema Module**

```
1208 <xsd:schema
1209     targetNamespace=
1210     "urn:un:unece:uncefact:data:draft:ReusableAggregateBusinessInformationEntitySchemaModu
1211     le:0:3:6"
1212     xmlns:ram=
1213     "urn:un:unece:uncefact:data:draft:ReusableAggregateBusinessInformationSchemaModule:0:3
1214     :6"
```

1215

1216 **4.3.3 Imports and Includes**

1217 [R 93] The `ram:ReusableAggregateBusinessInformationEntity` schema MUST import the
1218 following schema modules:
1219 – `udt:UnqualifiedDataType` Module
1220 – `qdt:QualifiedDataType` Module
1221

1222 **Example 4-7: Import of required modules**

```
1223 <!-- ===== Imports ===== -->
1224 <!-- ===== -->
1225 <!-- ===== Import of Qualified Data Type Schema Module (QDT) ===== -->
1226 <!-- ===== -->
1227 <xsd:import
1228     namespace=
1229     "urn:un:unece:uncefact:data:draft:QualifiedDataTypeSchemaModule:0:3:6"
1230     schemaLocation="QualifiedDataTypeSchemaModule_0.3.6.xsd"/>
1231 <!-- ===== -->
```

```

1232 <!-- ===== Import of Unqualified Data Type Schema Module (UDT) ===== -->
1233 <!-- ===== -->
1234 <xsd:import
1235 namespace=
1236   "urn:un:uncefact:data:draft:UnqualifiedDataTypeSchemaModule:
1237   0:3:6"
1238   schemaLocation="UnqualifiedDataTypesSchemaModule_0.3.6.xsd"/>

```

1239

1240 4.3.4 Type Definitions

1241 [R 94] For every object class (ABIE) identified in the UN/CEFACT syntax-neutral model, a named
 1242 **xsd:complexType** MUST be defined.

1243 [R 95] The name of the ABIE **xsd:complexType** MUST be the **ccts:DictionaryEntryName** with the
 1244 separators removed and with the "Details" suffix replaced with "Type".

1245 For every complex type definition based on an ABIE object class, its **xsd:content model** will be
 1246 defined such that it reflects each property of the object class as a local element declaration, with its
 1247 cardinality and sequencing within the schema **xsd:content model** determined by the details of the
 1248 source business information entity (ABIE).

1249 [R 96] Every aggregate business information entity (ABIE) **xsd:complexType** definition **xsd:content**
 1250 **model** MUST use the **xsd:sequence** and/or **xsd:choice** elements with appropriate local
 1251 element declarations to reflect each property (BBIE or ASBIE) of its class.

1252 [R 97] Recursion of **xsd:sequence** and/or **xsd:choice** MUST NOT occur.

1253 No complex type may contain a sequence followed by another sequence or a choice followed by another
 1254 choice. However, it is permissible to alternate sequence and choice as in example 4-10.

1255 Example 4-8: Sequence within an object class

```

1256 <xsd:complexType name="AccountType" >
1257 <xsd:annotation>
1258   ...see annotation...
1259 </xsd:annotation>
1260 <xsd:sequence>
1261   <xsd:element name="ID" type="udt:IdentifierType"
1262     minOccurs="0" maxOccurs="unbounded">
1263     <xsd:annotation>
1264       ...see annotation...
1265     </xsd:annotation>
1266   </xsd:element>
1267   <xsd:element name="Status" type="ram:StatusType"
1268     minOccurs="0" maxOccurs="unbounded">
1269     <xsd:annotation>
1270       ...see annotation...
1271     </xsd:annotation>
1272   </xsd:element>
1273   <xsd:element name="Name" type="udt:NameType"
1274     minOccurs="0" maxOccurs="unbounded">
1275     <xsd:annotation>
1276       ...see annotation...
1277     </xsd:annotation>
1278   </xsd:element>
1279   ...
1280 </xsd:sequence>
1281 </xsd:complexType>

```

1282 Example 4-9: Choice

```

1283 <xsd:complexType name="LocationType">
1284 <xsd:annotation>
1285   ... see annotation ...
1286 </xsd:annotation>
1287 <xsd:choice>
1288   <xsd:element name="GeoCoordinate" type="ram:GeoCoordinateType"

```

```

1289     minOccurs="0">
1290     <xsd:annotation>
1291         ... see annotation ...
1292     </xsd:annotation>
1293 </xsd:element>
1294 <xsd:element name="Address" type="ram:AddressType"
1295     minOccurs="0">
1296     <xsd:annotation>
1297         ... see annotation ...
1298     </xsd:annotation>
1299 </xsd:element>
1300 <xsd:element name="Location" type="ram:LocationType"
1301     minOccurs="0">
1302     <xsd:annotation>
1303         ... see annotation ...
1304     </xsd:annotation>
1305 </xsd:element>
1306 </xsd:choice>
1307 </xsd:complexType>

```

1308 **Example 4-10: Sequence + Choice within Object Class "PeriodType"**

```

1309 <xsd:complexType name="PeriodType">
1310     ...
1311     <xsd:sequence>
1312         <xsd:element name="DurationDateTime"
1313             type="qdt:DurationDateTimeType" minOccurs="0"
1314             maxOccurs="unbounded">
1315             ...
1316         </xsd:element>
1317         ...
1318         <xsd:choice>
1319             <xsd:sequence>
1320                 <xsd:element name="StartTime" type="udt:TimeType"
1321                     minOccurs="0">
1322                     ...
1323                 </xsd:element>
1324                 <xsd:element name="EndTime" type="udt:TimeType"
1325                     minOccurs="0">
1326                     ...
1327                 </xsd:element>
1328             </xsd:sequence>
1329             <xsd:sequence>
1330                 <xsd:element name="StartDate" type="udt:DateType"
1331                     minOccurs="0">
1332                     ...
1333                 </xsd:element>
1334                 <xsd:element name="EndDate" type="udt:DateType"
1335                     minOccurs="0">
1336                     ...
1337                 </xsd:element>
1338             </xsd:sequence>
1339             <xsd:sequence>
1340                 <xsd:element name="StartDateTime" type="udt:DateTimeType"
1341                     minOccurs="0">
1342                     ...
1343                 </xsd:element>
1344                 <xsd:element name="EndDateTime" type="udt:DateTimeType"
1345                     minOccurs="0">
1346                     ...
1347                 </xsd:element>
1348             </xsd:sequence>
1349         </xsd:choice>
1350     </xsd:sequence>
1351 </xsd:complexType>

```

1352 [R 98] The order and cardinality of the elements within an ABIE **xsd:complexType** MUST be according
1353 to the structure of the ABIE as defined in the model.

1354 **Example 4-11: Type definition of an ABIE**

```

1355 <!-- ===== Type Definitions ===== -->
1356 <!-- ===== -->
1357 <xsd:complexType name="AccountType" >

```

```

1358 <xsd:annotation>
1359   ... see annotation ...
1360 </xsd:annotation>
1361 <xsd:sequence>
1362   <xsd:element name="ID" type="udt:IdentifierType"
1363     minOccurs="0" maxOccurs="unbounded">
1364     <xsd:annotation>
1365       ... see annotation ...
1366     </xsd:annotation>
1367   </xsd:element>
1368   ... see element declaration ....
1369 </xsd:sequence>
1370 </xsd:complexType>
1371

```

1372 4.3.5 Element Declarations

1373 [R 99] For every attribute of an object class (BBIE) identified in the UN/CEFACT syntax-neutral model, a
1374 named **xsd:element** MUST be locally declared within the **xsd:complexType** representing the
1375 ABIE.

1376 [R 100] Each BBIE element name declaration MUST be based on the property term and qualifiers and
1377 the representation term of the basic business information entity (BBIE). If there are duplicate
1378 words in the property term and representation terms of the source dictionary entry name, then the
1379 duplicate words must be removed.

1380 [R 101] The BBIE element MUST be based on an appropriate data type that is defined in the
1381 UN/CEFACT **qdt:QualifiedDataType** or **udt:UnqualifiedDataType** schema modules.

1382 [R 102] For every association (ASBIE) identified in the UN/CEFACT syntax-neutral model, a named
1383 **xsd:element** MUST be locally declared within the **xsd:complexType** representing the ABIE.

1384 [R 103] Each ASBIE element name declaration MUST be based on the property term and object class of
1385 the association business information entity (ASBIE). If there are duplicate words in the property
1386 term and representation terms of the source dictionary entry name, then the duplicate words
1387 MUST be removed.

1388 [R 104] The element representing an association business information entity (ASBIE) MUST be of the
1389 complex type corresponding to its associated aggregate business information (ABIE).

1390 Example 4-12: Element declaration within an ABIE

```

1391 ... see type defintion ...
1392 <xsd:element name="ID" type="udt:IdentifierType"
1393   minOccurs="0" maxOccurs="unbounded">
1394   <xsd:annotation>
1395     ... see annotation ...
1396   </xsd:annotation>
1397 </xsd:element>
1398 <xsd:element name="Status" type="ram:StatusType"
1399   minOccurs="0" maxOccurs="unbounded">
1400   <xsd:annotation>
1401     ... see annotation ...
1402   </xsd:annotation>
1403 </xsd:element>
1404 <xsd:element name="Name" type="udt:NameType"
1405   minOccurs="0" maxOccurs="unbounded">
1406   <xsd:annotation>
1407     ... see annotation ...
1408   </xsd:annotation>
1409 </xsd:element>
1410 <xsd:element name="CurrencyCode" type="qdt:CurrencyCodeType"
1411   minOccurs="0" maxOccurs="unbounded">
1412   <xsd:annotation>
1413     ... see annotation ...
1414   </xsd:annotation>
1415 </xsd:element>

```

1416
1417

... see type definition ...

1418 4.3.6 Annotation

1419 [R 105] Each **xsd:complexType** and **xsd:element** definition MUST use the **xsd:annotation**
1420 element.

1421 [R 106] A **xsd:annotation** element declaration MUST appear immediately after the starting tag of the
1422 **xsd:complexType** or **xsd:element**.

1423 [R 107] For every ABIE **xsd:complexType** definition a structured set of annotations MUST be present
1424 in the following pattern:

- 1425 ▪ **UniqueID** (mandatory): The identifier that references an Aggregate Business
1426 Information Entity instance in a unique and unambiguous way.
- 1427 ▪ **CategoryCode** (mandatory): The category to which the object belongs. In this
1428 case the value will always be ABIE.
- 1429 ▪ **DictionaryEntryName** (mandatory): The official name of an Aggregate
1430 Business Information Entity.
- 1431 ▪ **VersionID** (mandatory): An indication of the evolution over time of an
1432 Aggregate Business Information Entity instance.
- 1433 ▪ **DefinitionText** (mandatory): The semantic meaning of an Aggregate Business
1434 Information Entity.
- 1435 ▪ **ObjectClassTermName** (mandatory): The Object Class Term of the
1436 associated Aggregate Core Component.
- 1437 ▪ **QualifierTermName** (optional): Qualifies the Object Class Term of the
1438 associated Aggregate Core Component.
- 1439 ▪ **UsageRuleText** (optional, repetitive): A constraint that describes specific
1440 conditions that are applicable to the Aggregate Business Information Entity.
- 1441 ▪ **BusinessTermName** (optional, repetitive): A synonym term under which the
1442 Aggregate Business Information Entity is commonly known and used in the
1443 business.
- 1444 ▪ **BusinessProcessContext** (optional, repetitive): The business process with
1445 which this Aggregate Business Information Entity is associated.
- 1446 ▪ **GeopoliticalorRegionContext** (optional, repetitive): The geopolitical/region
1447 contexts for this Aggregate Business Information Entity.
- 1448 ▪ **OfficialConstraintContext** (optional, repetitive): The official constraint
1449 context for this Aggregate Business Information Entity.
- 1450 ▪ **ProductContext** (optional, repetitive): The product context for this Aggregate
1451 Business Information Entity.
- 1452 ▪ **IndustryContext** (optional, repetitive): The industry context for this
1453 Aggregate Business Information Entity.
- 1454 ▪ **BusinessProcessRoleContext** (optional, repetitive): The role context for this
1455 Aggregate Business Information Entity.

1456
1457
1458
1459
1460
1461
1462

- SupportingRoleContext (optional, repetitive): The supporting role context for this Aggregate Business Information Entity.
- SystemCapabilitiesContext (optional, repetitive): The system capabilities context for this Aggregate Business Information Entity.
- Example (optional, repetitive): Example of a possible value of an Aggregate Business Information Entity.

1463 **Example 4-13: Example of annotation of an ABIE**

1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480

```
<xsd:complexType name="AccountType" >  
<xsd:annotation>  
  <xsd:documentation xml:lang="en">  
    <ccts:UniqueID>UN00000001</ccts:UniqueID>  
    <ccts:CategoryCode>ABIE</ccts:CategoryCode>  
    <ccts:DictionaryEntryName>Account. Details</ccts:DictionaryEntryName>  
    <ccts:VersionID>1.0</ccts:VersionID>  
    <ccts:Definition>A business arrangement whereby debits and/or credits arising  
      from transactions are recorded. This could be with a bank, i.e. a financial  
      account, or a trading partner offering supplies or services 'on account', i.e.  
      a commercial account</ccts:Definition>  
    <ccts:ObjectClassTermName>Account</ccts:ObjectClassTermName>  
  </xsd:documentation>  
</xsd:annotation>  
  ...  
</xsd:complexType>
```

1481 [R 108] For every BBIE **xsd:element** declaration a structured set of annotations MUST be present in
1482 the following pattern:

1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502

- UniqueID (mandatory): The identifier that references a Basic Business Information Entity instance in a unique and unambiguous way.
- CategoryCode (mandatory): The category to which the object belongs. In this case the value will always be BBIE.
- Dictionary Entry Name (mandatory): The official name of a Basic Business Information Entity.
- VersionID (mandatory): An indication of the evolution over time of a Basic Business Information Entity instance.
- Definition (mandatory): The semantic meaning of a Basic Business Information Entity.
- CardinalityText (mandatory): Indication whether the Basic Business Information Entity Property represents a not-applicable, optional, mandatory and/or repetitive characteristic of the Aggregate Business Information Entity.
- ObjectClassTermName (mandatory): The Object Class Term of the associated Aggregate Core Component.
- PropertyTermName (mandatory): The Property Term of the associated Aggregate Core Component.
- RepresentationTermName (mandatory): The Representation Term of the associated Core Component in the associated Aggregate Core Component.

1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527

- **QualifierTermName** (optional): Qualifies the Property Term of the associated Core Component Property in the associated Aggregate Core Component.
- **UsageRuleText** (optional, repetitive): A constraint that describes specific conditions that are applicable to the Basic Business Information Entity.
- **BusinessTermName** (optional, repetitive): A synonym term under which the Basic Business Information Entity is commonly known and used in the business.
- **BusinessProcessContext** (optional, repetitive): The business process with which this Business Information Entity is associated.
- **GeopoliticalorRegionContext** (optional, repetitive): The geopolitical/region contexts for this Business Information Entity.
- **OfficialConstraintContext** (optional, repetitive): The official constraint context for this Business Information Entity.
- **ProductContext** (optional, repetitive): The product context for this Business Information Entity.
- **IndustryContext** (optional, repetitive): The industry context for this Business Information Entity.
- **BusinessProcessRoleContext** (optional, repetitive): The role context for this Business Information Entity.
- **SupportingRoleContext** (optional, repetitive): The supporting role context for this Business Information Entity.
- **SystemCapabilitiesContext** (optional, repetitive): The system capabilities context for this Business Information Entity.
- **Example** (optional, repetitive): Example of a possible value of a Basic Business Information Entity.

1528 **Example 4-14: Annotation of a BBIE**

1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547

```
<xsd:element name="ID" type="udt:IdentifierType"
minOccurs="0" maxOccurs="unbounded">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      <ccts:UniqueID>UN00000002</ccts:UniqueID>
      <ccts:CategoryCode>BBIE</ccts:CategoryCode>
      <ccts:DictionaryEntryName>Account. Identifier</ccts:DictionaryEntryName>
      <ccts:VersionID>1.0</ccts:VersionID>
      <ccts:DefinitionText>The identification of a specific account.
      </ccts:DefinitionText>
      </ccts:CardinalityText>0..n</ccts:CardinalityText>
      <ccts:ObjectClassTermName>Account</ccts:ObjectClassTermName>
      <ccts:PropertyTermName>Identifier</ccts:PropertyTermName>
      <ccts:RepresentationTermName>Identifier</ccts:RepresentationTermName>
      <ccts:BusinessTermName>Account Number</ccts:BusinessTermName>
    </xsd:documentation>
  </xsd:annotation>
</xsd:element>
```

1548 [R 109] For every ASBIE **xsd:element** declaration a structured set of annotations MUST be present in
1549 the following pattern:

1550
1551

- **UniqueID** (mandatory): The identifier that references an Association Business Information Entity instance in a unique and unambiguous way.

- 1552 ▪ **CategoryCode (mandatory):** The category to which the object belongs. In this
1553 case the value will always be ASBIE.
- 1554 ▪ **DictionaryEntryName (mandatory):** The official name of an Association
1555 Business Information Entity.
- 1556 ▪ **VersionID (mandatory):** An indication of the evolution over time of an
1557 Association Business Information Entity instance.
- 1558 ▪ **DefinitionText (mandatory):** The semantic meaning of an Association
1559 Business Information Entity.
- 1560 ▪ **CardinalityText (mandatory):** Indication whether the Association Business
1561 Information Entity Property represents a not-applicable, optional, mandatory
1562 and/or repetitive characteristic of the Aggregate Business Information
1563 Entity.
- 1564 ▪ **ObjectClassTermName (mandatory):** The Object Class Term of the
1565 associated Aggregate Core Component.
- 1566 ▪ **PropertyTermName (mandatory):** The Property Term of the associated
1567 Aggregate Core Component.
- 1568 ▪ **AssociatedObjectClassTermName (mandatory):** The Associated Object Class
1569 Term of the associated Aggregate Core Component.
- 1570 ▪ **QualifierTermName (optional):** Qualifies the Property Term of the associated
1571 Core Component Property in the associated Aggregate Core Component.
- 1572 ▪ **UsageRuleText (optional, repetitive):** A constraint that describes specific
1573 conditions that are applicable to the Association Business Information
1574 Entity.
- 1575 ▪ **BusinessTermName (optional, repetitive):** A synonym term under which the
1576 Association Business Information Entity is commonly known and used in
1577 the business.
- 1578 ▪ **BusinessProcessContext (optional, repetitive):** The business process with
1579 which this Association Business Information Entity is associated.
- 1580 ▪ **GeopoliticalorRegionContext (optional, repetitive):** The geopolitical/region
1581 contexts for this Association Business Information Entity.
- 1582 ▪ **OfficialConstraintContext (optional, repetitive):** The official constraint
1583 context for this Association Business Information Entity.
- 1584 ▪ **ProductContext (optional, repetitive):** The product context for this
1585 Association Business Information Entity.
- 1586 ▪ **IndustryContext (optional, repetitive):** The industry context for this
1587 Association Business Information Entity.
- 1588 ▪ **BusinessProcessRoleContext (optional, repetitive):** The role context for this
1589 Association Business Information Entity.
- 1590 ▪ **SupportingRoleContext (optional, repetitive):** The supporting role context for
1591 this Association Business Information Entity.
- 1592 ▪ **SystemCapabilitiesContext (optional, repetitive):** The system capabilities
1593 context for this Association Business Information Entity.

1594
1595

▪ Example (optional, repetitive): Example of a possible value of an Association Business Information Entity.

1596 **Example 4-15: Annotation of an ASBIE**

1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615

```
<xsd:element name="Status" type="ram:StatusType"
minOccurs="0" maxOccurs="unbounded">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      <ccts:UniqueID>UN00000003</ccts:UniqueID>
      <ccts:CategoryCode>ASCC</ccts:CategoryCode>
      <ccts:DictionaryEntryName>Account. Status</ccts:DictionaryEntryName>
      <ccts:VersionID>1.0</ccts:VersionID>
      <ccts:DefinitionText>Associated status information related to account
        details.</ccts:DefinitionText>
      <ccts:CardinalityText>0..n</ccts:CardinalityText>
      <ccts:ObjectClassTermName>Account</ccts:ObjectClassTermName>
      <ccts:PropertyTermName>Status</ccts:PropertyTermName>
      <ccts:AssociatedObjectClassTermName>Status
      </ccts:AssociatedObjectClassTermName>
    </xsd:documentation>
  </xsd:annotation>
</xsd:element>
```

1616 **4.4 Core Component Type**

1617 **4.4.1 Use of Core Component Type Module**

1618 The purpose of the core component type module is to define the core component types on which the
1619 unqualified data types are based. There will only be a core component type defined for the primary
1620 representation term. This module is only for reference and will not be included/imported in any schema.
1621 The normative formatted schema for the CoreComponentType module is contained in Appendix F.

1622 **4.4.2 Schema Construct**

1623 The core component type schema module will be constructed in a standardized format in order to ensure
1624 consistency and ease of use. The specific format is shown below and must adhere to the format of the
1625 relevant sections as detailed in Appendix A.

1626 **Example 4-16: Structure of Core Component Type Schema Module**

1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652

```
<?xml version="1.0" encoding="utf-8"?>
<!-- =====>
<!-- ===== CCTS Core Component Types Schema Module =====>
<!-- =====>
<!--
Module of Core Component Types,
Agency: UN/CEFACT, OAGi,
Version: 0.3 Rev. 6,
Last change: 25. June 2004

Copyright (C) UN/CEFACT (2004). All Rights Reserved.

... see copyright information ...
-->
<xsd:schema
targetNamespace=
... see namespace ...
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified" attributeFormDefault="unqualified">
<!-- ===== Type Definitions =====>
<!-- ===== CCT: AmountType =====>
<!-- =====>
... see type definitions ...
</xsd:schema>
```

1653

1654 4.4.3 Namespace Scheme

1655 [R 110] The core component type (CCT) schema module MUST be represented by the token "cct".

1656 Example 4-17: Namespace of Core Component Type Schema Module

```
1657 "urn:un:unece:uncefact:documentation:draft:CCTS_CCT_SchemaModule:0:3:6"
```

1658 Example 4-18: Namespace of Core Component Type Schema Module

```
1659 <xsd:schema
1660   targetNamespace=
1661     "urn:un:unece:uncefact:documentation:draft:CCTS_CCT_SchemaModule:0:3:6"
1662   xmlns:cct=
1663     "urn:un:unece:uncefact:documentation:draft:CCTS_CCT_SchemaModule:0:3:6"
1664   xmlns:xsd="http://www.w3.org/2001/XMLSchema"
1665   elementFormDefault="qualified" attributeFormDefault="unqualified">
```

1666 4.4.4 Imports and Includes

1667 The core component types schema module does not import or include any other schema modules.

1668 4.4.5 Type Definitions

1669 [R 111] Every `cct:CoreComponentType` MUST be defined as a named `xsd:complexType` in the
1670 `cct:CoreComponentType` schema module.

1671 [R 112] The name of each `xsd:complexType` based on a `cct:CoreComponentType` MUST be the
1672 dictionary entry name of the core component type (CCT), with the separators and spaces
1673 removed.

1674 [R 113] Each `cct:CoreComponentType` `xsd:complexType` definition MUST contain one
1675 `xsd:simpleContent` element.

1676 [R 114] The `cct:CoreComponentType` `xsd:complexType` definition `xsd:simpleContent` element
1677 MUST contain one `xsd:extension` element. This `xsd:extension` element must include an
1678 `xsd` based attribute that defines the specific `xsd:built-in` data type required for the CCT
1679 content component.

1680 [R 115] Within the `cct:CoreComponentType` `xsd:extension` element a `xsd:attribute` MUST be
1681 declared for each supplementary component pertaining to that `cct:CoreComponentType`.

1682 Example 4-19: Type definition of a CCT

```
1683 <!-- ===== Type Definitions ===== -->
1684 <!-- ===== CCT: AmountType ===== -->
1685 <!-- ===== CCT: AmountType ===== -->
1686 <!-- ===== CCT: AmountType ===== -->
1687 <xsd:complexType name="AmountType">
1688   <xsd:annotation>
1689     ... see annotation ...
1690   </xsd:annotation>
1691   <xsd:simpleContent>
1692     <xsd:extension base="xsd:decimal">
1693       <xsd:attribute name="currencyID" type="xsd:token" use="optional"
1694       <xsd:annotation>
1695         ... see annotation ...
1696       </xsd:annotation>
1697     </xsd:attribute>
1698     ... see attribute declaration ...
1699   </xsd:extension>
1700 </xsd:simpleContent>
```

1701 `</xsd:complexType>`

1702

1703 4.4.6 Attribute Declarations

1704 [R 116] A `xsd:attribute` MUST be declared for each supplementary component.

1705 [R 117] Each supplementary component `xsd:attribute` "name" MUST be the supplementary
1706 component name with the separators and spaces removed.

1707 Example 4-10: Attribute of a supplementary component

```
1708 ... see type definition ...  
1709 <xsd:attribute name="currencyID" type="xsd:token" use="optional">  
1710 <xsd:annotation>  
1711 ... see annotation ...  
1712 </xsd:annotation>  
1713 </xsd:attribute>  
1714 <xsd:attribute name="currencyCodeListVersionID" type="xsd:token"  
1715 use="optional">  
1716 <xsd:annotation>  
1717 ... see annotation ...  
1718 </xsd:annotation>  
1719 </xsd:attribute>  
1720 ... see type definition ...
```

1721 4.4.7 Extension and Restriction

1722 The core component type schema module is a generic module that will be restricted in qualified and
1723 unqualified data type schema modules.

1724 4.4.8 Annotation

1725 [R 118] Each `xsd:complexType` definition MUST use the `xsd:annotation` element.

1726 [R 119] An `xsd:annotation` element declaration MUST appear immediately after the starting tag of the
1727 `xsd:complexType`.

1728 [R 120] For every `cct : CoreComponentType` `xsd:complexType` definition a structured set of
1729 annotations MUST be present in the following pattern:

- 1730 ▪ UniqueID (mandatory): The identifier that references the Core Component
1731 Type instance in a unique and unambiguous way.
- 1732 ▪ CategoryCode (mandatory): The category to which the object belongs. In this
1733 case the value will always be CCT.
- 1734 ▪ DictionaryEntryName (mandatory): The official name of a Core Component
1735 Type.
- 1736 ▪ VersionID (mandatory): An indication of the evolution over time of a Core
1737 Component Type instance.
- 1738 ▪ DefinitionText (mandatory): The semantic meaning of a Core Component
1739 Type.
- 1740 ▪ RepresentationTermName (mandatory): The primary representation term of
1741 the Core Component Type.

1742
1743
1744
1745
1746
1747
1748

- UsageRuleText (optional, repetitive): A constraint that describes specific conditions that are applicable to the Core Component Type.
- BusinessTermName (optional, repetitive): A synonym term under which the Core Component Type is commonly known and used in the business.
- Example (optional, repetitive): Example of a possible value of a Core Component Type.

1749

Example 4-11: Annotation of a CCT

1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763

```
... see type definition ...  
<xsd:annotation>  
  <xsd:documentation xml:lang="en">  
    <ccts:UniqueID>UNDT000001</ccts:UniqueID>  
    <ccts:CategoryCode>CCT</ccts:CategoryCode>  
    <ccts:DictionaryEntryName>Amount. Type</ccts:DictionaryEntryName>  
    </ccts:VersionID>1.0</ccts:VersionID>  
    <ccts:DefinitionText>A number of monetary units specified in a currency where the  
      unit of the currency is explicit or implied.</ccts:DefinitionText>  
    </ccts:RepresentationTermName>Amount</ccts:RepresentationTermName>  
  </xsd:documentation>  
</xsd:annotation>  
... see type definition ...
```

1764
1765

[R 121] For every supplementary component **xsd:attribute** declaration a structured set of annotations MUST be present in the following pattern:

1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786

- UniqueID (mandatory): The identifier that references a Supplementary Component instance in a unique and unambiguous way.
- CategoryCode (mandatory): The category to which the object belongs. In this case the value will always be SC.
- Dictionary Entry Name (mandatory): The official name of a Supplementary Component.
- VersionID (mandatory): An indication of the evolution over time of a Supplementary Component instance.
- DefinitionText (mandatory): The semantic meaning of a Supplementary Component.
- CardinalityText (mandatory): Indication whether the Supplementary Component Property represents a not-applicable, optional, mandatory and/or repetitive characteristic of the Core Component Type.
- PropertyTermName (mandatory): The Property Term of the associated Supplementary Component.
- RepresentationTermName (mandatory): The Representation term of the associated Supplementary Component.
- UsageRuleText (optional, repetitive): A constraint that describes specific conditions that are applicable to the Supplementary Core Component.
- Example (optional, repetitive): Example of a possible value of a Basic Core Component.

1787 **Example 4-12: Annotation of a supplementary component**

```
1788 ... see attribute declaration ...
1789 <xsd:annotation>
1790 <xsd:documentation xml:lang="en">
1791 <ccts:UniqueID>UNDT000001-SC2</ccts:UniqueID>
1792 <ccts:CategoryCode>SC</ccts:CategoryCode>
1793 <ccts:DictionaryEntryName>Amount. Currency. Identifier</ccts:DictionaryEntryName>
1794 <ccts:VersionID>1.0</ccts:VersionID >
1795 <ccts:DefinitionText>The currency of the amount.</ccts:DefinitionText>
1796 <ccts:CardinalityText>0..1</ccts:CardinalityText>
1797 <ccts:PropertyTermName>Currency</ccts:PropertyTermName>
1798 <ccts:RepresentationTermName>Identifier</ccts:RepresentationTermName>
1799 <ccts:UsageRuleText>Use 3 alpha codes UNECE Rec 9.</ccts:UsageRuleText>
1800 <ccts:UsageRuleText>It is recommended that the currency code is provided in this SC
1801 rather than in a separately defined BBIE.</ccts:UsageRuleText>
1802 </xsd:documentation>
1803 </xsd:annotation>
1804 ... see attribute declaration ...
```

1805 **4.5 Unqualified Data Type**

1806 **4.5.1 Use of Unqualified Data Type Module**

1807 The unqualified data type schema module will define data types for all primary and secondary
1808 representation terms as specified in the CCTS. All data types will be **xsd:complexType** or
1809 **xsd:simpleType** and will be without facet restrictions.

1810 **4.5.2 Schema Construct**

1811 The unqualified data types schema will be constructed in a standardized format in order to ensure
1812 consistency and ease of use. The specific format is shown below and must adhere to the format of the
1813 relevant sections as detailed in Appendix A.

1814 **Example 4-13: Structure of unqualified data type schema module**

```
1815 <?xml version="1.0" encoding="utf-8"?>
1816 <!-- =====>
1817 <!-- ===== UDT Unqualified Data Type Schema Module ===== -->
1818 <!-- =====>
1819 <!--
1820 Module of Unqualified Data Types,
1821 Agency: UN/CEFACT,
1822 Version: 0.3 Rev. 6,
1823 Last change: 25. June 2004
1824
1825 Copyright (C) UN/CEFACT (2004). All Rights Reserved.
1826
1827 ... see copyright information ...
1828
1829 -->
1830 <xsd:schema targetNamespace=
1831 ... see namespace ...
1832 xmlns:xsd="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
1833 attributeFormDefault="unqualified">
1834 <!-- ===== Imports ===== -->
1835 <!-- =====>
1836 ... see imports ...
1837 <!-- =====>
1838 <!-- ===== Type Definitions ===== -->
1839 <!-- =====>
1840 <!-- ===== Primary RT: Amount. Type ===== -->
1841 <!-- =====>
1842 <xsd:complexType name="AmountType">
1843 ... see type definition ...
1844 </xsd:complexType>
1845 ...
1846 </xsd:schema>
```

1847
1848

1849 4.5.3 Namespace Scheme

1850 [R 122] The unqualified data types schema module namespace MUST be represented by the token "udt".

1851 Example 4-14: Namespace of unqualified data types scheme module

```
1852 "urn:un:unece:unefact:data:draft:UnqualifiedDataTypesSchemaModule:0:3:6"
```

1853 Example 4-15: Schema-element of unqualified data types scheme module

```
1854 <xsd:schema
1855   targetNamespace=
1856   "urn:un:unece:unefact:data:draft:UnqualifiedDataTypesSchemaModule:0:3:6"
1857   xmlns:udt=
1858   "urn:un:unece:unefact:data:draft:UnqualifiedDataTypesSchemaModule:0:3:6"
1859   xmlns:xsd="http://www.w3.org/2001/XMLSchema"
1860   elementFormDefault="qualified" attributeFormDefault="unqualified">
```

1861 4.5.4 Imports and Includes

1862 [R 123] The `udt:UnqualifiedDataType` schema MUST import the following schema modules:

- 1863 – `ids:IdentifierList` schema modules
 - 1864 – `clm:CodeList` schema modules
-

1865 Example 4-16: Imports

```
1866 <!-- ===== Imports ===== -->
1867 <!-- ===== -->
1868 <!-- ===== Imports of Code Lists ===== -->
1869 <!-- ===== -->
1870 <xsd:import namespace=
1871   "urn:un:unece:unefact:codelist:draft:UNECE_Calculation_Amount:6:1"
1872   schemaLocation="http://www.unece.org/unefact/codelist/UNECE_Calculation_Amount_6.1_dr
1873   aft.xsd"/>
1874 <!-- ===== Imports of Identifier Lists ===== -->
1875 <!-- ===== -->
1876 <xsd:import namespace=
1877   "urn:un:unece:unefact:codelist:draft:ISO_Country_Identifier:2:4"
1878   schemaLocation="http://www.unece.org/unefact/codelist/ISO_Country_Identifier_2.4_draf
1879   t.xsd"/>
1880
```

1881 4.5.5 Type Definitions

1882 Each unqualified data type is represented in the unqualified data type schema module as either a
1883 `xsd:complexType` or a `xsd:simpleType`. Unqualified data types are defined based on the core
1884 component types as defined in the CCTS.

1885 [R 124] A `udt:UnqualifiedDataType` MUST be defined for each approved primary and secondary
1886 representation terms identified in the CCTS Permissible Representation Terms table.

1887 [R 125] The name of each `udt:UnqualifiedDataType` MUST be the dictionary entry name of the
1888 primary or secondary representation term, with "Type" at the end and the separators and spaces
1889 removed.

1890 In accordance with rules and principles in this document, the unqualified data types will be based on
1891 `xsd:built-in` data types whenever the facets of the `xsd:built-in` data type are equivalent to the
1892 supplementary components for that data type.

1893 [R 126] For every `udt:UnqualifiedDataType` whose supplementary components map directly to the
1894 properties of a built-in `xsd:data type`, the `udt:UnqualifiedDataType` MUST be defined
1895 as a named `xsd:simpleType` in the `udt:UnqualifiedDataType` schema module.

1896 [R 127] Every `udt:UnqualifiedDataType` defined as a `xsd:simpleType` MUST contain one
1897 `xsd:restriction` element. This `xsd:restriction` element MUST include an `xsd:base`
1898 attribute that defines the specific built-in `xsd:datatype` required for the content component.

1899 When the unqualified data type does not directly map to an `xsd:simpleType` due to the supplementary
1900 components needing to be expressed, it will be defined as an `xsd:complexType`.

1901 [R 128] For every `udt:UnqualifiedDataType` whose supplementary components are not equivalent to
1902 the properties of a `xsd:built-in` data type, a `udt:UnqualifiedDataType` MUST be defined
1903 as an `xsd:complexType` in the `udt:UnqualifiedDataType` schema module.

1904 [R 129] Every `udt:UnqualifiedDataType` `xsd:complexType` definition MUST contain one
1905 `xsd:simpleContent` element.

1906 [R 130] Every `udt:UnqualifiedDataType` `xsd:complexType` `xsd:simpleContent` element
1907 MUST contain one `xsd:extension` element. This `xsd:extension` element must include an
1908 `xsd:base` attribute that defines the specific `xsd:built-in` data type required for the content
1909 component.

1910 Each core component supplementary component will normally be declared as an attribute of the complex
1911 type. However, the namespace scheme for code lists and identification scheme lists has been designed
1912 to include some of the supplementary components for the CCTs Code. Type and Identifier. Type. Thus
1913 those attributes that are included in the namespace will not be declared as part of the unqualified data
1914 type.

1915 [R 131] Within the `udt:UnqualifiedDataType` `xsd:complexType` `xsd:extension` element an
1916 `xsd:attribute` MUST be declared for each supplementary component pertaining to the
1917 underlying CCT, unless the attribute is contained in the namespace declaration.

1918 For some CCT's, the CCTS identifies restrictions in the form of pointing to certain restrictive code or
1919 identifier lists. These restrictive lists will be declared in the code list or identifier schema module and the
1920 unqualified data type will reference these.

1921 [R 132] Each CCT code type or CCT identifier type supplementary component that has a restriction
1922 identified in CCTS, MUST have the name of the CCT supplementary component with the primary
1923 representation term removed at the beginning. The `xsd:type` for this `xsd:attribute` must be
1924 the name of the code list or identifier scheme containing the restricted set of values. The
1925 `xsd:attribute` must contain an `xsd:use` attribute who's value must be set to required.

1926 Example 4-17: Type definitions of unqualified data types

```
1927 <!-- ===== Type Definitions ===== -->  
1928 <!-- ===== Primary RT: Amount. Type ===== -->  
1929 <!-- ===== Primary RT: Amount. Type ===== -->  
1930 <!-- ===== Primary RT: Amount. Type ===== -->  
1931 <xsd:complexType name="AmountType">  
1932 <xsd:annotation>  
1933 ... see annotation ...  
1934 </xsd:annotation>  
1935 <xsd:simpleContent>  
1936 <xsd:extension base="xsd:decimal">  
1937 <xsd:attribute name="currencyID"  
1938 type="clm54217:CurrencyCodeContentType" use="required">  
1939 <xsd:annotation>  
1940 ... see annotation ...  
1941 </xsd:annotation>  
1942 </xsd:attribute>  
1943 <xsd:attribute name="currencyCodeListVersionID" type="xsd:token" use="optional">  
1944 <xsd:annotation>  
1945 ... see annotation ...  
1946 </xsd:annotation>  
1947 </xsd:attribute>  
1948 </xsd:extension>  
1949 </xsd:simpleContent>  
1950 </xsd:complexType>  
1951 <!-- ===== Primary RT: Binary Object. Type ===== -->  
1952 <!-- ===== Primary RT: Binary Object. Type ===== -->
```

```

1953 <xsd:complexType name="BinaryObjectType">
1954 <xsd:annotation>
1955   ... see annotation ...
1956 </xsd:annotation>
1957 <xsd:simpleContent>
1958   <xsd:extension base="xsd:base64Binary">
1959     <xsd:attribute name="binaryObjectMimeType"
1960       type="clmIANAMIMEMediaTypes:BinaryObjectMimeTypeContent"
1961     >
1962       <xsd:annotation>
1963         ... see annotation ...
1964       </xsd:annotation>
1965     </xsd:attribute>
1966     <xsd:attribute name="binaryObjectEncodingCode" type="xsd:token" use="optional">
1967       <xsd:annotation>
1968         ... see annotation ...
1969       </xsd:annotation>
1970     </xsd:attribute>
1971     <xsd:attribute name="binaryObjectCharacterSetCode" type="xsd:token" use="optional">
1972       <xsd:annotation>
1973         ... see annotation ...
1974       </xsd:annotation>
1975     </xsd:attribute>
1976     <xsd:attribute name="binaryObjectURI" type="xsd:anyURI" use="optional">
1977       <xsd:annotation>
1978         ... see annotation ...
1979       </xsd:annotation>
1980     </xsd:attribute>
1981     <xsd:attribute name="binaryObjectFilename" type="xsd:token" use="optional">
1982       <xsd:annotation>
1983         ... see annotation ...
1984       </xsd:annotation>
1985     </xsd:attribute>
1986 </xsd:extension>
1987 </xsd:simpleContent>
1988 </xsd:complexType>

```

1988 4.5.6 Attribute Declarations

1989 [R 133] Each supplementary component **xsd:attribute** name MUST be the supplementary
1990 component name with the separators and spaces removed.

1991 Example 4-18: Attribute declarations of unqualified data types

```

1992 <xsd:simpleContent>
1993   <xsd:restriction base="cct:AmountType">
1994     <xsd:attribute name="currencyID"
1995       type="clm54217:CurrencyCodeContent" use="required">
1996     <xsd:annotation>
1997       ... see annotation ...
1998     </xsd:annotation>
1999   </xsd:attribute>
2000 </xsd:restriction>
2001 </xsd:simpleContent>

```

2002

2003 4.5.7 Restriction

2004 The unqualified data types can be further restricted in the qualified data type module.

2005 4.5.8 Annotation

2006 [R 134] Each **udt:UnqualifiedDataType** **xsd:complexType** and **xsd:simpleType** definition
2007 MUST use the **xsd:annotation** element.

2008 [R 135] An **xsd:annotation** element declaration MUST appear immediately after the starting tag of the
2009 **xsd:complexType** or **xsd:simpleType**.

2010 [R 136] For every `udt:UnqualifiedDataType` `xsd:complexType` or `xsd:simpleType` definition a
2011 structured set of annotations MUST be present in the following pattern:

- 2012 ▪ **UniqueID** (mandatory): The identifier that references an Unqualified Data
2013 Type instance in a unique and unambiguous way.
- 2014 ▪ **CategoryCode** (mandatory): The category to which the object belongs. In this
2015 case the value will always be UDT.
- 2016 ▪ **DictionaryEntryName** (mandatory): The official name of an Unqualified Data
2017 Type.
- 2018 ▪ **VersionID** (mandatory): An indication of the evolution over time of
2019 anUnqualified Data Type instance.
- 2020 ▪ **DefinitionText** (mandatory): The semantic meaning of an Unqualified Data
2021 Type.
- 2022 ▪ **RepresentationTermName** (mandatory): The primary or secondary
2023 representation term of the associated Core Component Type.
- 2024 ▪ **UsageRuleText** (optional, repetitive): A constraint that describes specific
2025 conditions that are applicable to the Unqualified Data Type.
- 2026 ▪ **BusinessTermName** (optional, repetitive): A synonym term under which the
2027 Unqualified Data Type is commonly known and used in the business.
- 2028 ▪ **Example** (optional, repetitive): Example of a possible value of an Unqualified
2029 Data Type.

2030 **Example 4-19: Annotation of unqualified type definition**

```
2031     ... see complex type definition ...  
2032     <xsd:annotation>  
2033     <xsd:documentation xml:lang="en">  
2034         <ccts:UniqueID>UNDT000001</ccts:UniqueID>  
2035         <ccts:CategoryCode>UDT</ccts:CategoryCode>  
2036         <ccts:DictionaryEntryName>Amount. Type</ccts:DictionaryEntryName>  
2037         <ccts:VersionID>1.0</ccts:VersionID>  
2038         <ccts:DefinitionText> A number of monetary units specified in a currency where the  
2039         unit of the currency is explicit or implied.</ccts:DefinitionText>  
2040         <ccts:RepresentationTermName>Amount</ccts:RepresentationTermName>  
2041     </xsd:documentation>  
2042     </xsd:annotation>  
2043     ... see complex type definition ...
```

2044

2045 [R 137] For every supplementary component `xsd:attribute` declaration a structured set of
2046 annotations MUST be present in the following pattern:

- 2047 ▪ **UniqueID** (mandatory): The identifier that references a Supplementary
2048 Component of a Core Component Type instance in a unique and
2049 unambiguous way.
- 2050 ▪ **CategoryCode** (mandatory): The category to which the object belongs. In this
2051 case the value will always be SC.
- 2052 ▪ **Dictionary Entry Name** (mandatory): The official name of a Supplementary
2053 Component.
- 2054 ▪ **VersionID** (mandatory): An indication of the evolution over time of a
2055 Supplementary Component instance.

2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068

- **DefinitionText (mandatory):** The semantic meaning of a Supplementary Component.
- **CardinalityText (mandatory):** Indication whether the Supplementary Component Property represents a not-applicable, optional, mandatory and/or repetitive characteristic of the Core Component Type.
- **PropertyTermName (mandatory):** The Property Term of the associated Supplementary Component.
- **RepresentationTermName (mandatory):** The Representation Term of the associated Supplementary Component.
- **UsageRuleText (optional, repetitive):** A constraint that describes specific conditions that are applicable to the Supplementary Component.
- **Example (optional, repetitive):** Example of a possible value of a Supplementary Component.

2069

Example 4-20: Annotation of a supplementary component

2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2090

```
... see complex type definition ...  
<xsd:attribute name="currencyID" type="iso4217:CurrencyCodeContentType"  
use="required">  
<xsd:annotation>  
  <xsd:documentation xml:lang="en">  
    <ccts:UniqueID>UNDT000001-SC2</ccts:UniqueID>  
    <ccts:CategoryCode>SC</ccts:CategoryCode>  
    <ccts:DictionaryEntryName>Amount. Currency. Identifier  
    </ccts:DictionaryEntryName>  
    <ccts:VersionID>1.0</ccts:VersionID>  
    <ccts:DefinitionText>The currency of the amount.</ccts:DefinitionText>  
    <ccts:CardinalityText>0..1</ccts:CardinalityText>  
    <ccts:PropertyTermName>Currency</ccts:PropertyTermName>  
    <ccts:RepresentationTermName>Identifier</ccts:RepresentationTermName>  
    <ccts:UsageRuleText>Use 3 alpha codes UNECE Rec 9.</ccts:UsageRuleText>  
    <ccts:UsageRuleText>It is recommended that the currency code is provided in this  
    SC rather than in a separately defined BBIE.</ccts:UsageRuleText>  
  </xsd:documentation>  
</xsd:annotation>  
</xsd:attribute>  
... see complex type definition ...
```

2091

2092

4.6 Qualified Data Type

2093
2094
2095
2096
2097
2098
2099

The data types defined in the unqualified data types schema module are **xsd:complexType** or **xsd:simpleTypes** without facet restrictions. These types are intended to be suitable as the **xsd:base** type for some but not all expected complex types. As business process modeling reveals the need for specialized data types, new 'qualified' types will need to be defined. These new 'qualified' data types must be restricted technically by facets and semantically by one or more additional qualifier terms. These 'qualified' data types will be defined using **xsd:restriction** on the unqualified data types or can be based on new specific simple types.

2100

4.6.1 Use of Qualified Data Type Module

2101
2102
2103
2104
2105

To ensure consistency of qualified data types with the UN/CEFACT modularity and reuse goals will require creating a single schema module that defines all qualified data types. The qualified data type schema module name must follow the UN/CEFACT module naming approach. The qualified data type schema module will be used by the reusable ABIE schema module and all root schema modules.

2106 4.6.2 Schema Construct

2107 The qualified data type schema will be constructed in a standardized format in order to ensure
2108 consistency and ease of use. The specific format is shown below and must adhere to the format of the
2109 relevant sections as detailed in Appendix A.
2110

```
2111 <?xml version="1.0" encoding="utf-8"?>
2112 <!-- edited with XMLSPY v2004 rel. 3 U (http://www.xmlspy.com) by Gunther Stuhec (UBL)
2113 -->
2114 <!-- =====
2115 <!-- ===== QDT Qualified Data Type Schema Module ===== -->
2116 <!-- =====
2117 <!--
2118     Module of   Qualified Data Types,
2119     Agency:    UN/CEFACT, OAGi,
2120     Version:   0.3 Rev. 6,
2121     Last change: 25. June 2004
2122
2123
2124     Copyright (C) UN/CEFACT (2004). All Rights Reserved.
2125
2126     ... see copyright information ...
2127
2128 -->
2129 <xsd:schema targetNamespace=
2130     ... see namespace ...
2131     xmlns:xsd="http://www.w3.org/2001/XMLSchema"
2132     elementFormDefault="qualified" attributeFormDefault="unqualified">
2133 <!-- ===== Imports ===== -->
2134 <!-- =====
2135     ... see imports ...
2136 <!-- ===== Type Definitions ===== -->
2137 <!-- =====
2138     ... see type definitions ...
2139 </xsd:schema>
```

2140

2141 4.6.3 Namespace Scheme

2142 [R 138] The UN/CEFACT:QualifiedDataType schema module namespace MUST be represented by the
2143 token "qdt".

2144 Example 4-21: Namespace name

```
2145 "urn:un:unece:unefact:data:draft:QualifiedDataTypeSchemaModule:0:3:6"
```

2146

2147 Example 4-21: Schema element

```
2148 <xsd:schema targetNamespace="urn:un:unece:unefact:data:draft:
2149     QualifiedDataTypeSchemaModule:0:3:6"
2150     xmlns:udt="urn:un:unece:unefact:data:draft:
2151     UnqualifiedDataTypeSchemaModule:0:3:6"
2152     xmlns:qdt="urn:un:unece:unefact:data:draft:
2153     QualifiedDataTypeSchemaModule:0:3:6"
2154     xmlns:xsd="http://www.w3.org/2001/XMLSchema"
2155     elementFormDefault="qualified" attributeFormDefault="unqualified">
```

2156

2157 4.6.4 Imports and Includes

2158 Qualified data types will be derived from data types defined in the unqualified data types, code list, and
2159 identifier list schema modules. These schema modules reside in their own namespaces and as such will
2160 be imported rather than included in the qualified data types schema module.

2161 [R 139] The `qdt:QualifiedDataType` schema module MUST import the
2162 – `udt:UnqualifiedDataType` schema module
2163 – `ids:IdentifierList` schema modules
2164 – `clm:CodeList` schema module

2165 Note: If needed, relevant UN/CEFACT and external code list and identifier scheme
2166 schema modules may be imported.

```
2167 <!-- ===== Imports ===== -->
2168 <!-- ===== -->
2169 <!-- ===== Import of Unqualified Data Type Schema Module (UDT) ===== -->
2170 <!-- ===== -->
2171 <xsd:import
2172 namespace="urn:un:unece:uncefact:data:draft:UnqualifiedDataTypeSchemaModule:0:3:6"
2173 schemaLocation="http://www.unece.org/uncefact/data/
2174 UnqualifiedDataTypeSchemaModule_0.3.6_draft.xsd"/>
2175 <!-- ===== Imports of Code Lists ===== -->
2176 <!-- ===== -->
2177 <xsd:import namespace=
2178 "urn:un:unece:uncefact:codelist:draft:UNECE_Calculation_Amount:6:1"
2179 schemaLocation="http://www.unece.org/uncefact/codelist/UNECE_Calculation_Amount_6.1_dr
2180 aft.xsd"/>
2181 <!-- ===== Imports of Identifier Lists ===== -->
2182 <!-- ===== -->
2183 <xsd:import namespace=
2184 "urn:un:unece:uncefact:codelist:draft:ISO_Country_Identifier:2:4"
2185 schemaLocation="http://www.unece.org/uncefact/codelist/ISO_Country_Identifier_2.4_draf
2186 t.xsd"/>
2187
```

2188 4.6.5 Type Definitions

2189 [R 140] Where required to change facets of an existing `udt:UnqualifiedDataType`, a new data type
2190 MUST be defined in the `qdt:QualifiedDataType` schema module.

2191 [R 141] If a `qdt:QualifiedDataType` is based on an `udt:UnqualifiedDataType` that is an
2192 `xsd:simpleType`, the `qdt:QualifiedDataType` MUST also be an `xsd:simpleType`.

2193 [R 142] If a `qdt:QualifiedDataType` is based on an `udt:UnqualifiedDataType` that is a
2194 `xsd:complexType`, the `qdt:QualifiedDataType` MUST also be an `xsd:complexType`.

2195 [R 143] If a `qdt:QualifiedDataType`'s restrictions map directly to the properties of a `xsd:built-`
2196 `in` data type, the `qdt:QualifiedDataType` MUST be defined as a named
2197 `xsd:simpleType` in the `qdt:QualifiedDataType` schema module.

2198 [R 144] The name of a `qdt:QualifiedDataType` MUST be the dictionary entry name of its base
2199 `udt:UnqualifiedDataType` with an appropriate qualifier or qualifiers.

2200 [R 145] Each `qdt:QualifiedDataType` `xsd:complexType` definition MUST contain one
2201 `xsd:simpleContent` element.

2202 [R 146] The `qdt:QualifiedDataType` `xsd:complexType` definition `xsd:simpleContent` element
2203 MUST contain one `xsd:restriction` element with an `xsd:base` attribute whose value is
2204 equal to an `udt:UnqualifiedDataType` or an externally imported code list.

2205 NOTE: If a non-standard variation of the standard date time built-in data types are
2206 required, for example year month, then a qualified data type of textType needs to be
2207 defined, with the appropriate restriction specified, e.g. as a pattern, to specify the
2208 required format.

```
2209 <!-- ===== Type Definitions ===== -->
2210 <!-- ===== -->
2211 <!-- ===== Qualified Data Type based on DateTime Type ===== -->
```

```

2212 <!-- ===== Qualified DT: Day_ Date. Type ===== -->
2213 <!-- ===== Qualified DT: Day_ Date. Type ===== -->
2214 <!-- ===== Qualified DT: Day_ Date. Type ===== -->
2215 <xsd:simpleType name="DayDateType">
2216 <xsd:annotation>
2217 ... see annotation ...
2218 </xsd:annotation>
2219 <xsd:restriction base="xsd:gDay"/>
2220 </xsd:simpleType>
2221 ...
2222 <!-- ===== Qualified Data Type based on Text. Type ===== -->
2223 <!-- ===== Qualified Data Type based on Text. Type ===== -->
2224 <!-- ===== Qualified DT: Description_ Text. Type ===== -->
2225 <!-- ===== Qualified DT: Description_ Text. Type ===== -->
2226 <xsd:complexType name="DescriptionTextType">
2227 <xsd:annotation>
2228 ... see annotation ...
2229 </xsd:annotation>
2230 <xsd:simpleContent>
2231 <xsd:restriction base="udt:TextType"/>
2232 </xsd:simpleContent>
2233 </xsd:complexType>
2234 ...
2235 <!-- ===== Qualified Data Type based on Identifier. Type ===== -->
2236 <!-- ===== Qualified Data Type based on Identifier. Type ===== -->
2237 <!-- ===== Qualified DT: Uniform Resource_ Identifier. Type ===== -->
2238 <!-- ===== Qualified DT: Uniform Resource_ Identifier. Type ===== -->
2239 <xsd:simpleType name="UniformResourceIdentifierType">
2240 <xsd:annotation>
2241 ... see annotation ...
2242 </xsd:annotation>
2243 <xsd:restriction base="xsd:anyURI"/>
2244 </xsd:simpleType>
2245 ...
2246 <!-- ===== Qualified DT: Country_ Identifier. Type ===== -->
2247 <!-- ===== Qualified DT: Country_ Identifier. Type ===== -->
2248 <xsd:simpleType name="CountryIdentifierType">
2249 <xsd:annotation>
2250 ... see annotation ...
2251 </xsd:annotation>
2252 <xsd:restriction base="ids53166:CountryCodeContentType"/>
2253 </xsd:simpleType>
2254 ...

```

2255 4.6.6 Attribute and Element Declarations

2256 There will be no element declarations in the qualified data type schema module. Attribute names will
2257 appear in the qualified data type as defined in the unqualified data type schema module with further
2258 restrictions applied as required.
2259

2260 4.6.7 Extension and Restriction

2261 [R 147] The `qdt:QualifiedDataType` `xsd:complexType` definition `xsd:simpleContent` element
2262 MUST only restrict attributes declared in its base type, or must only restrict facets equivalent to
2263 allowed supplementary components.

2264 Example 4-22: Qualified Data Type Restricting an Identification Scheme

```

2265 <xsd:complexType name="PartyIdentifierType">
2266 <xsd:annotation>
2267 ... see annotation ...
2268 </xsd:annotation>
2269 <xsd:simpleContent>
2270 <xsd:restriction base="udt:IdentifierType">
2271 <xsd:attribute name="schemeName" use="prohibited"/>
2272 <xsd:attribute name="schemeAgencyName" use="prohibited"/>
2273 <xsd:attribute name="schemeVersionID" use="prohibited"/>
2274 <xsd:attribute name="schemeDataURI" use="prohibited"/>
2275 </xsd:restriction>
2276 </xsd:simpleContent>

```


2277

</xsd:complexType>

2278 4.6.8 Annotation

2279 [R 148] Each `qdt:QualifiedDataType` `xsd:complexType` and `xsd:simpleType` definition MUST
2280 use the `xsd:annotation` element.

2281 [R 149] An `xsd:annotation` element declaration MUST appear immediately after the starting tag of the
2282 `xsd:complexType` or `xsd:simpleType`.

2283 [R 150] Every `qdt:QualifiedDataType` definition MUST contain a structured set of annotations in the
2284 following sequence and pattern:
2285

- 2286 ▪ **UniqueID** (mandatory): The identifier that references a Qualified Data Type
2287 instance in a unique and unambiguous way.
- 2288 ▪ **CategoryCode** (mandatory): The category to which the object belongs. In this
2289 case the value will always be QDT.
- 2290 ▪ **DictionaryEntryName** (mandatory): The official name of a Qualified Data
2291 Type.
- 2292 ▪ **VersionID** (mandatory): An indication of the evolution over time of a
2293 Qualified Data Type instance.
- 2294 ▪ **DefinitionText** (mandatory): The semantic meaning of a Qualified Data Type.
- 2295 ▪ **RepresentationTermName** (mandatory): The Representation Term of the
2296 associated Unqualified Data Type.
- 2297 ▪ **Qualifier Term** (mandatory): Qualifies the Representation Term in order to
2298 differentiate it from its underlying Core Component Type and other Data
2299 Types.
- 2300 ▪ **UsageRuleText** (optional, repetitive): A constraint that describes specific
2301 conditions that are applicable to the Qualified Data Type.
- 2302 ▪ **BusinessTermName** (optional, repetitive): A synonym term under which the
2303 Qualified Data Type is commonly known and used in the business.
- 2304 ▪ **BusinessProcessContext** (optional, repetitive): The business process with
2305 which this Qualified Data Type is associated.
- 2306 ▪ **GeopoliticalorRegionContext** (optional, repetitive): The geopolitical/region
2307 contexts for this Qualified Data Type.
- 2308 ▪ **OfficialConstraintContext** (optional, repetitive): The official constraint
2309 context for this Qualified Data Type.
- 2310 ▪ **ProductContext** (optional, repetitive): The product context for this Qualified
2311 Data Type.
- 2312 ▪ **IndustryContext** (optional, repetitive): The industry context for this Qualified
2313 Data Type.
- 2314 ▪ **BusinessProcessRoleContext** (optional, repetitive): The role context for this
2315 Qualified Data Type.

2316
2317

- SupportingRoleContext (optional, repetitive): The supporting role context for this Qualified Data Type.

2318
2319

- SystemCapabilitiesContext (optional, repetitive): The system capabilities context for this Qualified Data Type.

2320
2321

- Example (optional, repetitive): Example of a possible value of a Qualified Data Type.

2322
2323

Example 4-23: Annotation of qualified data types

2324
2325
2326
2327
2328
2329
2330
2331
2332
2333
2334
2335
2336
2337
2338

```
... see type definition ...  
<xsd:annotation>  
<xsd:documentation xml:lang="en">  
<ccts:UniqueID/>  
<ccts:CategoryCode>QDT</ccts:CategoryCode>  
<ccts:DictionaryEntryName>Account_ Type_ Code. Type</ccts:DictionaryEntryName>  
<ccts:VersionID>1.0</ccts:VersionID>  
<ccts:DefinitionText> This code represents the type of an account.  
</ccts:DefinitionText>  
<ccts:RepresentationTermName>Code</ccts:RepresentationTermName>  
<ccts:ObjectClassQualifier>Account</ccts:ObjectClassQualifier>  
<ccts:ObjectClassQualifier>Type</ccts:ObjectClassQualifier>  
</xsd:documentation>  
</xsd:annotation>  
... see type definition ...
```

2339
2340

[R 151] For every supplementary component **xsd:attribute** declaration a structured set of annotations MUST be present in the following pattern:

2341
2342
2343

- UniqueID (mandatory): The identifier that references a Supplementary Component of a Core Component Type instance in a unique and unambiguous way.

2344
2345

- CategoryCode (mandatory): The category to which the object belongs. In this case the value will always be QDT.

2346
2347

- Dictionary Entry Name (mandatory): The official name of a Supplementary Component.

2348
2349

- VersionID (mandatory): An indication of the evolution over time of a Supplementary Component instance.

2350
2351

- DefinitionText (mandatory): The semantic meaning of a Supplementary Component.

2352
2353
2354

- CardinalityText (mandatory): Indication whether the Supplementary Component Property represents a not-applicable, optional, mandatory and/or repetitive characteristic of the Core Component Type.

2355
2356

- PropertyTermName: The Property Term of the associated Supplementary Component.

2357
2358

- RepresentationTermName: The Representation Term of the associated Supplementary Component.

2359
2360

- UsageRuleText (optional, repetitive): A constraint that describes specific conditions that are applicable to the Supplementary Component.

2361
2362

- BusinessProcessContext (optional, repetitive): The business process with which this Supplementary Component is associated.

2363
2364
2365
2366
2367
2368
2369
2370
2371
2372
2373
2374
2375
2376
2377
2378

- GeopoliticalorRegionContext (optional, repetitive): The geopolitical/region contexts for this Supplementary Component.
- OfficialConstraintContext (optional, repetitive): The official constraint context for this Supplementary Component.
- ProductContext (optional, repetitive): The product context for this Supplementary Component.
- IndustryContext (optional, repetitive): The industry context for this Supplementary Component.
- BusinessProcessRoleContext (optional, repetitive): The role context for this Qualified Data Type.
- SupportingRoleContext (optional, repetitive): The supporting role context for this Supplementary Component.
- SystemCapabilitiesContext (optional, repetitive): The system capabilities context for this Supplementary Component.
- Example (optional, repetitive): Example of a possible value of a Supplementary Component.

2379
2380

4.7 Code Lists

2381
2382

[R 152] Internal code lists MUST NOT duplicate existing external code lists where they are available in a schema module form that is capable of being imported.

2383
2384

External code lists must be used when they exist in schema module form and when they can be directly imported into a schema module.

2385
2386
2387
2388

UN/CEFACT *may* design and use an internal code list where an existing external code list needs to be extended, or where no suitable external code list exists. If a code list is created, the lists should be globally scoped and designed for reuse and sharing.

2389
2390

[R 153] Each UN/CEFACT maintained code list MUST be defined in its own schema module.

2391

4.7.1 Schema Construct

2392
2393
2394

The code list schema module will follow the general pattern for all UN/CEFACT schema modules. Following the generic module information, the body of the schema will consist of code list definitions of the following general form:

2395

Example 4-24: Structure of code lists

2396
2397
2398
2399
2400
2401
2402
2403
2404
2405
2406
2407
2408
2409
2410

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- ===== Code List: Account Type Code ; UNECE ===== -->
<!-- ===== Code List: Account Type Code ; UNECE ===== -->
<!-- ===== Code List: Account Type Code ; UNECE ===== -->
<!--
    Codelist of Account Type Code,
    Agency: UNECE,
    Version: D.01C
    Last change: 25. June 2004

    Copyright (C) UN/CEFACT (2004). All Rights Reserved.

    ... see copyright information ...
-->
```

```

2411 <xsd:schema targetNamespace=" ... see namespace ...
2412 xmlns:xsd="http://www.w3.org/2001/XMLSchema"
2413 elementFormDefault="qualified" attributeFormDefault="unqualified">
2414 <!-- ===== Root Element ===== -->
2415 <!-- =====
2416 ... see root element declaration ...
2417 <!-- ===== Type Definitions ===== -->
2418 <!-- =====
2419 <!-- ===== Code List Type Definition: Account Type Code ===== -->
2420 <!-- =====
2421 ... see type definition ...
2422 </xsd:schema>

```

2423

2424 4.7.2 Namespace Scheme

2425 In order to allow for uniqueness with a reasonably short token it is recommended that the token
 2426 representing the namespace for code lists is constructed based on the identifier of the agency
 2427 maintaining the code list and the identifier for the specific code list as issued by the maintenance agency.
 2428 In cases where there is no identifier, the name for the agency and/or code list should be used instead.
 2429 This will typically be true when proprietary code lists are used.

2430 [R 154] Each UN/CEFACT maintained code list schema module MUST be represented by a unique token
 2431 constructed as follows:
 2432 clm[Qualified data type name]<Code List. Agency. Identifier|Code List.
 2433 Agency Name. Text><Code List. Identification. Identifier|Code List.
 2434 Name. Text>
 2435 with any repeated words eliminated.

2436 The qualified data type name is required whenever the code list is used for a qualified data type with a
 2437 restricted set of valid code values.

2438 Example 4-25: Code list tokens

```

2439 Code list token for Name type. Code is clm63403
2440 where
2441 6 = the value for UN/ECE in UN/EDIFACT data element 3055 representing the
2442 Code List. Agency. Identifier
2443 3403 = UN/EDIFACT data element tag for Name status code representing the Code
2444 List. Identification. Identifier
2445
2446 Code list token for Person_Name type. Code is clmPersonNameType63403
2447 where
2448 PersonNameType = name of the qualified data type
2449 6 = the value for UN/ECE in UN/EDIFACT data element 3055 representing the
2450 Code List. Agency. Identifier
2451 3403 = UN/EDIFACT data element tag for Name status code representing the Code
2452 List. Identification. Identifier
2453
2454 Code list token for a proprietary code list for Document Security is
2455 clmzzzDocumentSecurity
2456 where
2457 zzz = the value for Mutually defined in UN/EDIFACT data element 3055
2458 representing the Code List. Agency. Identifier
2459 DocumentSecurity = the value for Code List. Name. Text
2460

```

2461 Example 4-26: Schema element of code lists

```

2462 <xsd:schema
2463 targetNamespace="urn:un:unece:unefact:codelist:draft:UNECE_Account_Type_Code:
2464 D.01C"
2465 xmlns:clm64437="urn:un:unece:unefact:codelist:draft:UNECE_Account_Type_Code:D.01C"
2466 xmlns:xsd="http://www.w3.org/2001/XMLSchema"
2467 elementFormDefault="qualified" attributeFormDefault="unqualified">

```

2468

2469 Note: External developers are encouraged to follow the above construct rule when customizing schemas
 2470 for codelists to ensure that there is no namespace conflict.

2471

2472 4.7.3 Imports and Includes

2473 UN/CEFACT Code List Schema Modules are intended to be standalone schema modules and will not
2474 import or include any other schema modules.

2475 [R 155] Code List schema modules MUST not import or include any other schema modules.

2476 4.7.4 Type Definitions

2477 [R 156] For the top-level element a corresponding top-level `xsd:complexType` MUST be defined.

2478 [R 157] The name for the `xsd:complexType` MUST be based on the value of the `code list. name.`
2479 `text.` with the word “type” appended.

2480 [R 158] Each `xsd:complexType` MUST contain one `xsd:simpleContent` element.

2481 [R 159] The `xsd:complexType` definition `xsd:simpleContent` element MUST contain one
2482 `xsd:extension` element. This `xsd:extension` element MUST include an `xsd` based attribute
2483 that defines the specific list containing the actual code values.

2484 A restriction has to be declared in order to define the content component (the simple type) as a restriction
2485 of the unqualified data type in order to comply with parser requirements. The restriction itself is the list of
2486 enumerations.

2487 [R 160] Within each code list module one, and only one, named `xsd:simpleType` MUST be defined for
2488 the content component.

2489 [R 161] The name of the `xsd:simpleType` MUST be the name of root element based on the value of
2490 the `code list. name. text.` with the word “ContentType” appended.

2491 Example 4-27: Simple type definition of code lists

```
2492 <!-- ===== Type Definitions ===== -->  
2493 <!-- =====  
2494 <!-- ===== Code List Type Definition: Account Type Code ===== -->  
2495 <!-- =====  
2496 <xsd:simpleType name="AccountTypeCodeContentType">  
2497 <xsd:restriction base="xsd:token">  
2498 <xsd:enumeration value="2">  
2499 ... see enumeration ...  
2500 </xsd:enumeration>  
2501 </xsd:restriction>  
2502 </xsd:simpleType>
```

2503

2504 [R 162] The `xsd:restriction` element base attribute value MUST be set to “`xsd:token`”.

2505 [R 163] Each code in the code list MUST be expressed as an `xsd:enumeration`, where the
2506 `xsd:value` for the enumeration is the actual code value.

2507 Example 4-28: Enumeration facet of code lists

```
2508 ... see type defintion ...  
2509 <xsd:enumeration value="2">  
2510 <xsd:annotation>  
2511 ... see annotation  
2512 </xsd:annotation>  
2513 </xsd:enumeration>  
2514 <xsd:enumeration value="15">  
2515 <xsd:annotation>
```

2516
2517
2518
2519

```
... see annotation
</xsd:annotation>
</xsd:enumeration>
...
```

2520

2521 [R 164] The XSD facet feature `MUST NOT` be used in the code list schema module.

2522 4.7.5 Element Declarations

2523 [R 165] For each code list a single root element `MUST` be globally declared.

2524 [R 166] The name of root element `MUST` be based on the `code list.name.text` following the
2525 naming rules as defined in section 2.3.

2526 [R 167] The root element `MUST` be of a type representing the actual list of code values.

2527 Example 4-29: Root element declaration of code lists

2528
2529
2530

```
<!-- ===== Root Element ===== -->
<!-- ===== -->
<xsd:element name="AccountTypeCode" type="clm64437:AccountTypeCodeContentType"/>
```

2531

2532 4.7.6 Extension and Restriction

2533 Users of the UN/CEFACT library may identify any subset they wish from a specific identifier list for their
2534 own trading community requirements by defining a qualified data type.

2535 Representation of qualified data type of code lists could be

- 2536 • a combination of several individual code lists using `xsd:union`
- 2537 • a choice between several code lists, using `xsd:choice`

2538 Both of these can easily be accommodated in this syntax solution, although they are not likely to be
2539 frequently used.

2540

2541 XML declarations for using codelists in qualified data types are shown in the following examples.

2542 Example 4-30: Usage of only one Code List

2543
2544
2545
2546
2547
2548
2549
2550
2551
2552
2553
2554
2555
2556
2557
2558
2559
2560
2561
2562
2563
2564
2565
2566
2567
2568

```
<xsd:simpleType name="TemperatureMeasureUnitCodeType">
<xsd:annotation>
... see annotation ...
</xsd:annotation>
<xsd:restriction base="clm66411:UnitCodeContentType">
<xsd:length value="3"/>
<xsd:enumeration value="BTU">
<xsd:annotation>
<xsd:documentation source="code" xml:lang="en">
<ccts:CodeName>British thermal unit</ccts:CodeName>
</xsd:documentation>
</xsd:annotation>
</xsd:enumeration>
<xsd:enumeration value="CEL">
<xsd:annotation>
<xsd:documentation source="code" xml:lang="en">
<ccts:CodeName>degree Celsius</ccts:CodeName>
</xsd:documentation>
</xsd:annotation>
</xsd:enumeration>
<xsd:enumeration value="FAH">
<xsd:annotation>
<xsd:documentation source="code" xml:lang="en">
<ccts:CodeName>degree Fahrenheit</ccts:CodeName>
</xsd:documentation>
</xsd:annotation>
</xsd:enumeration>
```

2569
2570
2571

```
</xsd:enumeration>  
</xsd:restriction>  
</xsd:simpleType>
```

2572 **Example 4-31 Usage of alternative Code Lists**

2573
2574
2575
2576
2577
2578
2579
2580
2581

```
<xsd:complexType name="PersonPropertyCodeType">  
<xsd:annotation>  
  ... see annotation ...  
</xsd:annotation>  
<xsd:choice>  
  <xsd:element ref="clm63479:MaritalCode"/>  
  <xsd:element ref="clm63499:GenderCode"/>  
</xsd:choice>  
</xsd:complexType>
```

2582 **Example 4-32: Combination of Code Lists**

2583
2584
2585
2586
2587
2588
2589

```
<xsd:simpleType name="AccountDutyCodeType">  
<xsd:annotation>  
  ... see annotation ...  
</xsd:annotation>  
<xsd:union memberTypes="clm64437:AccountTypeCodeContentType  
  clm65153:DutyTaxFeeTypeCodeContentType"/>  
</xsd:simpleType>
```

2590

2591 **4.7.7 Annotation**

2592 [R 168] Each `xsd:enumeration` MUST include an annotation documentation providing the code name
2593 and the code description.

2594 **Example 4-33: Annotation of codes**

2595
2596
2597
2598
2599
2600
2601
2602
2603
2604
2605

```
...  
<xsd:enumeration value="2">  
<xsd:annotation>  
  <xsd:documentation xml:lang="en">  
    <ccts:CodeName>Budgetary account</ccts:CodeName>  
    <ccts:CodeDescription>Code identifying a budgetary account.  
  </ccts:CodeDescription>  
  </xsd:documentation>  
</xsd:annotation>  
</xsd:enumeration>  
...
```

2606

2607 **4.8 Identifier Schemas**

2608 When required separate schema modules will be defined for identification schemes that have a token,
2609 and optionally a description, and that have the same functionality as a code list. Other identifier schemes
2610 should be defined as a qualified or unqualified data type as appropriate.

2611 [R 169] Where an identification scheme is identified (contains an identifier, a token for the identifier, and
2612 an optionally, a description) to have the same functionality as a code list and a business
2613 requirement exists for enumerating the identifier list, a schema module of the same form as the
2614 code list schema module MUST be created.

2615 External identifier lists must be used when they exist in schema module form and when they can be
2616 directly imported into a schema module.
2617 UN/CEFACT *may* design and use an internal identifier list where an existing external identifier list needs
2618 to be extended, or where no suitable external identifier list exists. If an identifier list is created, the lists
2619 should be globally scoped and designed for reuse and sharing.

2620 [R 170] Internal identifier lists MUST NOT not duplicate existing external identifier lists where they are
2621 available in a schema module form that is capable of being imported.

2622 [R 171] Each UN/CEFACT maintained identifier list MUST be defined in its own schema module.

2623 4.8.1 Schema Construct

2624 The identifier list schema module will follow the general pattern for all UN/CEFACT schema modules.
2625 Following the generic module information, the body of the schema will consist of identifier list definitions
2626 of the following general form:

2627 Example 4-34: Structure of identifier lists

```
2628 <?xml version="1.0" encoding="UTF-8"?>
2629 <!-- ===== -->
2630 <!-- ===== ISO Country Identifier - Identifier List Schema Module ===== -->
2631 <!-- ===== -->
2632 <!--
2633 Identifier ofCountry Identifier,
2634 Agency: ISO,
2635 Version: 2
2636 Last change: 25. June 2004
2637
2638 Copyright (C) UN/CEFACT (2004). All Rights Reserved.
2639
2640 ... see copyright information ...
2641
2642 -->
2643 <xsd:schema targetNamespace=" ... see namespace ...
2644 xmlns:xsd="http://www.w3.org/2001/XMLSchema"
2645 elementFormDefault="qualified" attributeFormDefault="unqualified">
2646 <!-- ===== Root Element ===== -->
2647 <!-- ===== -->
2648 ... see root element declaration ...
2649 <!-- ===== Type Definitions ===== -->
2650 <!-- ===== -->
2651 <!-- ===== Identifier List Type Definition: Country Identifier ===== -->
2652 <!-- ===== -->
2653 ... see type definition ...
2654 </xsd:schema>
```

2655

2656 4.8.2 Namespace Scheme

2657 [R 172] Each UN/CEFACT maintained identifier list schema module MUST be represented by a unique
2658 token constructed as follows:

2659 ids[Qualified data type name]<Identification Scheme Agency.
2660 Identifier><Identification Scheme. Identifier>

2661 Example 4-35: Identifier list token

```
2662 Token for the ISO Country Codes would be: ids53166-1
2663 where:
2664 5 = the Identification Scheme Agency Identifier for ISO in codelist 3055
2665 3166-1 = the Identification Scheme Identifier as allocated by ISO.
```

2666 Example 4-36: Namespace of Identifier list

```
2667 "urn:un:unece:uncefact:codelist:draft:ISO_Country_Identifier:2:4"
```

2668 Example 4-37: Schema element of identifier list

```
2669 <xsd:schema
2670 targetNamespace="urn:un:unece:uncefact:codelist:draft:ISO_Country_Identifier:2:4"
2671 xmlns:ids53166-1="urn:un:unece:uncefact:codelist:draft:ISO_Country_Identifier:2:4"
2672 xmlns:xsd="http://www.w3.org/2001/XMLSchema"
2673 elementFormDefault="qualified" attributeFormDefault="unqualified">
```

2674

2675 Note: External developers are encouraged to follow the above construct rule when customizing schemas
2676 for identifier lists to ensure that there is no namespace conflict.

2677 4.8.3 Imports and Includes

2678 UN/CEFACT Identifier List Schema Modules are intended to be standalone schema modules and will not
2679 import or include any other schema modules.

2680 [R 173] Identifier list schema modules MUST not import or include any other schema modules.

2681 4.8.4 Type Definitions

2682 A restriction has to be declared in order to define the content component (the simple type) as a restriction
2683 of the unqualified data type in order to comply with parser requirements. The restriction itself is the list of
2684 enumerations.

2685 [R 174] Within each identifier list module one, and only one, named **xsd:simpleType** MUST be defined
2686 for the content component.

2687 [R 175] The name of the **xsd:simpleType** MUST be the name of root element with the word
2688 "ContentType" appended.

2689 Example 4-38: Simple type definition of an identifier list

```
2690 <!-- ===== Type Definitions ===== -->  
2691 <!-- ===== -->  
2692 <xsd:simpleType name="CountryIdentifierContentType">  
2693 <xsd:restriction base="xsd:token">  
2694 <xsd:enumeration value="AU">  
2695 ... see enumeration ...  
2696 </xsd:enumeration>  
2697 </xsd:restriction>  
2698 </xsd:simpleType>
```

2699

2700 [R 176] The **xsd:restriction** element base attribute value MUST be set to "**xsd:token**".

2701 [R 177] Each identifier in the identifier list MUST be expressed as an **xsd:enumeration**, where the
2702 **xsd:value** for the enumeration is the actual identifier value.

2703 Example 4-39: Enumeration facet of an identifier list

```
2704 ... see type defintion ...  
2705 <xsd:enumeration value="AU">  
2706 <xsd:annotation>  
2707 ... see annotation  
2708 </xsd:annotation>  
2709 </xsd:enumeration>  
2710 <xsd:enumeration value="US">  
2711 <xsd:annotation>  
2712 ... see annotation  
2713 </xsd:annotation>  
2714 </xsd:enumeration>  
2715 ...
```

2716

2717 [R 178] The XSD facet feature MUST not be used in the identifier list schema module.

2718 4.8.5 Attribute and Element Declarations

2719 [R 179] For each identifier list a single root element MUST be globally declared.

2720 [R 180] The name of the root element MUST be based on the identification scheme. name.
2721 text following the naming rules as defined in section 2.3.

2722 [R 181] The root element MUST be of a type representing the actual list of identifier values.

2723 Example 4-40: Root element declaration of identifier lists

```
2724 <!-- ===== Root Element ===== -->  
2725 <!-- ===== -->  
2726 <xsd:element name="CountryIdentifier" type="ids53166:CountryIdentifierContentType"/>
```

2727

2728 4.8.6 Extension and Restriction

2729 Users of the UN/CEFACT library may identify any subset they wish from a specific identifier list for their
2730 own trading community requirements by defining a qualified data type.

2731

2732 Representation of qualified data type of identifier lists could be

- 2733 • a combination of several individual identifier lists using `xsd:union`
- 2734 • a choice between several identifier lists, using `xsd:choice`

2735 Both of these can easily be accommodated in this syntax solution, although they are not likely to be
2736 frequently used.

2737 Example 4-41: Enumeration facet of identifier scheme

```
2738 ... see type definition ...  
2739 <xsd:enumeration value="AD">  
2740 <xsd:annotation>  
2741 ... see annotation ...  
2742 </xsd:annotation>  
2743 </xsd:enumeration>  
2744 <xsd:enumeration value="AE">  
2745 <xsd:annotation>  
2746 ... see annotation ...  
2747 </xsd:annotation>  
2748 </xsd:enumeration>  
2749 <xsd:enumeration value="AF">  
2750 <xsd:annotation>  
2751 ... see annotation ...  
2752 </xsd:annotation>  
2753 </xsd:enumeration>  
2754 ... see type definition ...
```

2755 Example 4-42: Usage of only one identifier scheme

```
2756 <xsd:simpleType name="CountryIdentifierType">  
2757 <xsd:annotation>  
2758 ... see annotation ...  
2759 </xsd:annotation>  
2760 <xsd:restriction base="ids53166:CountryIdentifierContentType"/>  
2761 </xsd:simpleType>
```

2762 Example 4-43: Usage of alternative identifier schemes

```
2763 <xsd:complexType name="GeopoliticalIdentifierType">  
2764 <xsd:annotation>  
2765 ... see annotation ...  
2766 </xsd:annotation>  
2767 <xsd:choice>  
2768 <xsd:element ref="ids53166:CountryCode"/>  
2769 <xsd:element ref="ids53166-2:RegionCode"/>  
2770 </xsd:choice>  
2771 </xsd:complexType>
```

2772 4.8.7 Annotation

2773 [R 182] Each `xsd:enumeration` MUST include an annotation documentation providing the identifier
2774 name and optionally the description of the identifier.

2775 Example 4-44: Annotation of Identifiers

```
2776 ...  
2777 <xsd:enumeration value="AU">  
2778 <xsd:annotation>  
2779   <xsd:documentation xml:lang="en">  
2780     <ccd:IdentifierName>Australia</ccd:IdentifierName>  
2781   </xsd:documentation>  
2782 </xsd:annotation>  
2783 </xsd:enumeration>  
2784 ...
```

2785 **5 XML Instance Documents**

2786 All UN/CEFACT compliant XML instance documents will be based on UN/CEFACT compliant XML
2787 schemas for core components. The instance documents must be valid against the relevant UN/CEFACT
2788 compliant XML schemas. The XML instance documents should be readable and understandable for
2789 human and application, and should enable reasonably intuitive interactions. It should represent all
2790 truncated tag names as described in section 4. A Xpath navigation path should describe the complete
2791 semantic understanding by concatenating the nested elements. This navigation path should also reflect
2792 the meaning of each dictionary entry name of a BBIE or ASBIE.
2793

2794 **5.1 Character Encoding**

2795 In conformance with ISO/IETF/ITU/UNCEFACT Memorandum of Understanding Management Group
2796 (MOUMG) Resolution 01/08 (MOU/MG01n83) as agreed to by UN/CEFACT, all UN/CEFACT XML will be
2797 instantiated using UTF. UTF-8 is the preferred encoding, but UTF-16 may be used where necessary to
2798 support other languages.

2799 [R 183] All UN/CEFACT XML MUST be instantiated using UTF . UTF-8 should be used as the preferred
2800 encoding. If UTF-8 is not used, UTF-16 MUST be used.

2801 **5.2 Empty Content**

2802 Empty elements do not provide the level of assurance necessary for business information exchanges and
2803 as such, will not be used.

2804 [R 184] UN/CEFACT conformant instance documents MUST NOT contain an element devoid of content.

2805 **5.3 xsi:type**

2806 The **xsi:type** attribute allows for substitution during an instantiation of a xml document. In the same
2807 way that substitution groups are not allowed, the **xsi:type** attribute is not allowed.

2808 [R 185] The **xsi:type** attribute MUST NOT be used.

2809

Appendix A. Overall Structure

2810 The structure of an UN/CEFACT compliant XML schema must contain one or more of the following
2811 sections as relevant. Relevant sections must appear in the order given:

- 2812 1. XML Declaration
- 2813 2. Schema Module Identification and Copyright Information
- 2814 3. Schema Start-Tag
- 2815 4. Includes
- 2816 5. Imports
- 2817 6. Root element
- 2818 7. Type Definitions
- 2819

2820

1.XML Declaration

2821 A UTF-8 encoding is adopted throughout all UN/CEFACT XML schemas.

2822 **Example 1-1: XML Declaration**

2823

```
<?xml version="1.0" encoding="UTF-8"?>
```

2824

2.Schema Module Identification and Copyright Information

2825

Example 2-1: Copyright Information

2826

```
<!-- ===== -->
```

2827

```
<!-- ===== Examples Schema Module; 0.3 Rev.6 ===== -->
```

2828

```
<!-- ===== -->
```

2829

```
<!--
```

2830

```
Module:      Example
```

2831

```
Agency:     UN/CEFACT
```

2832

```
Version:    0.3 Rev. 6
```

2833

```
Last change: 25. June 2004
```

2834

2835

```
Copyright (C) UN/CEFACT (2004). All Rights Reserved.
```

2836

2837

```
This document and translations of it may be copied and furnished to others, and
```

2838

```
derivative works that comment on or otherwise explain it or assist in its
```

2839

```
implementation may be prepared, copied, published and distributed, in whole or in
```

2840

```
part, without restriction of any kind, provided that the above copyright notice and
```

2841

```
this paragraph are included on all such copies and derivative works. However, this
```

2842

```
document itself may not be modified in any way, such as by removing the copyright
```

2843

```
notice or references to UN/CEFACT, except as needed for the purpose of developing
```

2844

```
UN/CEFACT specifications, in which case the procedures for copyrights defined in the
```

2845

```
UN/CEFACT Intellectual Property Rights document must be followed, or as required to
```

2846

```
translate it into languages other than English.
```

2847

2848

```
The limited permissions granted above are perpetual and will not be revoked by
```

2849

```
UN/CEFACT or its successors or assigns.
```

2850

2851

```
This document and the information contained herein is provided on an "AS IS" basis and
```

2852

```
UN/CEFACT DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO
```

2853

```
ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR
```

2854

```
ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.
```

2855

```
-->
```

2856

3. Schema Start-Tag

2857 The Schema Start-Tag section of an UN/CEFACT compliant XML schema must contain one or more of
2858 the below declarations as relevant. Relevant declarations must appear in the order given:

- 2859 • Version
- 2860 • Namespaces
 - 2861 • targetNamespace attribute
 - 2862 • xmlns:xsd attribute
 - 2863 • namespace declaration for reusable ABIEs actually used in the schema
 - 2864 • namespace declaration for unqualified data types actually used in the schema
 - 2865 • namespace declaration for qualified data types actually used in the schema
 - 2866 • namespace declaration for code lists actually used in the schema
 - 2867 • namespace declaration for identifier schemes actually used in the schema
- 2868 • Form Defaults
 - 2869 • elementFormDefault
 - 2870 • attributeFormDefault
- 2871 • Others
 - 2872 • other schema attributes with schema namespace
 - 2873 • other schema attributes with non-schema namespace

2874 Example 3-1: XML Schema Start Tag

2875
2876
2877
2878
2879
2880
2881
2882
2883
2884
2885
2886
2887
2888
2889
2890
2891
2892
2893
2894
2895
2896
2897
2898
2899
2900
2901
2902
2903
2904
2905

```
<xsd:schema
targetNamespace="urn:un:unece:uncefact:data:draft:ExamplesSchemaModule:0:3:6"
xmlns:rsm="urn:un:unece:uncefact:data:draft:ExamplesSchemaModule:0:3:6"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"

xmlns:ram="urn:un:unece:uncefact:data:draft:ReusableAggregateBusinessInformationEntity
SchemaModule:0:3:6"

xmlns:udt="urn:un:unece:uncefact:data:draft:UnqualifiedDataTypeSchemaModule:0:3:6"
xmlns:qdt="urn:un:unece:uncefact:data:draft:QualifiedDataTypeSchemaModule:0:3:6"

xmlns:ids53166="urn:un:unece:uncefact:codelist:draft:ISO_Country_Code:3"
xmlns:ids53166-2="urn:un:unece:uncefact:codelist:draft:ISO_Region_Code:3"

xmlns:clm65153="urn:un:unece:uncefact:codelist:draft:UNECE_Duty_Or_Tax_Or_Fee_Type_Cod
e:D.01C" xmlns:clm64405="urn:un:unece:uncefact:codelist:draft:UNECE_Status_Code:D.01C"
xmlns:clm69143="urn:un:unece:uncefact:codelist:draft:UNECE_Relationship_Code:D.01C"
xmlns:clmPerson_Characteristic_Code63289="urn:un:unece:uncefact:codelist:draft:UNECE_P
erson_Characteristic_Code:D.01C"
xmlns:clm63479="urn:un:unece:uncefact:codelist:draft:UNECE_Marital_Code:D.01C"
xmlns:clm63499="urn:un:unece:uncefact:codelist:draft:UNECE_Gender_Code:D.01C"
xmlns:clm1161131="urn:un:unece:uncefact:codelist:draft:
ANSI_ASC_X12_Calculation_Code:4031"
xmlns:
clm66411="urn:un:unece:uncefact:codelist:draft:UNECE_Codes_for_Units_of_Measure:2001"
xmlns:clm54217="urn:un:unece:uncefact:codelist:draft:ISO_Currency_Codes:2001"
xmlns:clm5639="urn:un:unece:uncefact:codelist:draft:ISO_Language_Code:1988:3"
xmlns:clm64437="urn:un:unece:uncefact:codelist:draft:UNECE_Account_Type_Code:D.01C"

elementFormDefault="qualified"
attributeFormDefault="unqualified">
```

2906

4. Includes

2907 The Include section of an UN/CEFACT compliant XML schema must contain one or more of the below
2908 declarations as relevant. Relevant declarations must appear in the order given:

- 2909 • Inclusion of the internal ABIE schema module if used

2910 **Example 4-1: Includes**

2911

2912

2913

2914

2915

2916

2917

2918

2919

2920

2921

2922

```
<!-- ===== -->
<!-- ===== Include ===== -->
<!-- ===== -->
<!-- ===== Inclusion of internal ABIE ===== -->
<!-- ===== -->
<xsd:include
namespace="urn:un:unece:uncefact:data:draft:InternalAggregateBusinessInformationEntity
SchemaModule:0:3:6"
schemaLocation="http://www.unece.org/uncefact/data/InternalAggregateBusinessInformatio
nEntitySchemaModule_0.3.6_draft.xsd"/>
```

2923

5. Imports

2924 The Import section of an UN/CEFACT compliant XML schema must contain one or more of the below
2925 declarations as relevant. Relevant declarations must appear in the order given:

- 2926 • Import of the reusable ABIE schema module if used
- 2927 • Import of the unqualified data type schema module if used
- 2928 • Import of the qualified data type schema module if used
- 2929 • Import of codelist schema modules actually used
- 2930 • Import of identifier scheme schema modules actually used

2931 Example 5-1: Imports

```
2932 <!-- ===== Imports ===== -->
2933 <!-- ===== Imports ===== -->
2934 <!-- ===== Imports ===== -->
2935 <!-- ===== Import of reusable Aggregate Business Information Entity == -->
2936 <!-- ===== Imports ===== -->
2937 <xsd:import
2938 namespace="urn:un:unece:uncefact:data:draft:ReusableAggregateBusinessInformationSchema
2939 Module:0:3:6" schemaLocation=" http://www.unece.org/uncefact/data/
2940 ReusableAggregateBusinessInformationEntitySchemaModule_0.3.6_draft.xsd"/>
2941 <!-- ===== Imports ===== -->
2942 <!-- ===== Import of Unqualified Data Type ===== -->
2943 <!-- ===== Imports ===== -->
2944 <xsd:import
2945 namespace="urn:un:unece:uncefact:data:draft:UnqualifiedDataTypeSchemaModule:0:3:6"
2946 schemaLocation=" http://www.unece.org/uncefact/data/
2947 UnqualifiedDataTypeSchemaModule_0.3.6_draft.xsd"/>
2948 <!-- ===== Imports ===== -->
2949 <!-- ===== Import of Qualified Data Type ===== -->
2950 <!-- ===== Imports ===== -->
2951 <xsd:import
2952 namespace="urn:un:unece:uncefact:data:draft:QualifiedDataTypeSchemaModule:0:3:6"
2953 schemaLocation=" http://www.unece.org/uncefact/data/
2954 QualifiedDataTypeSchemaModule_0.3.6_draft.xsd"/>
2955 <!-- ===== Imports ===== -->
2956 <!-- ===== Import of Code lists ===== -->
2957 <!-- ===== Imports ===== -->
2958 <xsd:import namespace="urn:un:unece:uncefact:codelist:draft:Account_Type_Code:D.01C"
2959 schemaLocation="http://www.unece.org/uncefact/
2960 codelist/UNECE_AccountTypeCode_0.3.6_draft.xsd"/>
2961 <xsd:import
2962 namespace="urn:un:unece:uncefact:codelist:draft:UNECE_Codes_for_Units_of_Measure:2001"
2963 schemaLocation="
2964 http://www.unece.org/uncefact/codelist/UNECE_Codes_for_Units_of_Measure_2001_draft.xsd
2965 "/>
2966 <xsd:import namespace="urn:un:unece:uncefact:codelist:draft:ISO_Currency_Codes:2001"
2967 schemaLocation="
2968 http://www.unece.org/uncefact/codelist/ISO_CurrencyCode_2001_draft.xsd"/>
2969 <xsd:import namespace="urn:un:unece:uncefact:codelist:draft:ISO_Language_Code:1988"
2970 schemaLocation="http://www.unece.org/uncefact/codelist/ISO_LanguageCode_1988_draft.xsd
2971 "/>
2972 <xsd:import
2973 namespace="urn:un:unece:uncefact:codelist:draft:ANSI_ASC_X12_Calculation_Code:4031"
2974 schemaLocation="http://www.unece.org/uncefact/codelist/ANSI_ASC_X12_CalculationCode_40
2975 31_draft.xsd"/>
2976 <xsd:import namespace="urn:un:unece:uncefact:codelist:draft:UNECE_Gender_Code:D.01C"
2977 schemaLocation="http://www.unece.org/uncefact/codelist/UNECE_GenderCode_D.01C_draft.xs
2978 d"/>
2979 <xsd:import namespace="urn:un:unece:uncefact:codelist:draft:UNECE_Marital_Code:D.01C"
2980 schemaLocation="http://www.unece.org/uncefact/codelist/UNECE_MaritalCode_D.01C_draft.x
2981 sd"/>
2982 <xsd:import
2983 namespace="urn:un:unece:uncefact:codelist:draft:UNECE_Person_Characteristic_Code:D.01C
2984 "
2985 schemaLocation="http://www.unece.org/uncefact/codelist/UNECE_PersonCharacteristicCode_
2986 D.01C_draft.xsd"/>
2987 <xsd:import
2988 namespace="urn:un:unece:uncefact:codelist:draft:UNECE_Relationship_Code:D.01C"
2989 schemaLocation="http://www.unece.org/uncefact/codelist/UNECE_RelationshipCode_D.01C_dr
2990 aft.xsd"/>
```



```

2991 <xsd:import namespace="urn:un:unece:uncefact:codelist:draft:UNECE_Status_Code:D.01C"
2992 schemaLocation="http://www.unece.org/uncefact/codelist/UNECE_StatusCode_D.01C_draft.xsd"
2993 />
2994 <xsd:import
2995 namespace="urn:un:unece:uncefact:codelist:draft:UNECE_Duty_Or_Tax_Or_Fee_Type_Code:D.0
2996 1C" schemaLocation="http://www.unece.org/uncefact/UNECE_
2997 CodeList_DutyTaxFeeTypeCode_D.01C_draft.xsd"/>
2998 <!-- ===== -->
2999 <!-- ===== Import of Identifier Schemes ===== -->
3000 <!-- ===== -->
3001 <xsd:import namespace="urn:un:unece:uncefact:codelist:draft:ISO_Country_Code:3"
3002 schemaLocation="http://www.unece.org/uncefact/codelist/ISO_CountryCode_3_draft.xsd"/>
3003 <xsd:import namespace="urn:un:unece:uncefact:codelist:draft:ISO_Region_Code:3"
3004 schemaLocation="http://www.unece.org/uncefact/codelist/ISO_RegionCode_3_draft.xsd"/>
3005

```

3006 6. Root element

3007 The root element's type definition is defined immediately following the definition of the global root element
3008 to provide clear visibility of the root element's type, of which this particular schema is all about.

3009 Example 6-1:

```

3010 <!-- ===== -->
3011 <!-- ===== Root element ===== -->
3012 <!-- ===== -->
3013 <xsd:element name="PurchaseOrder" type="exp:PurchaseOrderType">
3014 <xsd:annotation>
3015 <xsd:documentation>
3016 <ccts:UniqueID>UNM0000001</ccts:UniqueID>
3017 <ccts:CategoryCode>RSM</ccts:CategoryCode>
3018 <ccts:Name>PurchaseOrder</ccts:Name>
3019 <ccts:VersionID>1.0</ccts:VersionID>
3020 <ccts:Description>A document that contains information directly relating to
3021 the economic event of ordering products.</ccts:Description>
3022 <ccts:BusinessDomain>TBG1</ccts:BusinessDomain>
3023 <ccts:BusinessProcessContext>Purchase Order</ccts:BusinessProcessContext>
3024 </xsd:documentation>
3025 </xsd:annotation>
3026 </xsd:element>

```

3027

7. Type Definitions

3028

- Definition of types for Basic Business Information Entities in alphabetical order, if applicable.

3029

- Definition of types for Aggregate Business Information Entities in alphabetical order, if applicable.

3030

Example 7-1:

3031

3032

3033

3034

3035

3036

3037

3038

3039

3040

3041

3042

3043

3044

3045

3046

3047

3048

3049

3050

3051

3052

3053

3054

3055

3056

3057

3058

3059

3060

3061

3062

3063

3064

3065

3066

3067

3068

3069

3070

3071

3072

3073

3074

3075

3076

3077

3078

3079

3080

3081

3082

3083

3084

3085

3086

3087

3088

3089

3090

3091

3092

3093

3094

3095

```
<!-- ===== -->
<!-- ===== Type Definitions ===== -->
<!-- ===== -->
<!-- ===== Type Definitions: Account type ===== -->
<!-- ===== -->
<xsd:complexType name="AccountType">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      <ccts:UniqueID>UN00000001</ccts:UniqueID>
      <ccts:CategoryCode>ABIE</ccts:CategoryCode>
      <ccts:DictionaryEntryName>Account. Details</ccts:DictionaryEntryName>
      <ccts:VersionID>1.0</ccts:VersionID>
      <ccts:DefinitionText>A business arrangement whereby debits and/or credits arising
from transactions are recorded. This could be with a bank, i.e. a financial account,
or a trading partner offering supplies or services 'on account', i.e. a commercial
account</ccts:DefinitionText>
      <ccts:ObjectClassTermName>Account</ccts:ObjectClassTermName>
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="ID" type="udt:IdentifierType" minOccurs="0"
maxOccurs="unbounded">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          <ccts:UniqueID>UN00000002</ccts:UniqueID>
          <ccts:CategoryCode>BBIE</ccts:CategoryCode>
          <ccts:DictionaryEntryName>Account. Identifier</ccts:DictionaryEntryName>
          <ccts:VersionID>1.0</ccts:VersionID>
          <ccts:DefinitionText>The identification of a specific
account.</ccts:DefinitionText>
          <ccts:CardinalityText>0..n</ccts:CardinalityText>
          <ccts:ObjectClassTermName>Account</ccts:ObjectClassTermName>
          <ccts:PropertyTermName>Identifier</ccts:PropertyTermName>
          <ccts:RepresentationTermName>Identifier</ccts:RepresentationTermName>
          <ccts:BusinessTermName>Account Number</ccts:BusinessTermName>
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
    <xsd:element name="Status" type="ram:StatusType" minOccurs="0"
maxOccurs="unbounded">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          <ccts:UniqueID>UN00000003</ccts:UniqueID>
          <ccts:CategoryCode>ASBIE</ccts:CategoryCode>
          <ccts:DictionaryEntryName>Account. Status</ccts:DictionaryEntryName>
          <ccts:VersionID>1.0</ccts:VersionID>
          <ccts:DefinitionText>Status information related to account
details.</ccts:DefinitionText>
          <ccts:CardinalityText>0..n</ccts:CardinalityText>
          <ccts:ObjectClassTermName>Account</ccts:ObjectClassTermName>
          <ccts:PropertyTermName>Status</ccts:PropertyTermName>
          <ccts:RepresentationTermName>Code</ccts:RepresentationTermName>
          <ccts:AssociatedObjectClassTermName>Status</ccts:AssociatedObjectClassTermName>
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
    <xsd:element name="Name" type="udt:NameType" minOccurs="0" maxOccurs="unbounded">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          <ccts:UniqueID>UN00000004</ccts:UniqueID>
          <ccts:CategoryCode>BBIE</ccts:CategoryCode>
          <ccts:DictionaryEntryName>Account. Name. Text</ccts:DictionaryEntryName>
          <ccts:VersionID>1.0</ccts:VersionID>
          <ccts:DefinitionText>The text name for a specific account</ccts:DefinitionText>
          <ccts:CardinalityText>0..n</ccts:CardinalityText>

```

3096
3097
3098
3099
3100
3101
3102
3103
3104
3105
3106
3107
3108
3109
3110
3111
3112
3113
3114
3115
3116
3117
3118
3119
3120
3121
3122
3123
3124
3125
3126
3127
3128
3129
3130
3131
3132
3133
3134
3135
3136
3137
3138
3139
3140
3141
3142
3143
3144
3145
3146
3147
3148
3149
3150
3151
3152
3153
3154
3155
3156
3157
3158
3159
3160
3161
3162
3163
3164
3165
3166
3167
3168
3169
3170
3171

```
<ccts:ObjectClassTermName>Account</ccts:ObjectClassTermName>
<ccts:PropertyTermName>Name</ccts:PropertyTermName>
<ccts:RepresentationTermName>Text</ccts:RepresentationTermName>
</xsd:documentation>
</xsd:annotation>
</xsd:element>
<xsd:element name="CurrencyCode" type="qdt:CurrencyCodeType" minOccurs="0"
maxOccurs="unbounded">
<xsd:annotation>
<xsd:documentation xml:lang="en">
<ccts:UniqueID>UN00000005</ccts:UniqueID>
<ccts:CategoryCode>BBIE</ccts:CategoryCode>
<ccts:DictionaryEntryName>Account. Currency. Code</ccts:DictionaryEntryName>
<ccts:VersionID>1.0</ccts:VersionID>
<ccts:DefinitionText>A code specifying the currency in which monies are held
within the account.</ccts:DefinitionText>
<ccts:CardinalityText>0..n</ccts:CardinalityText>
<ccts:ObjectClassTermName>Account</ccts:ObjectClassTermName>
<ccts:PropertyTermName>Currency</ccts:PropertyTermName>
<ccts:RepresentationTermName>Code</ccts:RepresentationTermName>
</xsd:documentation>
</xsd:annotation>
</xsd:element>
<xsd:element name="TypeCode" type="qdt:AccountTypeCodeType" minOccurs="0"
maxOccurs="unbounded">
<xsd:annotation>
<xsd:documentation xml:lang="en">
<ccts:UniqueID>UN00000006</ccts:UniqueID>
<ccts:CategoryCode>BBIE</ccts:CategoryCode>
<ccts:DictionaryEntryName>Account. Type. Code</ccts:DictionaryEntryName>
<ccts:VersionID>1.0</ccts:VersionID>
<ccts:DefinitionText>This provides the ability to indicate what type of account
this is (checking, savings, etc).</ccts:DefinitionText>
<ccts:CardinalityText>0..1</ccts:CardinalityText>
<ccts:ObjectClassTermName>Account</ccts:ObjectClassTermName>
<ccts:PropertyTermName>Type</ccts:PropertyTermName>
<ccts:RepresentationTermName>Code</ccts:RepresentationTermName>
</xsd:documentation>
</xsd:annotation>
</xsd:element>
<xsd:element name="Country" type="ram:CountryType" minOccurs="0"
maxOccurs="unbounded">
<xsd:annotation>
<xsd:documentation xml:lang="en">
<ccts:UniqueID>UN00000007</ccts:UniqueID>
<ccts:CategoryCode>ASBIE</ccts:CategoryCode>
<ccts:DictionaryEntryName>Account. Country</ccts:DictionaryEntryName>
<ccts:VersionID>1.0</ccts:VersionID>
<ccts:DefinitionText>Country information related to account
details.</ccts:DefinitionText>
<ccts:CardinalityText>0..n</ccts:CardinalityText>
<ccts:ObjectClassTermName>Account</ccts:ObjectClassTermName>
<ccts:PropertyTermName>Country</ccts:PropertyTermName>
<ccts:AssociatedObjectClassTermName>Country</ccts:AssociatedObjectClassTermName>
</xsd:documentation>
</xsd:annotation>
</xsd:element>
<xsd:element name="Person" type="ram:PersonType" minOccurs="0"
maxOccurs="unbounded">
<xsd:annotation>
<xsd:documentation xml:lang="en">
<ccts:UniqueID>UN00000008</ccts:UniqueID>
<ccts:CategoryCode>ASBIE</ccts:CategoryCode>
<ccts:DictionaryEntryName>Account. Person</ccts:DictionaryEntryName>
<ccts:VersionID>1.0</ccts:VersionID>
<ccts:DefinitionText>Associated person information related to account details.
This can be used to identify multiple people related to an account, for instance, the
account holder.</ccts:DefinitionText>
<ccts:CardinalityText>0..n</ccts:CardinalityText>
<ccts:ObjectClassTermName>Account</ccts:ObjectClassTermName>
<ccts:PropertyTermName>Person</ccts:PropertyTermName>
<ccts:AssociatedObjectClassTermName>Person</ccts:AssociatedObjectClassTermName>
</xsd:documentation>
</xsd:annotation>
```

```

3172     </xsd:element>
3173     <xsd:element name="Organisation" type="ram:OrganisationType" minOccurs="0"
3174     maxOccurs="unbounded">
3175         <xsd:annotation>
3176             <xsd:documentation xml:lang="en">
3177                 <ccts:UniqueID>UN00000009</ccts:UniqueID>
3178                 <ccts:CategoryCode>ASBIE</ccts:CategoryCode>
3179                 <ccts:DictionaryEntryName>Account. Organisation</ccts:DictionaryEntryName>
3180                 <ccts:VersionID>1.0</ccts:VersionID>
3181                 <ccts:DefinitionText>The associated organisation information related to account
3182                 details. This can be used to identify multiple organisations related to this account,
3183                 for instance, the account holder.</ccts:DefinitionText>
3184                 <ccts:CardinalityText>0..n</ccts:CardinalityText>
3185                 <ccts:ObjectClassTermName>Account</ccts:ObjectClassTermName>
3186                 <ccts:PropertyTermName>Organisation</ccts:PropertyTermName>
3187
3188                 <ccts:AssociatedObjectClassTermName>Organisation</ccts:AssociatedObjectClassTermName>
3189             </xsd:documentation>
3190         </xsd:annotation>
3191     </xsd:element>
3192 </xsd:sequence>
3193 </xsd:complexType>

```

3194 Example 7-2: Complete Structure

```

3195 <?xml version="1.0" encoding="UTF-8"?>
3196 <!-- =====>
3197 <!-- ===== [MODULENAME] Schema Module; [VERSION] =====>
3198 <!-- =====>
3199 <!--
3200 Module:      [MODULENAME]
3201 Agency:     UN/CEFACT
3202 Version:    [VERSION]
3203 Last change: [DATE OF LAST CHANGE]
3204
3205 Copyright (C) UN/CEFACT (2004). All Rights Reserved.
3206
3207 This document and translations of it may be copied and furnished to others, and
3208 derivative works that comment on or otherwise explain it or assist in its
3209 implementation may be prepared, copied, published and distributed, in whole or in
3210 part, without restriction of any kind, provided that the above copyright notice and
3211 this paragraph are included on all such copies and derivative works. However, this
3212 document itself may not be modified in any way, such as by removing the copyright
3213 notice or references to UN/CEFACT, except as needed for the purpose of developing
3214 UN/CEFACT specifications, in which case the procedures for copyrights defined in the
3215 UN/CEFACT Intellectual Property Rights document must be followed, or as required
3216 to translate it into languages other than English.
3217
3218 The limited permissions granted above are perpetual and will not be revoked by
3219 UN/CEFACT or its successors or assigns.
3220
3221 This document and the information contained herein is provided on an "AS IS" basis and
3222 UN/CEFACT DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO
3223 ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR
3224 ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.
3225 -->
3226 <xsd:schema
3227     targetNamespace="urn:un:unece:unefact:data:draft:[MODULENAME]:[VERSION]"
3228     xmlns:xsd="http://www.w3.org/2001/XMLSchema"
3229     ... FURTHER NAMESPACES ...
3230     elementFormDefault="qualified" attributeFormDefault="unqualified">
3231 <!-- =====>
3232 <!-- ===== Include =====>
3233 <!-- =====>
3234 <!-- ===== Inclusion of [TYPE OF MODULE] =====>
3235 <!-- =====>
3236 <xsd:include namespace="..." schemaLocation="..." />
3237 <!-- =====>
3238 <!-- ===== Imports =====>
3239 <!-- =====>
3240 <!-- ===== Import of [TYPE OF MODULE] =====>
3241 <!-- =====>
3242 <xsd:import namespace="..." schemaLocation="..." />
3243 <!-- =====>
3244 <!-- ===== Root element =====>

```

3245
3246
3247
3248
3249
3250
3251
3252
3253
3254
3255
3256
3257
3258

```
<!-- ===== -->  
<xsd:element name="[ELEMENTNAME]" type="[TOKEN]:[TYPENAME]">  
<!-- ===== -->  
<!-- ===== Type Definitions ===== -->  
<!-- ===== Type Definitions: [TYPE] ===== -->  
<!-- ===== -->  
<xsd:complexType name="[TYPENAME]">  
<xsd:restriction base="xsd:token">  
    ... see type definition ...  
</xsd:restriction>  
</xsd:complexType>  
</xsd:schema>
```

Appendix B. Annotation Templates

3259

The following templates define the annotation for each of the schema modules.

3260

3261

```
<!-- Root Schema Documentation -->
```

3262

```
  <xsd:annotation>
```

3263

```
    <xsd:documentation xml:lang="en">
```

3264

```
      <ccts:UniqueID></ccts:UniqueID>
```

3265

```
      <ccts:CategoryCode>RSM</ccts:CategoryCode>
```

3266

```
      <ccts:Name></ccts:Name>
```

3267

```
      <ccts:VersionID></ccts:VersionID>
```

3268

```
      <ccts:Description></ccts:Description>
```

3269

```
      <ccts:BusinessDomain></ccts:BusinessDomain>
```

3270

```
      <ccts:BusinessProcessContext></ccts:BusinessProcessContext>
```

3271

```
      <ccts:GeopoliticalOrRegionContext></ccts:GeopoliticalOrRegionContext>
```

3272

```
      <ccts:OfficialConstraintContext></ccts:OfficialConstraintContext >
```

3273

```
      <ccts:ProductContext></ccts:ProductContext>
```

3274

```
      <ccts:IndustryContext></ccts:IndustryContext >
```

3275

```
      <ccts:BusinessProcessRoleContext></ccts:BusinessProcessRoleContext >
```

3276

```
      <ccts:SupportingRoleContext></ccts:SupportingRoleContext>
```

3277

```
      <ccts:SystemCapabilitiesContext></ccts:SystemCapabilitiesContext>
```

3278

```
    </xsd:documentation>
```

3279

```
  </xsd:annotation>
```

3280

3281

3282

```
<!-- ABIE's Documentation -->
```

3283

```
  <xsd:annotation>
```

3284

```
    <xsd:documentation xml:lang="en">
```

3285

```
      <ccts:UniqueID></ccts:UniqueID>
```

3286

```
      <ccts:CategoryCode>ABIE</ccts:CategoryCode>
```

3287

```
      <ccts:DictionaryEntryName></ccts:DictionaryEntryName>
```

3288

```
      <ccts:VersionID></ccts:VersionID>
```

3289

```
      <ccts:DefinitionText></ccts:DefinitionText>
```

3290

```
      <ccts:ObjectClassTermName></ccts:ObjectClassTermName>
```

3291

```
      <ccts:QualifierTermName></ccts:QualifierTermName>
```

3292

```
      <ccts:UsageRuleText ></ccts:UsageRuleText>
```

3293

```
      <ccts:BusinessTermName ></ccts:BusinessTermName>
```

3294

```
      <ccts:BusinessProcessContext></ccts:BusinessProcessContext>
```

3295

```
      <ccts:GeopoliticalOrRegionContext></ccts:GeopoliticalOrRegionContext>
```

3296

```
      <ccts:OfficialConstraintContext></ccts:OfficialConstraintContext >
```

3297

```
      <ccts:ProductContext></ccts:ProductContext>
```

3298

```
      <ccts:IndustryContext></ccts:IndustryContext >
```

3299

```
      <ccts:BusinessProcessRoleContext></ccts:BusinessProcessRoleContext >
```

3300

```
      <ccts:SupportingRoleContext></ccts:SupportingRoleContext>
```

3301

```
      <ccts:SystemCapabilitiesContext></ccts:SystemCapabilitiesContext>
```

3302

```
      <ccts:Example></ccts:Example>
```

3303

```
    </xsd:documentation>
```

3304

```
  </xsd:annotation>
```

3305

3306

```
<!-- BBIE's Documentation -->
```

3307

```
  <xsd:annotation>
```

3308

```
    <xsd:documentation xml:lang="en">
```

3309

```
      <ccts:UniqueID></ccts:UniqueID>
```

3310

```
      <ccts:CategoryCode>BBIE</ccts:CategoryCode>
```

3311

```
      <ccts:DictionaryEntryName></ccts:DictionaryEntryName>
```

3312

```
      <ccts:VersionID></ccts:VersionID>
```

3313

```
      <ccts:DefinitionText></ccts:DefinitionText>
```

3314

```
      <ccts:CardinalityText></ccts:CardinalityText>
```

3315

```
      <ccts:ObjectClassTermName></ccts:ObjectClassTermName>
```

3316

```

3317     <ccts:PropertyTermName></ccts:PropertyTermName>
3318     <ccts:QualifierTermName></ccts:QualifierTermName>
3319     <ccts:UsageRuleText ></ccts:UsageRuleText>
3320     <ccts:BusinessTermName ></ccts:BusinessTermName>
3321     <ccts:BusinessProcessContext></ccts:BusinessProcessContext>
3322     <ccts:GeopoliticalOrRegionContext></ccts:GeopoliticalOrRegionContext>
3323     <ccts:OfficialConstraintContext></ccts:OfficialConstraintContext >
3324     <ccts:ProductContext></ccts:ProductContext>
3325     <ccts:IndustryContext></ccts:IndustryContext >
3326     <ccts:BusinessProcessRoleContext></ccts:BusinessProcessRoleContext >
3327     <ccts:SupportingRoleContext></ccts:SupportingRoleContext>
3328     <ccts:SystemCapabilitiesContext></ccts:SystemCapabilitiesContext>
3329     <ccts:Example></ccts:Example>
3330   </xsd:documentation>
3331 </xsd:annotation>
3332
3333 <!-- ASBIE's Documentation -->
3334   <xsd:annotation>
3335     <xsd:documentation xml:lang="en">
3336       <ccts:UniqueID></ccts:UniqueID>
3337       <ccts:CategoryCode>ASBIE</ccts:CategoryCode>
3338       <ccts:DictionaryEntryName></ccts:DictionaryEntryName>
3339       <ccts:VersionID></ccts:VersionID>
3340       <ccts:DefinitionText></ccts:DefinitionText>
3341       <ccts:CardinalityText></ccts:CardinalityText>
3342       <ccts:ObjectClassTermName></ccts:ObjectClassTermName>
3343       <ccts:PropertyTermName></ccts:PropertyTermName>
3344       <ccts:AssociatedObjectClassTermName></ccts:AssociatedObjectClassTermName>
3345       <ccts:QualifierTermName></ccts:QualifierTermName>
3346       <ccts:UsageRuleText ></ccts:UsageRuleText>
3347       <ccts:BusinessTermName ></ccts:BusinessTermName>
3348       <ccts:BusinessProcessContext></ccts:BusinessProcessContext>
3349       <ccts:GeopoliticalOrRegionContext></ccts:GeopoliticalOrRegionContext>
3350       <ccts:OfficialConstraintContext></ccts:OfficialConstraintContext >
3351       <ccts:ProductContext></ccts:ProductContext>
3352       <ccts:IndustryContext></ccts:IndustryContext >
3353       <ccts:BusinessProcessRoleContext></ccts:BusinessProcessRoleContext >
3354       <ccts:SupportingRoleContext></ccts:SupportingRoleContext>
3355       <ccts:SystemCapabilitiesContext></ccts:SystemCapabilitiesContext>
3356       <ccts:Example></ccts:Example>
3357     </xsd:documentation>
3358   </xsd:annotation>
3359
3360 <!-- Qualified Data Types Documentation -->
3361   <xsd:annotation>
3362     <xsd:documentation xml:lang="en">
3363       <ccts:UniqueID/>
3364       <ccts:CategoryCode>QDT</ccts:CategoryCode>
3365       <ccts:DictionaryEntryName></ccts:DictionaryEntryName>
3366       <ccts:VersionID></ccts:VersionID>
3367       <ccts:DefinitionText></ccts:DefinitionText>
3368       <ccts:RepresentationTermName></ccts:RepresentationTermName>
3369       <ccts:QualifierTermName></ccts:QualifierTermName>
3370       <ccts:UsageRuleText></ccts:UsageRuleText>
3371       <ccts:BusinessTermName></ccts:BusinessTermName>
3372       <ccts:BusinessProcessContext></ccts:BusinessProcessContext>
3373       <ccts:GeopoliticalOrRegionContext></ccts:GeopoliticalOrRegionContext>
3374       <ccts:OfficialConstraintContext></ccts:OfficialConstraintContext >
3375       <ccts:ProductContext></ccts:ProductContext>
3376       <ccts:IndustryContext></ccts:IndustryContext >

```

```

3377         <ccts:BusinessProcessRoleContext></ccts:BusinessProcessRoleContext >
3378         <ccts:SupportingRoleContext></ccts:SupportingRoleContext>
3379         <ccts:SystemCapabilitiesContext></ccts:SystemCapabilitiesContext>
3380         <ccts:Example></ccts:Example>
3381     </xsd:documentation>
3382 </xsd:annotation>
3383
3384 <!-- Unqualified Data Type Documentation-->
3385     <xsd:annotation>
3386         <xsd:documentation xml:lang="en">
3387             <ccts:UniqueID></ccts:UniqueID>
3388             <ccts:CategoryCode>RT</ccts:CategoryCode>
3389             <ccts:DictionaryEntryName></ccts:DictionaryEntryName>
3390             <ccts:VersionID></ccts:VersionID>
3391             <ccts:DefinitionText></ccts:DefinitionText>
3392             <ccts:RepresentationTermName></ccts:RepresentationTermName>
3393             <ccts:UsageRuleText></ccts:UsageRuleText>
3394             <ccts:BusinessTermName></ccts:BusinessTermName>
3395             <ccts:Example></ccts:Example>
3396         </xsd:documentation>
3397     </xsd:annotation>
3398
3399 <!-- Unqualified Data Type Supplementary Component Documentation-->
3400     <xsd:annotation>
3401         <xsd:documentation xml:lang="en">
3402             <ccts:UniqueID></ccts:UniqueID>
3403             <ccts:CategoryCode>SC</ccts:CategoryCode>
3404             <ccts:DictionaryEntryName></ccts:DictionaryEntryName>
3405             <ccts:VersionID></ccts:VersionID>
3406             <ccts:DefinitionText></ccts:DefinitionText>
3407             <ccts:CardinalityText></ccts:CardinalityText>
3408             <ccts:PropertyTermName></ccts:PropertyTermName>
3409             <ccts:RepresentationTermName></ccts:RepresentationTermName>
3410             <ccts:UsageRuleText></ccts:UsageRuleText>
3411             <ccts:Example></ccts:Example>
3412         </xsd:documentation>
3413     </xsd:annotation>
3414
3415 <!-- Core Component Type Documentation -->
3416     <xsd:annotation>
3417         <xsd:documentation xml:lang="en">
3418             <ccts:UniqueID></ccts:UniqueID>
3419             <ccts:CategoryCode>CCT</ccts:CategoryCode>
3420             <ccts:DictionaryEntryName></ccts:DictionaryEntryName>
3421             <ccts:VersionID></ccts:VersionID>
3422             <ccts:DefinitionText></ccts:DefinitionText>
3423             <ccts:RepresentationTermName></ccts:RepresentationTermName>
3424             <ccts:UsageRuleText></ccts:UsageRuleText>
3425             <ccts:BusinessTermName></ccts:BusinessTermName>
3426             <ccts:Example></ccts:Example>
3427         </xsd:documentation>
3428     </xsd:annotation>
3429
3430 <!-- Code List / Identification Schema Documentation-->
3431     <xsd:annotation>
3432         <xsd:documentation xml:lang="en">
3433             <ccts:CodeName></ccts:CodeName>
3434             <ccts:CodeDescription></ccts:CodeDescription>
3435         </xsd:documentation>
3436     </xsd:annotation>

```


3437

Appendix C. Naming & Design Rules List

3438
3439

[R 1] CONFORMANCE SHALL BE DETERMINED THROUGH ADHERENCE TO THE CONTENT OF NORMATIVE SECTIONS, RULES AND DEFINITIONS.

3440
3441

[R 2] UN/CEFACT SCHEMA MUST FOLLOW THE STANDARD STRUCTURE DEFINED IN APPENDIX A.

3442
3443

[R 3] EACH ELEMENT OR ATTRIBUTE XML NAME MUST HAVE *ONE* AND ONLY *ONE* FULLY QUALIFIED XPATH (FQXP).

3444
3445

[R 4] ELEMENT, ATTRIBUTE AND TYPE NAMES MUST BE IN THE ENGLISH LANGUAGE, USING THE PRIMARY ENGLISH SPELLINGS PROVIDED IN THE *OXFORD ENGLISH DICTIONARY*.

3446

[R 5] LOWER-CAMEL-CASE (LCC) MUST BE USED FOR NAMING ATTRIBUTES.

3447

[R 6] UPPER-CAMEL-CASE (UCC) MUST BE USED FOR NAMING ELEMENTS AND TYPES.

3448

[R 7] NAMES MUST BE IN SINGULAR FORM UNLESS THE CONCEPT ITSELF IS PLURAL.

3449
3450

[R 8] NAMES MUST NOT CONTAIN NON-LETTER CHARACTERS, UNLESS REQUIRED BY LANGUAGE-SPECIFIC RULES.

3451
3452
3453

[R 9] XML NAMES CONSTRUCTED FROM DICTIONARY ENTRY NAMES MUST NOT INCLUDE PERIODS, SPACES, OR OTHER SEPARATORS; OR CHARACTERS NOT ALLOWED BY W3C XML 1.0 FOR XML NAMES.

3454
3455
3456

[R 10] ELEMENT AND `XSD:SIMPLE` AND `XSD:COMPLEXTYPE` NAMES MUST NOT USE ACRONYMS, ABBREVIATIONS, OR OTHER WORD TRUNCATIONS, EXCEPT THOSE INCLUDED IN THE UN/CEFACT CONTROLLED VOCABULARY.

3457
3458

[R 11] ALL ELEMENT DECLARATIONS FOR BBIES AND ASBIES MUST BE LOCALLY DECLARED WITHIN THE PARENT ABIE TYPE.

3459
3460
3461
3462
3463
3464

[R 12] EACH ELEMENT NAME DECLARATION MUST BE BASED ON THE PROPERTY TERM AND QUALIFIERS AND THE REPRESENTATION TERM OF THE BBIE OR THE PROPERTY TERM AND OBJECT CLASS OF THE ASBIE. IF THERE ARE SUCCESSIVE DUPLICATE WORDS IN THE PROPERTY TERM AND REPRESENTATION TERMS OF THE SOURCE DICTIONARY ENTRY NAME WHICH REPRESENT EXACTLY THE SAME SEMANTICS, THEN THE DUPLICATE WORDS MUST BE REMOVED.

3465
3466

[R 13] A ROOT SCHEMA MUST BE CREATED FOR EACH UNIQUE BUSINESS INFORMATION EXCHANGE.

3467
3468
3469

[R 14] A ROOT SCHEMA MUST NOT REPLICATE REUSABLE CONSTRUCTS AVAILABLE IN SCHEMA MODULES CAPABLE OF BEING REFERENCED THROUGH `XSD:INCLUDE` OR `XSD:IMPORT`.

3470 [R 15] UN/CEFACT SCHEMA MODULES MUST EITHER BE TREATED AS EXTERNAL SCHEMA
3471 MODULES OR AS INTERNAL SCHEMA MODULES OF THE ROOT SCHEMA.

3472 [R 16] ROOT SCHEMAS MUST ONLY DEFINE A SINGLE COMPLEX TYPE THAT FULLY
3473 DESCRIBES THE BUSINESS INFORMATION EXCHANGE.

3474 [R 17] ALL UN/CEFACT INTERNAL SCHEMA MODULES MUST BE IN THE SAME NAMESPACE
3475 AS THEIR CORRESPONDING RSM:ROOTSCHEMA.

3476 [R 18] EACH UN/CEFACT INTERNAL SCHEMA MODULE MUST BE NAMED
3477 {PARENTROOTSCHEMAMODULENAME} {INTERNALSCHEMAMODULEFUNCTION} {SCHEMA MODULE}

3478 [R 19] A CORE COMPONENT TYPE SCHEMA MODULE MUST BE CREATED

3479 [R 20] THE CCT: CORECOMPONENTTYPE SCHEMA MODULE MUST BE NAMED "CCTS CCT
3480 SCHEMA MODULE"

3481 [R 21] AN UNQUALIFIED DATA TYPE SCHEMA MODULE MUST BE CREATED

3482 [R 22] THE UDT: UNQUALIFIEDDATATYPE SCHEMA MODULE MUST BE NAMED "UN/CEFACT
3483 UNQUALIFIED DATA TYPE SCHEMA MODULE"

3484 [R 23] A QUALIFIED DATA TYPE SCHEMA MODULE MUST BE CREATED

3485 [R 24] THE QDT: QUALIFIEDDATATYPE SCHEMA MODULE MUST BE NAMED "UN/CEFACT
3486 QUALIFIED DATA TYPE SCHEMA MODULE"

3487 [R 25] AN AGGREGATE BUSINESS INFORMATION ENTITY SCHEMA MODULE MUST BE
3488 CREATED

3489 [R 26] THE RAM: REUSABLEAGGREGATEBUSINESSINFORMATIONENTITY SCHEMA MODULE MUST
3490 BE NAMED "UN/CEFACT AGGREGATE BUSINESS INFORMATION ENTITY SCHEMA MODULE"

3491 [R 27] REUSABLE CODE LIST SCHEMA MODULES MUST BE CREATED TO CONVEY CODE LIST
3492 ENUMERATIONS

3493 [R 28] THE NAME OF EACH CLT: CODELIST SCHEMA MODULE MUST BE OF THE FORM: <CODE
3494 LIST AGENCY NAME><CODE LIST NAME> - CODE LIST SCHEMA MODULE WHERE: CODE LIST
3495 AGENCY NAME = AGENCY THAT MAINTAINS THE CODE LIST CODE LIST NAME = THE NAME OF
3496 THE CODE LIST AS ASSIGNED BY THE AGENCY THAT MAINTAINS THE CODE LIST

3497 [R 29] AN IDENTIFIER LIST SCHEMA MODULE MUST BE CREATED FOR EACH IDENTIFIER LIST
3498 THAT MIMICS CODE LIST FUNCTIONALITY TO CONVEY ENUMERATIONS OF THE IDENTIFIER
3499 LIST VALUE AND TOKEN FOR THAT VALUE

3500 [R 30] THE NAME OF EACH UN/CEFACT IDENTIFIER LIST SCHEMA MODULE MUST BE OF THE
3501 FORM: <IDENTIFICATION SCHEME AGENCY NAME><IDENTIFICATION SCHEME NAME> -
3502 IDENTIFIER LIST SCHEMA MODULE WHERE: IDENTIFICATION SCHEME AGENCY NAME =
3503 AGENCY THAT MAINTAINS THE IDENTIFIER LIST IDENTIFICATION SCHEME NAME = NAME AS
3504 ASSIGNED BY THE AGENCY THAT MAINTAINS THE IDENTIFIER LIST

3505 [R 31] IMPORTED SCHEMA MODULES MUST BE FULLY CONFORMANT WITH UN/CEFACT
3506 NAMING AND DESIGN RULES AND THE CORE COMPONENTS TECHNICAL SPECIFICATION.

3507 [R 32] EVERY UN/CEFACT DEFINED OR USED SCHEMA MODULE MUST HAVE A NAMESPACE
3508 DECLARED, USING THE XSD:TARGETNAMESPACE ATTRIBUTE.

3509 [R 33] EVERY DEFINED OR USED SCHEMA MODULE VERSION OTHER THAN INTERNAL
3510 SCHEMA MODULES MUST HAVE ITS OWN UNIQUE NAMESPACE.

3511 [R 34] UN/CEFACT PUBLISHED NAMESPACE DECLARATIONS OR CONTENTS MUST NEVER BE
3512 CHANGED.

3513 [R 35] UN/CEFACT NAMESPACES MUST BE DEFINED AS URNS

3514 [R 36] THE NAMES FOR NAMESPACES MUST HAVE THE FOLLOWING STRUCTURE WHILE THE
3515 SCHEMAS ARE AT DRAFT STATUS:
3516 URN:UN:UNECE:UNCEFACT:<SCHEMATYPE>:DRAFT:<NAME>:<MAJOR>:[<MINOR>]:[<REVISION>]
3517 WHERE: SCHEMATYPE = A TOKEN IDENTIFYING THE TYPE OF SCHEMA MODULE:
3518 DATA|PROCESS|CODELIST NAME = THE NAME OF THE MODULE (USING UNDERSCORE AS
3519 SEPARATOR) MAJOR = THE MAJOR VERSION NUMBER. SEQUENTIALLY ASSIGNED, FIRST
3520 RELEASE STARTING WITH THE NUMBER 1. MINOR = THE MINOR VERSION NUMBER WITHIN A
3521 MAJOR RELEASE. SEQUENTIALLY ASSIGNED, FIRST RELEASE STARTING WITH THE NUMBER
3522 0. NOT APPLICABLE FOR CODELIST SCHEMA. REVISION = SEQUENTIALLY ASSIGNED
3523 ALPHANUMERIC CHARACTER FOR EACH REVISION OF A MINOR RELEASE. ONLY APPLICABLE
3524 WHERE STATUS = DRAFT AND SCHEMA TYPE DOES NOT EQUAL CODELIST.

3525 [R 37] THE NAMESPACE NAMES FOR SCHEMAS HOLDING SPECIFICATION STATUS MUST BE
3526 OF THE FORM:
3527 URN:UN:UNECE:UNCEFACT:<SCHEMATYPE>:STANDARD:<NAME>:<MAJOR>:[<MINOR>]
3528 WHERE: SCHEMATYPE = A TOKEN IDENTIFYING THE TYPE OF SCHEMA MODULE:
3529 DATA|PROCESS|CODELIST NAME = THE NAME OF THE MODULE (USING UNDERSCORE AS
3530 SEPARATOR) MAJOR = THE MAJOR VERSION NUMBER, SEQUENTIALLY ASSIGNED, FIRST
3531 RELEASE STARTING WITH THE NUMBER 1. MINOR = THE MINOR VERSION NUMBER WITHIN A
3532 MAJOR RELEASE, SEQUENTIALLY ASSIGNED, FIRST RELEASE STARTING WITH THE NUMBER
3533 0. NOT APPLICABLE FOR CODELIST SCHEMA.

3534 [R 38] UN/CEFACT NAMESPACES MUST ONLY CONTAIN UN/CEFACT DEVELOPED SCHEMA
3535 MODULES.

3536 [R 39] THE GENERAL STRUCTURE FOR SCHEMA LOCATION MUST BE:
3537 HTTP://WWW.UNECE.ORG/UNCEFACT/<SCHEMATYPE>/<NAME>_<MAJOR>.<MINOR>.
3538 [<REVISION>]_[<STATUS>].XSD WHERE: SCHEMATYPE = A TOKEN IDENTIFYING THE TYPE OF
3539 SCHEMA MODULE: DATA|PROCESS|CODELIST NAME = THE NAME OF THE MODULE (USING
3540 UNDERSCORE AS SEPARATOR) MAJOR = THE MAJOR VERSION NUMBER, SEQUENTIALLY
3541 ASSIGNED, FIRST RELEASE STARTING WITH THE NUMBER 1. MINOR = THE MINOR VERSION
3542 NUMBER WITHIN A MAJOR RELEASE, SEQUENTIALLY ASSIGNED, FIRST RELEASE STARTING
3543 WITH THE NUMBER 0. REVISION = SEQUENTIALLY ASSIGNED ALPHANUMERIC CHARACTER
3544 FOR EACH REVISION OF A MINOR RELEASE. ONLY APPLICABLE WHERE STATUS = DRAFT.
3545 STATUS = THE STATUS OF THE SCHEMA AS: DRAFT|STANDARD

3546 [R 40] EACH XSD:SCHEMALOCATION ATTRIBUTE DECLARATION MUST CONTAIN A
3547 PERSISTENT AND RESOLVABLE URL.

- 3548 [R 41] EACH XSD:SCHEMALOCATION ATTRIBUTE DECLARATION URL MUST CONTAIN AN
3549 ABSOLUTE PATH.
- 3550 [R 42] SCHEMA MODULES MUST BE LOCATED UNDER THE DIRECTORY:
3551 HTTP://WWW.UNECE.ORG/UNCEFACT/CC/SCHEMA/<SCHEMA-MOD-NAME>.XSD
- 3552 [R 43] A UN/CEFACT NAMESPACE URN IS DIVIDED INTO THREE PARTS. FIRST, IS THE
3553 STANDARD UN/CEFACT NAMESPACE INFORMATION. SECOND, IS THE DESCRIPTION OF THE
3554 PURPOSE OF THE NAMESPACE. THIRD, IS THE VERSION INFORMATION. THE VERSION
3555 INFORMATION WILL IN TURN BE DIVIDED INTO MAJOR (OR INCOMPATIBLE) AND MINOR (OR
3556 COMPATIBLE) FIELDS. THE MINOR FIELD HAS AN OPTIONAL REVISION EXTENSION.
- 3557 [R 44] EVERY SCHEMA MAJOR VERSION MUST HAVE THE URI OF:
3558 URN:UN:UNECE:UNCEFACT:<SCHEMATYPE>:<STATUS>:<NAME>:<MAJOR>:0:[<REVISION>]
- 3559 [R 45] EVERY UN/CEFACT SCHEMA AND SCHEMA MODULE MAJOR VERSION NUMBER MUST
3560 BE A SEQUENTIALLY ASSIGNED INCREMENTAL INTEGER GREATER THEN ZERO.
- 3561 [R 46] MINOR VERSIONING MUST BE LIMITED TO DECLARING NEW OPTIONAL XSD
3562 CONSTRUCTS, EXTENDING EXISTING XSD CONSTRUCTS AND REFINEMENTS OF AN OPTIONAL
3563 NATURE.
- 3564 [R 47] EVERY UN/CEFACT SCHEMA MINOR VERSION MUST HAVE THE URI OF:
3565 URN:UN:UNECE:UNCEFACT:CC:SCHEMA:<NAME>:<MAJOR-NUMBER>:<NON-ZERO
3566 INTEGER>:[<REVISION>]
- 3567 [R 48] FOR UN/CEFACT MINOR VERSION CHANGES, THE NAME OF THE SCHEMA CONSTRUCT
3568 MUST NOT CHANGE
- 3569 [R 49] CHANGES IN MINOR VERSIONS MUST NOT BREAK SEMANTIC COMPATIBILITY WITH
3570 PRIOR VERSIONS.
- 3571 [R 50] UN/CEFACT MINOR VERSION SCHEMA MUST INCORPORATE ALL XML CONSTRUCTS
3572 FROM THE IMMEDIATELY PRECEDING MAJOR OR MINOR VERSION SCHEMA.
- 3573 [R 51] ALL UN/CEFACT SCHEMA DESIGN RULES MUST BE BASED ON THE W3C XML SCHEMA
3574 RECOMMENDATIONS: XML SCHEMA PART 1: STRUCTURES AND XML SCHEMA PART 2:
3575 DATATYPES.
- 3576 [R 52] ALL SCHEMATA AND INSTANCES MUST BE BASED ON THE W3C SUITE OF TECHNICAL
3577 SPECIFICATIONS HOLDING RECOMMENDATION STATUS.
- 3578 [R 53] ELEMENT FORM DEFAULT MUST BE DECLARED AS QUALIFIED.
- 3579 [R 54] ATTRIBUTE FORM DEFAULT MUST BE DECLARED AS UNQUALIFIED.
- 3580 [R 55] THE "XSD" PREFIX MUST BE USED.
3581 XMLNS:XSD=HTTP://WWW.W3.ORG/2001/XMLSCHEMA
- 3582 [R 56] THE XSI PREFIX SHALL BE USED WHERE APPROPRIATE.

- 3583 [R 57] PROCESSING INSTRUCTIONS MUST NOT BE USED.
- 3584 [R 58] NOTATIONS MUST NOT BE USED.
- 3585 [R 59] WILDCARDS MUST NOT BE USED.
- 3586 [R 60] THE XSD:ANY ELEMENT MUST NOT BE USED.
- 3587 [R 61] THE XSD:ANY ATTRIBUTE MUST NOT BE USED.
- 3588 [R 62] MIXED CONTENT MUST NOT BE USED (EXCLUDING DOCUMENTATION).
- 3589 [R 63] SUBSTITUTION GROUPS MUST NOT BE USED.
- 3590 [R 64] ID/IDREF MUST NOT BE USED.
- 3591 [R 65] KEY/KEYREF MUST BE USED FOR INFORMATION ASSOCIATION.
- 3592 [R 66] THE ABSENCE OF A CONSTRUCT OR DATA MUST NOT CARRY MEANING.
- 3593 [R 67] USER DEFINED ATTRIBUTES MUST ONLY BE USED TO CONVEY CORE COMPONENT
3594 TYPE (CCT) SUPPLEMENTARY COMPONENT INFORMATION.
- 3595 [R 68] THE NILLABLE ATTRIBUTE MUST NOT BE USED.
- 3596 [R 69] ALL ELEMENT DECLARATIONS MUST BE LOCAL EXCEPT FOR A ROOT ELEMENT THAT
3597 MUST BE DECLARED GLOBALLY.
- 3598 [R 70] EMPTY ELEMENTS MUST NOT BE USED.
- 3599 [R 72] THE ELEMENT DECLARATION OF XSI:NIL MUST NOT APPEAR IN ANY CONFORMING
3600 INSTANCE.
- 3601 [R 73] THE XSD:ALL ELEMENT MUST NOT BE USED.
- 3602 [R 74] MIXED-CONTENT ELEMENTS MUST NOT BE USED.
- 3603 [R 75] ALL TYPE DEFINITIONS MUST BE NAMED.
- 3604 [R 76] TYPE DEFINITIONS MUST NOT DUPLICATE THE FUNCTIONALITY OF EXISTING
3605 XSD:BUILT-IN SIMPLE TYPES.
- 3606 [R 77] USER DEFINED SIMPLE TYPES DEFINITIONS MUST NOT BE USED.
- 3607 [R 78] EXTENSION MUST ONLY BE USED IN THE CCT: CORECOMPONENTTYPES SCHEMA
3608 MODULE AND THE UDT: UNQUALIFIEDDATATYPE SCHEMA MODULE. WHEN USED IT MUST ONLY
3609 EXTEND A BUILT-IN XSD:DATATYPE.

- 3610 [R 79] THE ROOT SCHEMA MODULE MUST BE REPRESENTED BY THE TOKEN “RSM”.
- 3611 [R 80] THE RSM:ROOTSCHEMA MUST IMPORT THE FOLLOWING SCHEMA MODULES: –
3612 RAM:REUSABLEABIE SCHEMA MODULE – UDT:UNQUALIFIEDDATATYPES MODULE –
3613 QDT:QUALIFIEDDATATYPES MODULE
- 3614 [R 81] A RSM:ROOTSCHEMA IN ONE UN/CEFACT NAMESPACE THAT IS DEPENDENT UPON TYPE
3615 DEFINITIONS OR ELEMENT DECLARATION DEFINED IN ANOTHER NAMESPACE MUST IMPORT
3616 THE RSM:ROOTSCHEMA FROM THAT NAMESPACE.
- 3617 [R 82] A RSM:ROOTSCHEMA IN ONE UN/CEFACT NAMESPACE THAT IS DEPENDANT UPON TYPE
3618 DEFINITIONS OR ELEMENT DECLARATIONS DEFINED IN ANOTHER NAMESPACE MUST NOT
3619 IMPORT SCHEMA MODULES FROM THAT NAMESPACE OTHER THEN THE RSM:ROOTSCHEMA.
- 3620 [R 83] THE RSM:ROOTSCHEMA MUST INCLUDE ANY INTERNAL SCHEMA MODULES THAT
3621 RESIDE IN THE ROOT SCHEMA NAMESPACE.
- 3622 [R 84] A SINGLE GLOBAL ELEMENT KNOWN AS THE ROOT ELEMENT MUST BE GLOBALLY
3623 DECLARED IN A RSM:ROOTSCHEMA.
- 3624 [R 85] THE ROOT ELEMENT MUST REFLECT THE MESSAGE ASSEMBLY THAT IS DEFINED
3625 FOR THE ACTUAL CONTENT OF THE BUSINESS INFORMATION.
- 3626 [R 86] THE NAME OF THE ROOT ELEMENT MUST BE THE NAME OF THE MESSAGE ASSEMBLY
3627 WITH SEPARATORS AND SPACES REMOVED.
- 3628 [R 87] FOR THE ROOT ELEMENT A CORRESPONDING COMPLEX TYPE THAT REPRESENTS
3629 THE MESSAGE ASSEMBLY MUST BE DEFINED.
- 3630 [R 88] THE NAME OF THE TOP-LEVEL COMPLEX TYPE MUST BE THE NAME OF THE ROOT
3631 ELEMENT WITH THE WORD “TYPE” APPENDED.
- 3632 [R 89] FOR EVERY RSM:ROOTSCHEMA ROOT ELEMENT DECLARATION A STRUCTURED SET OF
3633 ANNOTATIONS MUST BE PRESENT IN THE FOLLOWING PATTERN:
- 3634 [R 90] ALL UN/CEFACT INTERNAL SCHEMA MODULES MUST BE IN THE SAME NAMESPACE
3635 AS THEIR CORRESPONDING RSM:ROOTSCHEMA.
- 3636 [R 91] THE INTERNAL SCHEMA MODULE MUST BE REPRESENTED BY THE SAME TOKEN AS
3637 ITS RSM:ROOTSCHEMA.
- 3638 [R 92] THE SCHEMA MODULE MUST BE REPRESENTED BY THE TOKEN “RAM”.
- 3639 [R 93] THE RAM:REUSABLEAGGREGATEBUSINESSINFORMATIONENTITY SCHEMA MUST IMPORT
3640 THE FOLLOWING SCHEMA MODULES: – UDT:UNQUALIFIEDDATATYPE MODULE –
3641 QDT:QUALIFIEDDATATYPE MODULE
- 3642 [R 94] FOR EVERY OBJECT CLASS (ABIE) IDENTIFIED IN THE UN/CEFACT SYNTAX-NEUTRAL
3643 MODEL, A NAMED XSD:COMPLEXTYPE MUST BE DEFINED.

3644 [R 95] THE NAME OF THE ABIE `XSD:COMPLEXTYPE` MUST BE THE
3645 `CCTS:DICTIONARYENTRYNAME` WITH THE SEPARATORS REMOVED AND WITH THE "DETAILS"
3646 SUFFIX REPLACED WITH "TYPE".

3647 [R 96] EVERY AGGREGATE BUSINESS INFORMATION ENTITY (ABIE) `XSD:COMPLEXTYPE`
3648 DEFINITION `XSD:CONTENT` MODEL MUST USE THE `XSD:SEQUENCE` AND/OR `XSD:CHOICE`
3649 ELEMENTS WITH APPROPRIATE LOCAL ELEMENT DECLARATIONS TO REFLECT EACH
3650 PROPERTY (BBIE OR ASBIE) OF ITS CLASS.

3651 [R 97] RECURSION OF `XSD:SEQUENCE` AND/OR `XSD:CHOICE` MUST NOT OCCUR.

3652 [R 98] THE ORDER AND CARDINALITY OF THE ELEMENTS WITHIN AN ABIE `XSD:COMPLEXTYPE`
3653 MUST BE ACCORDING TO THE STRUCTURE OF THE ABIE AS DEFINED IN THE MODEL.

3654 [R 99] FOR EVERY ATTRIBUTE OF AN OBJECT CLASS (BBIE) IDENTIFIED IN THE UN/CEFACT
3655 SYNTAX-NEUTRAL MODEL, A NAMED `XSD:ELEMENT` MUST BE LOCALLY DECLARED WITHIN
3656 THE `XSD:COMPLEXTYPE` REPRESENTING THE ABIE.

3657 [R 100] EACH BBIE ELEMENT NAME DECLARATION MUST BE BASED ON THE PROPERTY
3658 TERM AND QUALIFIERS AND THE REPRESENTATION TERM OF THE BASIC BUSINESS
3659 INFORMATION ENTITY (BBIE). IF THERE ARE DUPLICATE WORDS IN THE PROPERTY TERM AND
3660 REPRESENTATION TERMS OF THE SOURCE DICTIONARY ENTRY NAME, THEN THE DUPLICATE
3661 WORDS MUST BE REMOVED.

3662 [R 101] THE BBIE ELEMENT MUST BE BASED ON AN APPROPRIATE DATA TYPE THAT IS
3663 DEFINED IN THE UN/CEFACT `QDT:QUALIFIEDDATATYPE` OR `UDT:UNQUALIFIEDDATATYPE`
3664 SCHEMA MODULES.

3665 [R 102] FOR EVERY ASSOCIATION (ASBIE) IDENTIFIED IN THE UN/CEFACT SYNTAX-
3666 NEUTRAL MODEL, A NAMED `XSD:ELEMENT` MUST BE LOCALLY DECLARED WITHIN THE
3667 `XSD:COMPLEXTYPE` REPRESENTING THE ABIE.

3668 [R 103] EACH ASBIE ELEMENT NAME DECLARATION MUST BE BASED ON THE PROPERTY
3669 TERM AND OBJECT CLASS OF THE ASSOCIATION BUSINESS INFORMATION ENTITY (ASBIE). IF
3670 THERE ARE DUPLICATE WORDS IN THE PROPERTY TERM AND REPRESENTATION TERMS OF
3671 THE SOURCE DICTIONARY ENTRY NAME, THEN THE DUPLICATE WORDS MUST BE REMOVED.

3672 [R 104] THE ELEMENT REPRESENTING AN ASSOCIATION BUSINESS INFORMATION ENTITY
3673 (ASBIE) MUST BE OF THE COMPLEX TYPE CORRESPONDING TO ITS ASSOCIATED
3674 AGGREGATE BUSINESS INFORMATION (ABIE).

3675 [R 105] EACH `XSD:COMPLEXTYPE` AND `XSD:ELEMENT` DEFINITION MUST USE THE
3676 `XSD:ANNOTATION` ELEMENT.

3677 [R 106] A `XSD:ANNOTATION` ELEMENT DECLARATION MUST APPEAR IMMEDIATELY AFTER
3678 THE STARTING TAG OF THE `XSD:COMPLEXTYPE` OR `XSD:ELEMENT`.

3679 [R 107] FOR EVERY ABIE `XSD:COMPLEXTYPE` DEFINITION A STRUCTURED SET OF
3680 ANNOTATIONS MUST BE PRESENT IN THE FOLLOWING PATTERN:

3681 [R 108] FOR EVERY BBIE `XSD:ELEMENT` DECLARATION A STRUCTURED SET OF
3682 ANNOTATIONS MUST BE PRESENT IN THE FOLLOWING PATTERN:

3683 [R 109] FOR EVERY ASBIE XSD: ELEMENT DECLARATION A STRUCTURED SET OF
3684 ANNOTATIONS MUST BE PRESENT IN THE FOLLOWING PATTERN:

3685 [R 110] THE CORE COMPONENT TYPE (CCT) SCHEMA MODULE MUST BE REPRESENTED BY
3686 THE TOKEN "CCT".

3687 [R 111] EVERY CCT: CORECOMPONENTTYPE MUST BE DEFINED AS A NAMED XSD: COMPLEXTYPE
3688 IN THE CCT: CORECOMPONENTTYPE SCHEMA MODULE.

3689 [R 112] THE NAME OF EACH XSD: COMPLEXTYPE BASED ON A CCT: CORECOMPONENTTYPE
3690 MUST BE THE DICTIONARY ENTRY NAME OF THE CORE COMPONENT TYPE (CCT), WITH THE
3691 SEPARATORS AND SPACES REMOVED.

3692 [R 113] EACH CCT: CORECOMPONENTTYPE XSD: COMPLEXTYPE DEFINITION MUST CONTAIN ONE
3693 XSD: SIMPLECONTENT ELEMENT.

3694 [R 114] THE CCT: CORECOMPONENTTYPE XSD: COMPLEXTYPE DEFINITION XSD: SIMPLECONTENT
3695 ELEMENT MUST CONTAIN ONE XSD: EXTENSION ELEMENT. THIS XSD: EXTENSION ELEMENT
3696 MUST INCLUDE AN XSD BASED ATTRIBUTE THAT DEFINES THE SPECIFIC XSD: BUILT-IN DATA
3697 TYPE REQUIRED FOR THE CCT CONTENT COMPONENT.

3698 [R 115] WITHIN THE CCT: CORECOMPONENTTYPE XSD: EXTENSION ELEMENT A XSD: ATTRIBUTE
3699 MUST BE DECLARED FOR EACH SUPPLEMENTARY COMPONENT PERTAINING TO THAT
3700 CCT: CORECOMPONENTTYPE.

3701 [R 116] A XSD: ATTRIBUTE MUST BE DECLARED FOR EACH SUPPLEMENTARY COMPONENT.

3702 [R 117] EACH SUPPLEMENTARY COMPONENT XSD: ATTRIBUTE "NAME" MUST BE THE
3703 SUPPLEMENTARY COMPONENT NAME WITH THE SEPARATORS AND SPACES REMOVED.

3704 [R 118] EACH XSD: COMPLEXTYPE DEFINITION MUST USE THE XSD: ANNOTATION ELEMENT.

3705 [R 119] AN XSD: ANNOTATION ELEMENT DECLARATION MUST APPEAR IMMEDIATELY AFTER
3706 THE STARTING TAG OF THE XSD: COMPLEXTYPE.

3707 [R 120] FOR EVERY CCT: CORECOMPONENTTYPE XSD: COMPLEXTYPE DEFINITION A
3708 STRUCTURED SET OF ANNOTATIONS MUST BE PRESENT IN THE FOLLOWING PATTERN:

3709 [R 121] FOR EVERY SUPPLEMENTARY COMPONENT XSD: ATTRIBUTE DECLARATION A
3710 STRUCTURED SET OF ANNOTATIONS MUST BE PRESENT IN THE FOLLOWING PATTERN:

3711 [R 122] THE UNQUALIFIED DATA TYPES SCHEMA MODULE NAMESPACE MUST BE
3712 REPRESENTED BY THE TOKEN "UDT".

3713 [R 123] THE UDT: UNQUALIFIEDDATATYPE SCHEMA MUST IMPORT THE FOLLOWING SCHEMA
3714 MODULES: – IDS: IDENTIFIERLIST SCHEMA MODULES – CLM: CODELIST SCHEMA MODULES

3715 [R 124] A UDT: UNQUALIFIEDDATATYPE MUST BE DEFINED FOR EACH APPROVED PRIMARY
3716 AND SECONDARY REPRESENTATION TERMS IDENTIFIED IN THE CCTS PERMISSABLE
3717 REPRESENTATION TERMS TABLE.

3718 [R 125] THE NAME OF EACH UDT:UNQUALIFIEDDATATYPE MUST BE THE DICTIONARY ENTRY
3719 NAME OF THE PRIMARY OR SECONDARY REPRESENTATION TERM, WITH "TYPE" AT THE END
3720 AND THE SEPARATORS AND SPACES REMOVED.

3721 [R 126] FOR EVERY UDT:UNQUALIFIEDDATATYPE WHOSE SUPPLEMENTARY COMPONENTS
3722 MAP DIRECTLY TO THE PROPERTIES OF A BUILT-IN XSD:DATA TYPE, THE
3723 UDT:UNQUALIFIEDDATATYPE MUST BE DEFINED AS A NAMED XSD:SIMPLETYPE IN THE
3724 UDT:UNQUALIFIEDDATATYPE SCHEMA MODULE.

3725 [R 127] EVERY UDT:UNQUALIFIEDDATATYPE DEFINED AS A XSD:SIMPLETYPE MUST CONTAIN
3726 ONE XSD:RESTRICTION ELEMENT. THIS XSD:RESTRICTION ELEMENT MUST INCLUDE AN
3727 XSD:BASE ATTRIBUTE THAT DEFINES THE SPECIFIC BUILT-IN XSD:DATATYPE REQUIRED FOR
3728 THE CONTENT COMPONENT.

3729 [R 128] FOR EVERY UDT:UNQUALIFIEDDATATYPE WHOSE SUPPLEMENTARY COMPONENTS
3730 ARE NOT EQUIVALENT TO THE PROPERTIES OF A XSD:BUILT-IN DATA TYPE, A
3731 UDT:UNQUALIFIEDDATATYPE MUST BE DEFINED AS AN XSD:COMPLEXTYPE IN THE
3732 UDT:UNQUALIFIEDDATATYPE SCHEMA MODULE.

3733 [R 129] EVERY UDT:UNQUALIFIEDDATATYPE XSD:COMPLEXTYPE DEFINITION MUST CONTAIN
3734 ONE XSD:SIMPLECONTENT ELEMENT.

3735 [R 130] EVERY UDT:UNQUALIFIEDDATATYPE XSD:COMPLEXTYPE XSD:SIMPLECONTENT
3736 ELEMENT MUST CONTAIN ONE XSD:EXTENSION ELEMENT. THIS XSD:EXTENSION ELEMENT
3737 MUST INCLUDE AN XSD:BASE ATTRIBUTE THAT DEFINES THE SPECIFIC XSD:BUILT-IN DATA
3738 TYPE REQUIRED FOR THE CONTENT COMPONENT.

3739 [R 131] WITHIN THE UDT:UNQUALIFIEDDATATYPE XSD:COMPLEXTYPE XSD:EXTENSION
3740 ELEMENT AN XSD:ATTRIBUTE MUST BE DECLARED FOR EACH SUPPLEMENTARY COMPONENT
3741 PERTAINING TO THE UNDERLYING CCT, UNLESS THE ATTRIBUTE IS CONTAINED IN THE
3742 NAMESPACE DECLARATION.

3743 [R 132] EACH CCT CODE TYPE OR CCT IDENTIFIER TYPE SUPPLEMENTARY COMPONENT
3744 THAT HAS A RESTRICTION IDENTIFIED IN CCTS, MUST HAVE THE NAME OF THE CCT
3745 SUPPLEMENTARY COMPONENT WITH THE PRIMARY REPRESENTATION TERM REMOVED AT
3746 THE BEGINNING. THE XSD:TYPE FOR THIS XSD:ATTRIBUTE MUST BE THE NAME OF THE CODE
3747 LIST OR IDENTIFIER SCHEME CONTAINING THE RESTRICTED SET OF VALUES. THE
3748 XSD:ATTRIBUTE MUST CONTAIN AN XSD:USE ATTRIBUTE WHO'S VALUE MUST BE SET TO
3749 REQUIRED.

3750 [R 133] EACH SUPPLEMENTARY COMPONENT XSD:ATTRIBUTE NAME MUST BE THE
3751 SUPPLEMENTARY COMPONENT NAME WITH THE SEPARATORS AND SPACES REMOVED.

3752 [R 134] EACH UDT:UNQUALIFIEDDATATYPE XSD:COMPLEXTYPE AND XSD:SIMPLETYPE
3753 DEFINITION MUST USE THE XSD:ANNOTATION ELEMENT.

3754 [R 135] AN XSD:ANNOTATION ELEMENT DECLARATION MUST APPEAR IMMEDIATELY AFTER
3755 THE STARTING TAG OF THE XSD:COMPLEXTYPE OR XSD:SIMPLETYPE.

3756 [R 136] FOR EVERY UDT:UNQUALIFIEDDATATYPE XSD:COMPLEXTYPE OR XSD:SIMPLETYPE
3757 DEFINITION A STRUCTURED SET OF ANNOTATIONS MUST BE PRESENT IN THE FOLLOWING
3758 PATTERN:

3759 [R 137] FOR EVERY SUPPLEMENTARY COMPONENT `XSD:ATTRIBUTE` DECLARATION A
3760 STRUCTURED SET OF ANNOTATIONS MUST BE PRESENT IN THE FOLLOWING PATTERN:

3761 [R 138] THE `UN/CEFACT:QUALIFIEDDATATYPE` SCHEMA MODULE NAMESPACE MUST BE
3762 REPRESENTED BY THE TOKEN "QDT".

3763 [R 139] THE `QDT:QUALIFIEDDATATYPE` SCHEMA MODULE MUST IMPORT THE –
3764 `UDT:UNQUALIFIEDDATATYPE` SCHEMA MODULE – `IDS:IDENTIFIERLIST` SCHEMA MODULES –
3765 `CLM:CODELIST` SCHEMA MODULE

3766 [R 140] WHERE REQUIRED TO CHANGE FACETS OF AN EXISTING
3767 `UDT:UNQUALIFIEDDATATYPE`, A NEW DATA TYPE MUST BE DEFINED IN THE
3768 `QDT:QUALIFIEDDATATYPE` SCHEMA MODULE.

3769 [R 141] IF A `QDT:QUALIFIEDDATATYPE` IS BASED ON AN `UDT:UNQUALIFIEDDATATYPE` THAT IS
3770 AN `XSD:SIMPLETYPE`, THE `QDT:QUALIFIEDDATATYPE` MUST ALSO BE AN `XSD:SIMPLETYPE`.

3771 [R 142] IF A `QDT:QUALIFIEDDATATYPE` IS BASED ON AN `UDT:UNQUALIFIEDDATATYPE` THAT IS
3772 A `XSD:COMPLEXTYPE`, THE `QDT:QUALIFIEDDATATYPE` MUST ALSO BE AN `XSD:COMPLEXTYPE`.

3773 [R 143] IF A `QDT:QUALIFIEDDATATYPE`'S RESTRICTIONS MAP DIRECTLY TO THE
3774 PROPERTIES OF A `XSD:BUILT-IN` DATA TYPE, THE `QDT:QUALIFIEDDATATYPE` MUST BE
3775 DEFINED AS A NAMED `XSD:SIMPLETYPE` IN THE `QDT:QUALIFIEDDATATYPE` SCHEMA MODULE.

3776 [R 144] THE NAME OF A `QDT:QUALIFIEDDATATYPE` MUST BE THE DICTIONARY ENTRY NAME
3777 OF ITS BASE `UDT:UNQUALIFIEDDATATYPE` WITH AN APPROPRIATE QUALIFIER OR QUALIFIERS.

3778 [R 145] EACH `QDT:QUALIFIEDDATATYPE` `XSD:COMPLEXTYPE` DEFINITION MUST CONTAIN ONE
3779 `XSD:SIMPLECONTENT` ELEMENT.

3780 [R 146] THE `QDT:QUALIFIEDDATATYPE` `XSD:COMPLEXTYPE` DEFINITION `XSD:SIMPLECONTENT`
3781 ELEMENT MUST CONTAIN ONE `XSD:RESTRICTION` ELEMENT WITH AN `XSD:BASE` ATTRIBUTE
3782 WHOSE VALUE IS EQUAL TO AN `UDT:UNQUALIFIEDDATATYPE` OR AN EXTERNALLY IMPORTED
3783 CODE LIST.

3784 [R 147] THE `QDT:QUALIFIEDDATATYPE` `XSD:COMPLEXTYPE` DEFINITION `XSD:SIMPLECONTENT`
3785 ELEMENT MUST ONLY RESTRICT ATTRIBUTES DECLARED IN ITS BASE TYPE, OR MUST ONLY
3786 RESTRICT FACETS EQUIVALENT TO ALLOWED SUPPLEMENTARY COMPONENTS.

3787 [R 148] EACH `QDT:QUALIFIEDDATATYPE` `XSD:COMPLEXTYPE` AND `XSD:SIMPLETYPE`
3788 DEFINITION MUST USE THE `XSD:ANNOTATION` ELEMENT.

3789 [R 149] AN `XSD:ANNOTATION` ELEMENT DECLARATION MUST APPEAR IMMEDIATELY AFTER
3790 THE STARTING TAG OF THE `XSD:COMPLEXTYPE` OR `XSD:SIMPLETYPE`.

3791 [R 150] EVERY `QDT:QUALIFIEDDATATYPE` DEFINITION MUST CONTAIN A STRUCTURED SET
3792 OF ANNOTATIONS IN THE FOLLOWING SEQUENCE AND PATTERN:

3793 [R 151] FOR EVERY SUPPLEMENTARY COMPONENT `XSD:ATTRIBUTE` DECLARATION A
3794 STRUCTURED SET OF ANNOTATIONS MUST BE PRESENT IN THE FOLLOWING PATTERN:

3795 [R 152] INTERNAL CODE LISTS MUST NOT DUPLICATE EXISTING EXTERNAL CODE LISTS
3796 WHERE THEY ARE AVAILABLE IN A SCHEMA MODULE FORM THAT IS CAPABLE OF BEING
3797 IMPORTED.

3798 [R 153] EACH UN/CEFACT MAINTAINED CODE LIST MUST BE DEFINED IN ITS OWN SCHEMA
3799 MODULE.

3800 [R 154] EACH UN/CEFACT MAINTAINED CODE LIST SCHEMA MODULE MUST BE
3801 REPRESENTED BY A UNIQUE TOKEN CONSTRUCTED AS FOLLOWS: CLM[QUALIFIED DATA
3802 TYPE NAME]<CODE LIST. AGENCY. IDENTIFIER|CODE LIST. AGENCY NAME. TEXT><CODE
3803 LIST. IDENTIFICATION. IDENTIFIER|CODE LIST. NAME. TEXT> WITH ANY REPEATED
3804 WORDS ELIMINATED.

3805 [R 155] CODE LIST SCHEMA MODULES MUST NOT IMPORT OR INCLUDE ANY OTHER
3806 SCHEMA MODULES.

3807 [R 156] FOR THE TOP-LEVEL ELEMENT A CORRESPONDING TOP-LEVEL XSD:COMPLEXTYPE
3808 MUST BE DEFINED.

3809 [R 157] THE NAME FOR THE XSD:COMPLEXTYPE MUST BE BASED ON THE VALUE OF THE
3810 CODE LIST. NAME. TEXT. WITH THE WORD "TYPE" APPENDED.

3811 [R 158] EACH XSD:COMPLEXTYPE MUST CONTAIN ONE XSD:SIMPLECONTENT ELEMENT.

3812 [R 159] THE XSD:COMPLEXTYPE DEFINITION XSD:SIMPLECONTENT ELEMENT MUST CONTAIN
3813 ONE XSD:EXTENSION ELEMENT. THIS XSD:EXTENSION ELEMENT MUST INCLUDE AN XSD
3814 BASED ATTRIBUTE THAT DEFINES THE SPECIFIC LIST CONTAINING THE ACTUAL CODE
3815 VALUES.

3816 [R 160] WITHIN EACH CODE LIST MODULE ONE, AND ONLY ONE, NAMED XSD:SIMPLETYPE
3817 MUST BE DEFINED FOR THE CONTENT COMPONENT.

3818 [R 161] THE NAME OF THE XSD:SIMPLETYPE MUST BE THE NAME OF ROOT ELEMENT
3819 BASED ON THE VALUE OF THE CODE LIST. NAME. TEXT. WITH THE WORD "CONTENTTYPE"
3820 APPENDED.

3821 [R 162] THE XSD:RESTRICTION ELEMENT BASE ATTRIBUTE VALUE MUST BE SET TO
3822 "XSD:TOKEN".

3823 [R 163] EACH CODE IN THE CODE LIST MUST BE EXPRESSED AS AN XSD:ENUMERATION,
3824 WHERE THE XSD:VALUE FOR THE ENUMERATION IS THE ACTUAL CODE VALUE.

3825 [R 164] THE XSD FACET FEATURE MUST NOT BE USED IN THE CODE LIST SCHEMA
3826 MODULE.

3827 [R 165] FOR EACH CODE LIST A SINGLE ROOT ELEMENT MUST BE GLOBALLY DECLARED.

3828 [R 166] THE NAME OF ROOT ELEMENT MUST BE BASED ON THE CODE LIST. NAME. TEXT
3829 FOLLOWING THE NAMING RULES AS DEFINED IN SECTION 2.3.

3830 [R 167] THE ROOT ELEMENT MUST BE OF A TYPE REPRESENTING THE ACTUAL LIST OF
3831 CODE VALUES.

3832 [R 168] EACH `XSD:ENUMERATION` MUST INCLUDE AN ANNOTATION DOCUMENTATION
3833 PROVIDING THE CODE NAME AND THE CODE DESCRIPTION.

3834 [R 169] WHERE AN IDENTIFICATION SCHEME IS IDENTIFIED (CONTAINS AN IDENTIFIER, A
3835 TOKEN FOR THE IDENTIFIER, AND AN OPTIONALLY, A DESCRIPTION) TO HAVE THE SAME
3836 FUNCTIONALITY AS A CODE LIST AND A BUSINESS REQUIREMENT EXISTS FOR ENUMERATING
3837 THE IDENTIFIER LIST, A SCHEMA MODULE OF THE SAME FORM AS THE CODE LIST SCHEMA
3838 MODULE MUST BE CREATED.

3839 [R 170] INTERNAL IDENTIFIER LISTS MUST NOT NOT DUPLICATE EXISTING EXTERNAL
3840 IDENTIFIER LISTS WHERE THEY ARE AVAILABLE IN A SCHEMA MODULE FORM THAT IS
3841 CAPABLE OF BEING IMPORTED.

3842 [R 171] EACH UN/CEFACT MAINTAINED IDENTIFIER LIST MUST BE DEFINED IN ITS OWN
3843 SCHEMA MODULE.

3844 [R 172] EACH UN/CEFACT MAINTAINED IDENTIFIER LIST SCHEMA MODULE MUST BE
3845 REPRESENTED BY A UNIQUE TOKEN CONSTRUCTED AS FOLLOWS: `IDS[QUALIFIED DATA`
3846 `TYPE NAME]<IDENTIFICATION SCHEME AGENCY. IDENTIFIER><IDENTIFICATION SCHEME.`
3847 `IDENTIFIER>`

3848 [R 173] IDENTIFIER LIST SCHEMA MODULES MUST NOT IMPORT OR INCLUDE ANY OTHER
3849 SCHEMA MODULES.

3850 [R 174] WITHIN EACH IDENTIFIER LIST MODULE ONE, AND ONLY ONE, NAMED
3851 `XSD:SIMPLETYPE` MUST BE DEFINED FOR THE CONTENT COMPONENT.

3852 [R 175] THE NAME OF THE `XSD:SIMPLETYPE` MUST BE THE NAME OF ROOT ELEMENT WITH
3853 THE WORD "CONTENTTYPE" APPENDED.

3854 [R 176] THE `XSD:RESTRICTION` ELEMENT BASE ATTRIBUTE VALUE MUST BE SET TO
3855 "XSD:TOKEN".

3856 [R 177] EACH IDENTIFIER IN THE IDENTIFIER LIST MUST BE EXPRESSED AS AN
3857 `XSD:ENUMERATION`, WHERE THE `XSD:VALUE` FOR THE ENUMERATION IS THE ACTUAL
3858 IDENTIFIER VALUE.

3859 [R 178] THE XSD FACET FEATURE MUST NOT BE USED IN THE IDENTIFIER LIST SCHEMA
3860 MODULE.

3861 [R 179] FOR EACH IDENTIFIER LIST A SINGLE ROOT ELEMENT MUST BE GLOBALLY
3862 DECLARED.

3863 [R 180] THE NAME OF THE ROOT ELEMENT MUST BE BASED ON THE `IDENTIFICATION`
3864 `SCHEME. NAME. TEXT` FOLLOWING THE NAMING RULES AS DEFINED IN SECTION 2.3.

3865 [R 181] THE ROOT ELEMENT MUST BE OF A TYPE REPRESENTING THE ACTUAL LIST OF
3866 IDENTIFIER VALUES.

3867 [R 182] EACH `xsd:enumeration` MUST INCLUDE AN ANNOTATION DOCUMENTATION
3868 PROVIDING THE IDENTIFIER NAME AND OPTIONALLY THE DESCRIPTION OF THE IDENTIFIER.

3869 [R 183] ALL UN/CEFACT XML MUST BE INSTANTIATED USING UTF . UTF-8 SHOULD BE USED
3870 AS THE PREFERRED ENCODING. IF UTF-8 IS NOT USED, UTF-16 MUST BE USED.

3871 [R 185] THE `xsi:type` ATTRIBUTE MUST NOT BE USED.
3872

3873

Appendix D. Definition of Terms

3874 **Aggregate Business Information Entity (ABIE)** – A collection of related pieces of business information
3875 that together convey a distinct business meaning in a specific *Business Context*. Expressed in modelling
3876 terms, it is the representation of an *Object Class*, in a specific *Business Context*.

3877 **Aggregate Core Component - (ACC)** – A collection of related pieces of business information that
3878 together convey a distinct business meaning, independent of any specific *Business Context*. Expressed in
3879 modelling terms, it is the representation of an *Object Class*, independent of any specific *Business*
3880 *Context*.

3881 **Assembly Rules** - *Assembly Rules* group sets of unrefined *Business Information Entities* into larger
3882 structures. *Assembly Rules* are more fully defined and explained in the *Assembly Rules Supplemental*
3883 *Document*.

3884 **Association Business Information Entity (ASBIE)** - A Business Information Entity that represents a
3885 complex business characteristic of a specific Object Class in a specific Business Context. It has a unique
3886 Business Semantic definition. An Association Business Information Entity represents an Association
3887 Business Information Entity Property and is therefore associated to an Aggregate Business Information
3888 Entity, which describes its structure. An Association Business Information Entity is derived from an
3889 Association Core Component.

3890 **Association Business Information Entity Property** - A Business Information Entity Property for which
3891 the permissible values are expressed as a complex structure, represented by an Aggregate Business
3892 Information Entity.

3893 **Association Core Component (ASCC)** - A *Core Component* which constitutes a complex business
3894 characteristic of a specific *Aggregate Core Component* that represents an *Object Class*. It has a unique
3895 *Business Semantic* definition. An *Association Core Component* represents an *Association Core*
3896 *Component Property* and is associated to an *Aggregate Core Component*, which describes its structure.

3897 **Association Core Component Property** – A *Core Component Property* for which the permissible values
3898 are expressed as a complex structure, represented by an *Aggregate Core Component*.

3899 **Attribute** – A named value or relationship that exists for some or all instances of some entity and is
3900 directly associated with that instance.

3901 **Basic Business Information Entity (BBIE)** – A Business Information Entity that represents a singular
3902 business characteristic of a specific Object Class in a specific Business Context. It has a unique Business
3903 Semantic definition. A Basic Business Information Entity represents a Basic Business Information Entity
3904 Property and is therefore linked to a Data Type, which describes its values. A Basic Business Information
3905 Entity is derived from a Basic Core Component.

3906 **Basic Business Information Entity Property** – A Business Information Entity Property for which the
3907 permissible values are expressed by simple values, represented by a Data Type.

3908 **Basic Core Component (BCC)** – A *Core Component* which constitutes a singular business characteristic
3909 of a specific *Aggregate Core Component* that represents a *Object Class*. It has a unique *Business*
3910 *Semantic* definition. A *Basic Core Component* represents a *Basic Core Component Property* and is
3911 therefore of a *Data Type*, which defines its set of values. *Basic Core Components* function as the
3912 properties of *Aggregate Core Components*.

3913 **Basic Core Component (CC) Property** – A *Core Component Property* for which the permissible values
3914 are expressed by simple values, represented by a *Data Type*.

3915 **Business Context** – The formal description of a specific business circumstance as identified by the
3916 values of a set of *Context Categories*, allowing different business circumstances to be uniquely
3917 distinguished.

3918 **Business Information Entity (BIE)** – A piece of business data or a group of pieces of business data with
3919 a unique Business Semantic definition. A Business Information Entity can be a Basic Business
3920 Information Entity (BBIE), an Association Business Information Entity (ASBIE), or an Aggregate Business
3921 Information Entity (ABIE).

3922 **Business Information Entity (BIE) Property** – A business characteristic belonging to the Object Class
3923 in its specific Business Context that is represented by an Aggregate Business Information Entity.

3924 **Business Libraries** – A collection of approved process models specific to a line of business (e.g.,
3925 shipping, insurance).

3926 **Business Process** – The Business Process as described using the UN/CEFACT Catalogue of Common
3927 Business Processes.

3928 **Business Process Context** – The Business Process name(s) as described using the UN/CEFACT
3929 Catalogue of Common Business Processes as extended by the user.

3930 **Business Process Role Context** – The actors conducting a particular Business Process, as identified in
3931 the UN/CEFACT Catalogue of Common Business Processes.

3932 **Business Semantic(s)** – A precise meaning of words from a business perspective.

3933 **Business Term** – This is a synonym under which the *Core Component* or *Business Information Entity* is
3934 commonly known and used in the business. A *Core Component* or *Business Information Entity* may have
3935 several *Business Terms* or synonyms.

3936 **Cardinality** – An indication whether a characteristic is optional, mandatory and/or repetitive.

3937 **Catalogue of Business Information Entities** – This represents the approved set of *Business*
3938 *Information Entities* from which to choose when applying the *Core Component* discovery process

3939 **Catalogue of Core Components** – see Core Component Catalogue.

3940 **CCL** – see Core Component Library.

3941 **Child Core Component** – A *Core Component* used as part of a larger aggregate construct.

3942 **Classification Scheme** – This is an officially supported scheme to describe a given *Context Category*.

3943 **Constraint Language** – A formal expression of actions occurring in specific *Contexts* to assemble,
3944 structurally refine, and semantically qualify *Core Components*. The result of applying the *Constraint*
3945 *Language* to a set of *Core Components* in a specific *Context* is a set of *Business Information Entities*.

3946 **Content Component** – Defines the *Primitive Type* used to express the content of a *Core Component*
3947 *Type*.

3948 **Content Component Restrictions** – The formal definition of a format restriction that applies to the
3949 possible values of a *Content Component*.

3950 **Context** – Defines the circumstances in which a *Business Process* may be used. This is specified by a
3951 set of *Context Categories* known as *Business Context*.

3952 **Context Category** – A group of one or more related values used to express a characteristic of a business
3953 circumstance.

3954 **Context Rules Construct** – The overall expression of a single set of rules used to apply *Context* to *Core*
3955 *Components*.

3956 **Controlled Vocabulary** – A supplemental vocabulary used to uniquely define potentially ambiguous
3957 words or *Business Terms*. This ensures that every word within any of the *Core Component* names and
3958 definitions is used consistently, unambiguously and accurately.

3959 **Core Component (CC)** – A building block for the creation of a semantically correct and meaningful
3960 information exchange package. It contains only the information pieces necessary to describe a specific
3961 concept.

3962 **Core Component Catalogue** – The temporary collection of all metadata about each *Core Component*
3963 discovered during the development and initial testing of this *Core Component Technical Specification*,
3964 pending the establishment of a permanent Registry/repository.

3965 **Core Component Dictionary** – An extract from the *Core Component Catalogue* that provides a ready
3966 reference of the *Core Component* through its *Dictionary Entry Name*, component parts, and definition.

3967 **Core Component Library** – The Core Component Library is the part of the registry/repository in which
3968 Core Components shall be stored as Registry Classes. The Core Component Library will contain all the
3969 Core Component Types, Basic Core Components, Aggregate Core Components, Basic Business
3970 Information Entities and Aggregate Business Information Entities.

3971 **Core Component Property** – A business characteristic belonging to the *Object Class* represented by an
3972 *Aggregate Core Component*.

3973 **Core Component Type (CCT)** – A *Core Component*, which consists of one and only one *Content*
3974 *Component*, that carries the actual content plus one or more *Supplementary Components* giving an
3975 essential extra definition to the *Content Component*. *Core Component Types* do not have *Business*
3976 *Semantics*.

3977 **Data Type** – Defines the set of valid values that can be used for a particular *Basic Core Component*
3978 *Property* or *Basic Business Information Entity Property*. It is defined by specifying restrictions on the *Core*
3979 *Component Type* that forms the basis of the *Data Type*.

3980 **Definition** – This is the unique semantic meaning of a Core Component, Business Information Entity,
3981 Business Context or Data Type.

3982 **Dictionary Entry Name** – This is the unique official name of a Core Component, Business Information
3983 Entity, Business Context or Data Type in the dictionary.

3984 **Geopolitical Context** – Geographic factors that influence *Business Semantics* (e.g., the structure of an
3985 address).

3986 **Industry Classification Context** – Semantic influences related to the industry or industries of the trading
3987 partners (e.g., product identification schemes used in different industries).

3988 **Information Entity** – A reusable semantic building block for the exchange of business-related
3989 information.

3990 **Lower-Camel-Case (LCC)** – a style that capitalizes the first character of each word except the first word
3991 and compounds the name.

3992 **Naming Convention** – The set of rules that together comprise how the *Dictionary entry Name* for *Core*
3993 *Components* (See Section 6.1.4.1.4) and *Business Information Entities* (See Section 6.1.4.2.4) are
3994 constructed.

3995 **Object Class** – The logical data grouping (in a logical data model) to which a data element belongs
3996 (ISO11179). The *Object Class* is the part of a *Core Component's Dictionary Entry Name* that represents
3997 an activity or object in a specific *Context*.

3998 **Object Class Term** – A component of the name of a *Core Component* or *Business Information Entity*
3999 which represents the *Object Class* to which it belongs.

4000 **Official Constraints Context** – Legal and governmental influences on semantics (e.g. hazardous
4001 materials information required by law when shipping goods).

4002 **Order** – In the *Constraint Language*, the *Property* on the *ContextRules Construct* that applies a sequence
4003 to the application of a set of rules. Two Rule constructs cannot have the same value for the *Property*
4004 *Order*.

4005 **Primitive Type** – Used for the representation of a value. Possible values are String, Decimal, Integer,
4006 Boolean, Date and Binary.

4007 **Product Classification Context** – Factors influencing semantics that are the result of the goods or
4008 services being exchanged, handled, or paid for, etc. (e.g. the buying of consulting services as opposed to
4009 materials)

4010 **Property** – A peculiarity common to all members of an *Object Class*.

4011 **Property Term** – A semantically meaningful name for the characteristic of the *Object Class* that is
4012 represented by the *Core Component Property*. It shall serve as basis for the *Dictionary Entry Name* of the
4013 *Basic* and *Association Core Components* that represents this *Core Component Property*.

4014 **Qualifier Term** – A word or group of words that help define and differentiate an item (e.g. a *Business*
4015 *Information Entity* or a *Data Type*) from its associated items (e.g. from a *Core Component*, a *Core*
4016 *Compont Type*, another *Business Information Entity* or another *Data Type*).

4017 **Registry Class** – The formal definition of all the information necessary to be recorded in the Registry
4018 about a *Core Component*, a *Business Information Entity*, a *Data Type* or a *Business Context*.

4019 **Representation Term** – The type of valid values for a Basic Core Component or Business Information
4020 Entity.

4021 **Supplementary Component** – Gives additional meaning to the Content Component in the Core
4022 Component Type.

4023 **Supplementary Component Restrictions** – The formal definition of a format restriction that applies to
4024 the possible values of a *Supplementary Component*.

4025 **Supporting Role Context** – Semantic influences related to non-partner roles (e.g., data required by a
4026 third-party shipper in an order response going from seller to buyer.)

4027 **Syntax Binding** – The process of expressing a *Business Information Entity* in a specific syntax.

4028 **System Capabilities Context** – This *Context category* exists to capture the limitations of systems (e.g.
4029 an existing back office can only support an address in a certain form).

4030 **UMM Information Entity** – A *UMM Information Entity* realizes structured business information that is
4031 exchanged by partner roles performing activities in a business transaction. Information entities include or
4032 reference other information entities through associations.”

4033 **Unique Identifier** – The identifier that references a *Registry Class* instance in a universally unique and
4034 unambiguous way.

4035 **Upper-Camel-Case (UCC)** – a style that capitalizes the first character of each word and compounds the
4036 name.

4037 **Usage Rules** – *Usage Rules* describe how and/or when to use the *Registry Class*.

4038 **User Community** – A *User Community* is a group of practitioners, with a publicised contact address, who
4039 may define *Context* profiles relevant to their area of business. Users within the community do not create,
4040 define or manage their individual *Context* needs but conform to the community's standard. Such a
4041 community should liase closely with other communities and with general standards-making bodies to
4042 avoid overlapping work. A community may be as small as two consenting organisations.

4043 **Version** – An indication of the evolution over time of an instance of a *Core Component*, *Data Type*,
4044 *Business Context*, or *Business Information Entity*.

4045 **XML schema** – A generic term used to identify the family of grammar based XML document structure
4046 validation languages to include the more formal W3C XML Schema Technical Specification, Document
4047 Type Definition, Schematron, Regular Language Description for XML (RELAX), and the OASIS RELAX
4048 NG.

4049

4050

Appendix E. Related Documents

4051 The following documents provided significant levels of influence in the development of this document:

- 4052 • UN/CEFACT Core Components Technical Specification, Part 8 of the ebXML Framework Version 2.01
- 4053 • ebXML Technical Architecture Specification v1.04
- 4054 • OASIS/ebXML Registry Information Model v2.0
- 4055 • ebXML Requirements Specification v1.06
- 4056 • Information Technology - Metadata registries: Framework for the Specification and Standardization of
- 4057 Data Elements, International Standardization Organization, ISO 11179-1
- 4058 • Information Technology - Metadata registries: Classification of Concepts for the Identification of
- 4059 Domains, International Standardization Organization,
- 4060 ISO 11179-2
- 4061 • Information Technology - Metadata registries: Registry Metamodel, International Standardization
- 4062 Organization, ISO 11179-3
- 4063 • Information Technology - Metadata registries: Rules and Guidelines for the Formulation of Data
- 4064 Definitions, International Standardization Organization, ISO 11179-4
- 4065 • Information Technology - Metadata registries: Naming and Identification Principles for Data Elements,
- 4066 International Standardization Organization, ISO 11179-5
- 4067 • Information Technology - Metadata registries: Framework for the Specification and Standardization of
- 4068 Data Elements, International Standardization Organization, ISO 11179-6

Appendix F. Core Component Schema Module

4069

4070

4071

4072

4073

4074

4075

4076

4077

4078

4079

4080

4081

4082

4083

4084

4085

4086

4087

4088

4089

4090

4091

4092

4093

4094

4095

4096

4097

4098

4099

4100

4101

4102

4103

4104

4105

4106

4107

4108

4109

4110

4111

4112

4113

4114

4115

4116

4117

4118

4119

4120

4121

4122

4123

4124

4125

4126

4127

4128

4129

4130

4131

4132

4133

4134

4135

4136

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- ===== -->
<!-- ===== CCTS Core Component Types Schema Module ===== -->
<!-- ===== -->
<!--
Module of      Core Component Types,
Agency:       UN/CEFACT
VersionID:    1.0
Last change:  02. August 2004

Copyright (C) UN/CEFACT (2004). All Rights Reserved.

This document and translations of it may be copied and furnished to others,
and derivative works that comment on or otherwise explain it or assist
in its implementation may be prepared, copied, published and distributed,
in whole or in part, without restriction of any kind, provided that the
above copyright notice and this paragraph are included on all such copies
and derivative works. However, this document itself may not be modified in
any way, such as by removing the copyright notice or references to
UN/CEFACT, except as needed for the purpose of developing UN/CEFACT
specifications, in which case the procedures for copyrights defined in the
UN/CEFACT Intellectual Property Rights document must be followed, or as
required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked
by UN/CEFACT or its successors or assigns.

This document and the information contained herein is provided on an "AS IS"
basis and UN/CEFACT DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING
BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL
NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR
FITNESS FOR A PARTICULAR PURPOSE.
-->
<xsd:schema targetNamespace="urn:un:unece:unefact:data:draft:CoreComponentTypesSchemaModule:1.0"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:cct="urn:un:unece:unefact:data:draft:CoreComponentTypesSchemaModule:1.0" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <!-- ===== -->
  <!-- ===== Type Definitions ===== -->
  <!-- ===== CCT: AmountType ===== -->
  <!-- ===== -->
  <xsd:complexType name="AmountType">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">
        <ccts:UniqueID>UNDT000001</ccts:UniqueID>
        <ccts:CategoryCode>CCT</ccts:CategoryCode>
        <ccts:DictionaryEntryName>Amount. Type</ccts:DictionaryEntryName>
        <ccts:VersionID>1.0</ccts:VersionID>
        <ccts:DefinitionText>A number of monetary units specified in a currency where the unit of the currency is
explicit or implied.</ccts:DefinitionText>
        <ccts:RepresentationTermName>Amount</ccts:RepresentationTermName>
      </xsd:documentation>
    </xsd:annotation>
    <xsd:simpleContent>
      <xsd:extension base="xsd:decimal">
        <xsd:attribute name="amountCurrencyID" type="xsd:token" use="optional">
          <xsd:annotation>
            <xsd:documentation xml:lang="en">
              <ccts:UniqueID>UNDT000001-SC2</ccts:UniqueID>
              <ccts:CategoryCode>SC</ccts:CategoryCode>
              <ccts:DictionaryEntryName>Amount. Currency. Identifier</ccts:DictionaryEntryName>
              <ccts:VersionID>1.0</ccts:VersionID>
              <ccts:DefinitionText>The currency of the amount.</ccts:DefinitionText>
              <ccts:CardinalityText>0..1</ccts:CardinalityText>
              <ccts:PropertyTermName>Currency</ccts:PropertyTermName>
            </xsd:documentation>
          </xsd:annotation>
        </xsd:attribute>
      </xsd:extension>
    </xsd:simpleContent>
  </xsd:complexType>

```

```

4137         <ccts:RepresentationTermName>Identifier</ccts:RepresentationTermName>
4138         <ccts:UsageRuleText>Use 3 alpha codes UNECE Rec 9.</ccts:UsageRuleText>
4139         <ccts:UsageRuleText>It is recommended that the currency code is provided in this SC rather
4140 than in a separately defined BBIE.</ccts:UsageRuleText>
4141         </xsd:documentation>
4142     </xsd:annotation>
4143 </xsd:attribute>
4144 <xsd:attribute name="amountCurrencyCodeListVersionID" type="xsd:token" use="optional">
4145     <xsd:annotation>
4146         <xsd:documentation xml:lang="en">
4147             <ccts:UniqueID>UNDT000001-SC3</ccts:UniqueID>
4148             <ccts:CategoryCode>SC</ccts:CategoryCode>
4149             <ccts:DictionaryEntryName>Amount Currency. Code List VersionID.
4150 Identifier</ccts:DictionaryEntryName>
4151             <ccts:VersionID>1.0</ccts:VersionID>
4152             <ccts:DefinitionText>The VersionID of the UN/ECE Rec9 code list.</ccts:DefinitionText>
4153             <ccts:CardinalityText>0..1</ccts:CardinalityText>
4154             <ccts:PropertyTermName>Code List VersionID</ccts:PropertyTermName>
4155             <ccts:RepresentationTermName>Identifier</ccts:RepresentationTermName>
4156             <ccts:UsageRuleText>If not used, latest version is assumed.</ccts:UsageRuleText>
4157         </xsd:documentation>
4158     </xsd:annotation>
4159 </xsd:attribute>
4160 </xsd:extension>
4161 </xsd:simpleContent>
4162 </xsd:complexType>
4163 <!-- ===== CCT: BinaryObjectType ===== -->
4164 <!-- ===== -->
4165 <xsd:complexType name="BinaryObjectType">
4166     <xsd:annotation>
4167         <xsd:documentation xml:lang="en">
4168             <ccts:UniqueID>UNDT000002</ccts:UniqueID>
4169             <ccts:CategoryCode>CCT</ccts:CategoryCode>
4170             <ccts:DictionaryEntryName>Binary Object. Type</ccts:DictionaryEntryName>
4171             <ccts:VersionID>1.0</ccts:VersionID>
4172             <ccts:DefinitionText>A set of finite-length sequences of binary octets.</ccts:DefinitionText>
4173             <ccts:RepresentationTermName>Binary Object</ccts:RepresentationTermName>
4174         </xsd:documentation>
4175     </xsd:annotation>
4176     <xsd:simpleContent>
4177         <xsd:extension base="xsd:base64Binary">
4178             <xsd:attribute name="binaryObjectFormatText" type="xsd:token" use="optional">
4179                 <xsd:annotation>
4180                     <xsd:documentation xml:lang="en">
4181                         <ccts:UniqueID>UNDT000002-SC2</ccts:UniqueID>
4182                         <ccts:CategoryCode>SC</ccts:CategoryCode>
4183                         <ccts:DictionaryEntryName>Binary Object. Format. Text</ccts:DictionaryEntryName>
4184                         <ccts:VersionID>1.0</ccts:VersionID>
4185                         <ccts:DefinitionText>The format of the binary content.</ccts:DefinitionText>
4186                         <ccts:CardinalityText>0..1</ccts:CardinalityText>
4187                         <ccts:PropertyTermName>Format</ccts:PropertyTermName>
4188                         <ccts:RepresentationTermName>Text</ccts:RepresentationTermName>
4189                     </xsd:documentation>
4190                 </xsd:annotation>
4191             </xsd:attribute>
4192             <xsd:attribute name="binaryObjectMimeCode" type="xsd:token" use="optional">
4193                 <xsd:annotation>
4194                     <xsd:documentation xml:lang="en">
4195                         <ccts:UniqueID>UNDT000002-SC3</ccts:UniqueID>
4196                         <ccts:CategoryCode>SC</ccts:CategoryCode>
4197                         <ccts:DictionaryEntryName>Binary Object. Mime. Code</ccts:DictionaryEntryName>
4198                         <ccts:VersionID>1.0</ccts:VersionID>
4199                         <ccts:DefinitionText>The mime type of the binary object.</ccts:DefinitionText>
4200                         <ccts:PropertyTermName>Mime</ccts:PropertyTermName>
4201                         <ccts:RepresentationTermName>Code</ccts:RepresentationTermName>
4202                     </xsd:documentation>
4203                 </xsd:annotation>
4204             </xsd:attribute>
4205             <xsd:attribute name="binaryObjectEncodingCode" type="xsd:token" use="optional">
4206                 <xsd:annotation>
4207                     <xsd:documentation xml:lang="en">

```

```

4208         <ccts:UniqueID>UNDT000002-SC4</ccts:UniqueID>
4209         <ccts:CategoryCode>SC</ccts:CategoryCode>
4210         <ccts:DictionaryEntryName>Binary Object. Encoding. Code</ccts:DictionaryEntryName>
4211         <ccts:VersionID>1.0</ccts:VersionID>
4212         <ccts:DefinitionText>Specifies the decoding algorithm of the binary object.</ccts:DefinitionText>
4213         <ccts:CardinalityText>0..1</ccts:CardinalityText>
4214         <ccts:PropertyTermName>Encoding</ccts:PropertyTermName>
4215         <ccts:RepresentationTermName>Code</ccts:RepresentationTermName>
4216         </xsd:documentation>
4217     </xsd:annotation>
4218 </xsd:attribute>
4219 <xsd:attribute name="binaryObjectCharacterSetCode" type="xsd:token" use="optional">
4220     <xsd:annotation>
4221         <xsd:documentation xml:lang="en">
4222             <ccts:UniqueID>UNDT000002-SC5</ccts:UniqueID>
4223             <ccts:CategoryCode>SC</ccts:CategoryCode>
4224             <ccts:DictionaryEntryName>Binary Object. Character Set. Code</ccts:DictionaryEntryName>
4225             <ccts:VersionID>1.0</ccts:VersionID>
4226             <ccts:DefinitionText>The character set of the binary object if the mime type is
4227 text.</ccts:DefinitionText>
4228             <ccts:CardinalityText>0..1</ccts:CardinalityText>
4229             <ccts:PropertyTermName>Character Set</ccts:PropertyTermName>
4230             <ccts:RepresentationTermName>Code</ccts:RepresentationTermName>
4231             </xsd:documentation>
4232         </xsd:annotation>
4233     </xsd:attribute>
4234 <xsd:attribute name="binaryObjectURI" type="xsd:anyURI" use="optional">
4235     <xsd:annotation>
4236         <xsd:documentation xml:lang="en">
4237             <ccts:UniqueID>UNDT000002-SC6</ccts:UniqueID>
4238             <ccts:CategoryCode>SC</ccts:CategoryCode>
4239             <ccts:DictionaryEntryName>Binary Object.Uniform Resource.
4240 Identifier</ccts:DictionaryEntryName>
4241             <ccts:VersionID>1.0</ccts:VersionID>
4242             <ccts:DefinitionText>The Uniform Resource Identifier that identifies where the binary object is
4243 located.i</ccts:DefinitionText>
4244             <ccts:CardinalityText>0..1</ccts:CardinalityText>
4245             <ccts:PropertyTermName>Uniform Resource</ccts:PropertyTermName>
4246             <ccts:RepresentationTermName>Identifier</ccts:RepresentationTermName>
4247             </xsd:documentation>
4248         </xsd:annotation>
4249     </xsd:attribute>
4250 <xsd:attribute name="binaryObjectFilenameText" type="xsd:token" use="optional">
4251     <xsd:annotation>
4252         <xsd:documentation xml:lang="en">
4253             <ccts:UniqueID>UNDT000002-SC7</ccts:UniqueID>
4254             <ccts:CategoryCode>SC</ccts:CategoryCode>
4255             <ccts:DictionaryEntryName>Binary Object. Filename.Text</ccts:DictionaryEntryName>
4256             <ccts:VersionID>1.0</ccts:VersionID>
4257             <ccts:DefinitionText>The filename of the binary object.</ccts:DefinitionText>
4258             <ccts:CardinalityText>0..1</ccts:CardinalityText>
4259             <ccts:PropertyTermName>Filename</ccts:PropertyTermName>
4260             <ccts:RepresentationTermName>Text</ccts:RepresentationTermName>
4261             </xsd:documentation>
4262         </xsd:annotation>
4263     </xsd:attribute>
4264 </xsd:extension>
4265 </xsd:simpleContent>
4266 </xsd:complexType>
4267 <!-- ===== CCT: CodeType ===== -->
4268 <!-- ===== -->
4269 <xsd:complexType name="CodeType">
4270     <xsd:annotation>
4271         <xsd:documentation xml:lang="en">
4272             <ccts:UniqueID>UNDT000007</ccts:UniqueID>
4273             <ccts:CategoryCode>CCT</ccts:CategoryCode>
4274             <ccts:DictionaryEntryName>Code. Type</ccts:DictionaryEntryName>
4275             <ccts:VersionID>1.0</ccts:VersionID>
4276             <ccts:DefinitionText>A character string (letters, figures, or symbols) that for brevity and/or language
4277 independence may be used to represent or replace a definitive value or text of an attribute together with relevant
4278 supplementary information.</ccts:DefinitionText>

```

```

4279         <ccts:RepresentationTermName>Code</ccts:RepresentationTermName>
4280     </xsd:documentation>
4281 </xsd:annotation>
4282 <xsd:simpleContent>
4283     <xsd:extension base="xsd:token">
4284         <xsd:attribute name="codeListID" type="xsd:token" use="optional">
4285             <xsd:annotation>
4286                 <xsd:documentation xml:lang="en">
4287                     <ccts:UniqueID>UNDT000007-SC2</ccts:UniqueID>
4288                     <ccts:CategoryCode>SC</ccts:CategoryCode>
4289                     <ccts:DictionaryEntryName>Code List. Identifier</ccts:DictionaryEntryName>
4290                     <ccts:VersionID>1.0</ccts:VersionID>
4291                     <ccts:DefinitionText>The identification of a list of codes.</ccts:DefinitionText>
4292                     <ccts:CardinalityText>0..1</ccts:CardinalityText>
4293                     <ccts:PropertyTermName>Identification</ccts:PropertyTermName>
4294                     <ccts:RepresentationTermName>Identifier</ccts:RepresentationTermName>
4295                     <ccts:UsageRuleText>Indicates the list of codes from which the code content is
4296 drawn.</ccts:UsageRuleText>
4297                 </xsd:documentation>
4298             </xsd:annotation>
4299         </xsd:attribute>
4300     <xsd:attribute name="codeListAgencyID" type="xsd:token" use="optional">
4301         <xsd:annotation>
4302             <xsd:documentation xml:lang="en">
4303                 <ccts:UniqueID>UNDT000007-SC3</ccts:UniqueID>
4304                 <ccts:CategoryCode>SC</ccts:CategoryCode>
4305                 <ccts:DictionaryEntryName>Code List. Agency. Identifier</ccts:DictionaryEntryName>
4306                 <ccts:VersionID>1.0</ccts:VersionID>
4307                 <ccts:DefinitionText>An agency that maintains one or more lists of codes.</ccts:DefinitionText>
4308                 <ccts:CardinalityText>0..1</ccts:CardinalityText>
4309                 <ccts:PropertyTermName>Agency</ccts:PropertyTermName>
4310                 <ccts:RepresentationTermName>Identifier</ccts:RepresentationTermName>
4311                 <ccts:UsageRuleText>Use UN EDIFACT 3055 code list.</ccts:UsageRuleText>
4312                 <ccts:UsageRuleText>It is recommended that the code list agency is defined either by this
4313 identifier or by the code list agency name supplementary component but not both.</ccts:UsageRuleText>
4314             </xsd:documentation>
4315         </xsd:annotation>
4316     </xsd:attribute>
4317     <xsd:attribute name="codeListAgencyNameText" type="xsd:token" use="optional">
4318         <xsd:annotation>
4319             <xsd:documentation xml:lang="en">
4320                 <ccts:UniqueID>UNDT000007-SC4</ccts:UniqueID>
4321                 <ccts:CategoryCode>SC</ccts:CategoryCode>
4322                 <ccts:DictionaryEntryName>Code List. Agency Name. Text</ccts:DictionaryEntryName>
4323                 <ccts:VersionID>1.0</ccts:VersionID>
4324                 <ccts:DefinitionText>The name of the agency that maintains the list of
4325 codes.</ccts:DefinitionText>
4326                 <ccts:CardinalityText>0..1</ccts:CardinalityText>
4327                 <ccts:PropertyTermName>Agency Name</ccts:PropertyTermName>
4328                 <ccts:RepresentationTermName>Text</ccts:RepresentationTermName>
4329             </xsd:documentation>
4330         </xsd:annotation>
4331     </xsd:attribute>
4332     <xsd:attribute name="codeListNameText" type="xsd:token" use="optional">
4333         <xsd:annotation>
4334             <xsd:documentation xml:lang="en">
4335                 <ccts:UniqueID>UNDT000007-SC5</ccts:UniqueID>
4336                 <ccts:CategoryCode>SC</ccts:CategoryCode>
4337                 <ccts:DictionaryEntryName>Code List. Name. Text</ccts:DictionaryEntryName>
4338                 <ccts:VersionID>1.0</ccts:VersionID>
4339                 <ccts:DefinitionText>The name of a list of codes.</ccts:DefinitionText>
4340                 <ccts:CardinalityText>0..1</ccts:CardinalityText>
4341                 <ccts:PropertyTermName>Name</ccts:PropertyTermName>
4342                 <ccts:RepresentationTermName>Text</ccts:RepresentationTermName>
4343             </xsd:documentation>
4344         </xsd:annotation>
4345     </xsd:attribute>
4346     <xsd:attribute name="codeListVersionID" type="xsd:token" use="optional">
4347         <xsd:annotation>
4348             <xsd:documentation xml:lang="en">
4349                 <ccts:UniqueID>UNDT000007-SC6</ccts:UniqueID>

```

4350 <ccts:CategoryCode>SC</ccts:CategoryCode>
4351 <ccts:DictionaryEntryName>Code List. VersionID. Identifier</ccts:DictionaryEntryName>
4352 <ccts:DefinitionText>The version of the list of codes.</ccts:DefinitionText>
4353 <ccts:CardinalityText>0..1</ccts:CardinalityText>
4354 <ccts:PropertyTermName>VersionID</ccts:PropertyTermName>
4355 <ccts:RepresentationTermName>Identifier</ccts:RepresentationTermName>
4356 <ccts:UsageRuleText>Identifies the version of the code list.</ccts:UsageRuleText>
4357 </xsd:documentation>
4358 </xsd:annotation>
4359 </xsd:attribute>
4360 <xsd:attribute name="codeNameText" type="xsd:token" use="optional">
4361 <xsd:annotation>
4362 <xsd:documentation xml:lang="en">
4363 <ccts:UniqueID>UNDT000007-SC7</ccts:UniqueID>
4364 <ccts:CategoryCode>SC</ccts:CategoryCode>
4365 <ccts:DictionaryEntryName>Code. Name. Text</ccts:DictionaryEntryName>
4366 <ccts:VersionID>1.0</ccts:VersionID>
4367 <ccts:DefinitionText>The textual equivalent of the code content
4368 component.</ccts:DefinitionText>
4369 <ccts:CardinalityText>0..1</ccts:CardinalityText>
4370 <ccts:PropertyTermName>Name</ccts:PropertyTermName>
4371 <ccts:RepresentationTermName>Text</ccts:RepresentationTermName>
4372 <ccts:UsageRuleText> It is recommended that the name of a code is provided in this SC rather
4373 than in a separately defined BBIE.</ccts:UsageRuleText>
4374 </xsd:documentation>
4375 </xsd:annotation>
4376 </xsd:attribute>
4377 <xsd:attribute name="codeLanguageID" type="xsd:language" use="optional">
4378 <xsd:annotation>
4379 <xsd:documentation xml:lang="en">
4380 <ccts:UniqueID>UNDT000007-SC8</ccts:UniqueID>
4381 <ccts:CategoryCode>SC</ccts:CategoryCode>
4382 <ccts:DictionaryEntryName>Code. Language. Identifier</ccts:DictionaryEntryName>
4383 <ccts:VersionID>1.0</ccts:VersionID>
4384 <ccts:DefinitionText>The identifier of the language used in the code name.</ccts:DefinitionText>
4385 <ccts:CardinalityText>0..1</ccts:CardinalityText>
4386 <ccts:PropertyTermName>Language</ccts:PropertyTermName>
4387 <ccts:RepresentationTermName>Identifier</ccts:RepresentationTermName>
4388 <ccts:UsageRuleText>Identifies the language in which the code name is
4389 presented.</ccts:UsageRuleText>
4390 </xsd:documentation>
4391 </xsd:annotation>
4392 </xsd:attribute>
4393 <xsd:attribute name="codeListUniformResourceID" type="xsd:anyURI" use="optional">
4394 <xsd:annotation>
4395 <xsd:documentation xml:lang="en">
4396 <ccts:UniqueID>UNDT000007-SC9</ccts:UniqueID>
4397 <ccts:CategoryCode>SC</ccts:CategoryCode>
4398 <ccts:DictionaryEntryName>Code List. Uniform Resource.
4399 Identifier</ccts:DictionaryEntryName>
4400 <ccts:VersionID>1.0</ccts:VersionID>
4401 <ccts:DefinitionText>The Uniform Resource Identifier that identifies where the code list is
4402 located.</ccts:DefinitionText>
4403 <ccts:CardinalityText>0..1</ccts:CardinalityText>
4404 <ccts:PropertyTermName>Uniform Resource</ccts:PropertyTermName>
4405 <ccts:RepresentationTermName>Identifier</ccts:RepresentationTermName>
4406 <ccts:UsageRuleText>May be used to identify the location of the code list if available across the
4407 world wide web</ccts:UsageRuleText>
4408 </xsd:documentation>
4409 </xsd:annotation>
4410 </xsd:attribute>
4411 <xsd:attribute name="codeListSchemeURI" type="xsd:anyURI" use="optional">
4412 <xsd:annotation>
4413 <xsd:documentation xml:lang="en">
4414 <ccts:UniqueID>UNDT000007-SC10</ccts:UniqueID>
4415 <ccts:CategoryCode>SC</ccts:CategoryCode>
4416 <ccts:DictionaryEntryName>Code List Scheme. Uniform Resource.
4417 Identifier</ccts:DictionaryEntryName>
4418 <ccts:VersionID>1.0</ccts:VersionID>
4419 <ccts:DefinitionText>The Uniform Resource identifier that identifies where the code list scheme
4420 is located.</ccts:DefinitionText>

```

4421         <ccts:CardinalityText>0..1</ccts:CardinalityText>
4422         <ccts:PropertyTermName>Uniform Resource</ccts:PropertyTermName>
4423         <ccts:RepresentationTermName>Identifier</ccts:RepresentationTermName>
4424         <ccts:UsageRuleText>May be used to identify the location of the schema of the code list if
4425 available across the world wide web</ccts:UsageRuleText>
4426     </xsd:documentation>
4427 </xsd:annotation>
4428 </xsd:attribute>
4429 </xsd:extension>
4430 </xsd:simpleContent>
4431 </xsd:complexType>
4432 <!-- ===== CCT: DateTimeType ===== -->
4433 <!-- =====>
4434 <xsd:complexType name="DateTimeType">
4435     <xsd:annotation>
4436         <xsd:documentation xml:lang="en">
4437             <ccts:UniqueID>UNDT000008</ccts:UniqueID>
4438             <ccts:CategoryCode>CCT</ccts:CategoryCode>
4439             <ccts:DictionaryEntryName>Date Time. Type</ccts:DictionaryEntryName>
4440             <ccts:VersionID>1.0</ccts:VersionID>
4441             <ccts:DefinitionText>A particular point in the progression of time together with the relevant supplementary
4442 information.</ccts:DefinitionText>
4443             <ccts:RepresentationTermName>Date Time</ccts:RepresentationTermName>
4444             <ccts:UsageRuleText>This is a string formatted to the date format text SC.</ccts:UsageRuleText>
4445         </xsd:documentation>
4446     </xsd:annotation>
4447     <xsd:simpleContent>
4448         <xsd:extension base="xsd:string">
4449             <xsd:attribute name="dateTimeFormatText" type="xsd:token" use="optional">
4450                 <xsd:annotation>
4451                     <xsd:documentation xml:lang="en">
4452                         <ccts:UniqueID>UNDT000008-SC1</ccts:UniqueID>
4453                         <ccts:CategoryCode>SC</ccts:CategoryCode>
4454                         <ccts:DictionaryEntryName>Date Time. Format. Text</ccts:DictionaryEntryName>
4455                         <ccts:VersionID>1.0</ccts:VersionID>
4456                         <ccts:DefinitionText>The format of the date time content</ccts:DefinitionText>
4457                         <ccts:CardinalityText>0..1</ccts:CardinalityText>
4458                         <ccts:PropertyTermName>Format</ccts:PropertyTermName>
4459                         <ccts:RepresentationTermName>Text</ccts:RepresentationTermName>
4460                         <ccts:UsageRuleText>Use one of the following 8601 format strings: CC; CCYY; CCYYMM;
4461 CCYYMMDD; CCYYMMDDhhmmssZhmm; CCYYDDWW; MM; DD; WW; MMDD; hhZhmm; hmmmZhmm;
4462 hhmssZhmm; hhmss,ss..Zhmm; mm; ss; mmss</ccts:UsageRuleText>
4463                         <ccts:UsageRuleText>If this SC is not used it is assumed that the format is
4464 CCYYMMDDhhmmZhmm unless the format is included in the syntax implementation of the Date Content
4465 Component.</ccts:UsageRuleText>
4466                     </xsd:documentation>
4467                 </xsd:annotation>
4468             </xsd:attribute>
4469         </xsd:extension>
4470     </xsd:simpleContent>
4471 </xsd:complexType>
4472 <!-- ===== CCT: IdentifierType ===== -->
4473 <!-- =====>
4474 <xsd:complexType name="IdentifierType">
4475     <xsd:annotation>
4476         <xsd:documentation xml:lang="en">
4477             <ccts:UniqueID>UNDT000011</ccts:UniqueID>
4478             <ccts:CategoryCode>CCT</ccts:CategoryCode>
4479             <ccts:DictionaryEntryName>Identifier. Type</ccts:DictionaryEntryName>
4480             <ccts:VersionID>1.0</ccts:VersionID>
4481             <ccts:DefinitionText>A character string to identify and distinguish uniquely, one instance of an object in an
4482 identification scheme from all other objects in the same scheme together with relevant supplementary
4483 information.</ccts:DefinitionText>
4484             <ccts:RepresentationTermName>Identifier</ccts:RepresentationTermName>
4485         </xsd:documentation>
4486     </xsd:annotation>
4487     <xsd:simpleContent>
4488         <xsd:extension base="xsd:token">
4489             <xsd:attribute name="identificationSchemeID" type="xsd:token" use="optional">
4490                 <xsd:annotation>
4491                     <xsd:documentation xml:lang="en">

```



```

4492      <ccts:UniqueID>UNDT000011-SC2</ccts:UniqueID>
4493      <ccts:CategoryCode>SC</ccts:CategoryCode>
4494      <ccts:DictionaryEntryName>Identification Scheme. Identifier</ccts:DictionaryEntryName>
4495      <ccts:VersionID>1.0</ccts:VersionID>
4496      <ccts:DefinitionText>The identification of the identification scheme.</ccts:DefinitionText>
4497      <ccts:CardinalityText>0..1</ccts:CardinalityText>
4498      <ccts:PropertyTermName>IdentificationScheme</ccts:PropertyTermName>
4499      <ccts:RepresentationTermName>Identifier</ccts:RepresentationTermName>
4500      <ccts:UsageRuleText>Indicates the identification scheme maintained by the
4501 agency.</ccts:UsageRuleText>
4502      <ccts:UsageRuleText>It is recommended that this is identified either by this identifier or by the
4503 identification schema name supplementary component not both.</ccts:UsageRuleText>
4504      </xsd:documentation>
4505      </xsd:annotation>
4506    </xsd:attribute>
4507    <xsd:attribute name="identificationSchemeNameText" type="xsd:token" use="optional">
4508      <xsd:annotation>
4509        <xsd:documentation xml:lang="en">
4510          <ccts:UniqueID>UNDT000011-SC3</ccts:UniqueID>
4511          <ccts:CategoryCode>SC</ccts:CategoryCode>
4512          <ccts:DictionaryEntryName>Identification Scheme. Name. Text</ccts:DictionaryEntryName>
4513          <ccts:VersionID>1.0</ccts:VersionID>
4514          <ccts:DefinitionText>The name of the identification scheme.</ccts:DefinitionText>
4515          <ccts:CardinalityText>0..1</ccts:CardinalityText>
4516          <ccts:PropertyTermName>Name</ccts:PropertyTermName>
4517          <ccts:RepresentationTermName>Text</ccts:RepresentationTermName>
4518        </xsd:documentation>
4519      </xsd:annotation>
4520    </xsd:attribute>
4521    <xsd:attribute name="identificationSchemeAgencyID" type="xsd:token" use="optional">
4522      <xsd:annotation>
4523        <xsd:documentation xml:lang="en">
4524          <ccts:UniqueID>UNDT000011-SC4</ccts:UniqueID>
4525          <ccts:CategoryCode>SC</ccts:CategoryCode>
4526          <ccts:DictionaryEntryName>Identification Scheme. Agency.
4527 Identifier</ccts:DictionaryEntryName>
4528          <ccts:VersionID>1.0</ccts:VersionID>
4529          <ccts:DefinitionText>The identification of the agency that maintains the identification
4530 scheme.</ccts:DefinitionText>
4531          <ccts:CardinalityText>0..1</ccts:CardinalityText>
4532          <ccts:PropertyTermName>SchemeAgency</ccts:PropertyTermName>
4533          <ccts:RepresentationTermName>Identifier</ccts:RepresentationTermName>
4534        </xsd:documentation>
4535      </xsd:annotation>
4536    </xsd:attribute>
4537    <xsd:attribute name="identificatonSchemeAgencyNameText" type="xsd:token" use="optional">
4538      <xsd:annotation>
4539        <xsd:documentation xml:lang="en">
4540          <ccts:UniqueID>UNDT000011-SC5</ccts:UniqueID>
4541          <ccts:CategoryCode>SC</ccts:CategoryCode>
4542          <ccts:DictionaryEntryName>Identification Scheme. Agency Name.
4543 Text</ccts:DictionaryEntryName>
4544          <ccts:VersionID>1.0</ccts:VersionID>
4545          <ccts:DefinitionText>The name of the agency that maintains the identification
4546 scheme.</ccts:DefinitionText>
4547          <ccts:CardinalityText>0..1</ccts:CardinalityText>
4548          <ccts:PropertyTermName>Agency Name</ccts:PropertyTermName>
4549          <ccts:RepresentationTermName>Text</ccts:RepresentationTermName>
4550        </xsd:documentation>
4551      </xsd:annotation>
4552    </xsd:attribute>
4553    <xsd:attribute name="identificationSchemeVersionID" type="xsd:token" use="optional">
4554      <xsd:annotation>
4555        <xsd:documentation xml:lang="en">
4556          <ccts:UniqueID>UNDT000011-SC6</ccts:UniqueID>
4557          <ccts:CategoryCode>SC</ccts:CategoryCode>
4558          <ccts:DictionaryEntryName>Identification Scheme. VersionID.
4559 Identifier</ccts:DictionaryEntryName>
4560          <ccts:VersionID>1.0</ccts:VersionID>
4561          <ccts:DefinitionText>The version of the identification scheme.</ccts:DefinitionText>
4562          <ccts:CardinalityText>0..1</ccts:CardinalityText>

```

```

4563         <ccts:PropertyTermName>VersionID</ccts:PropertyTermName>
4564         <ccts:RepresentationTermName>Identifier</ccts:RepresentationTermName>
4565         <ccts:UsageRuleText>Identifies the version of the identification scheme.</ccts:UsageRuleText>
4566     </xsd:documentation>
4567 </xsd:annotation>
4568 </xsd:attribute>
4569 <xsd:attribute name="identificationSchemeDataURI" type="xsd:anyURI" use="optional">
4570     <xsd:annotation>
4571         <xsd:documentation xml:lang="en">
4572             <ccts:UniqueID>UNDT000011-SC7</ccts:UniqueID>
4573             <ccts:CategoryCode>SC</ccts:CategoryCode>
4574             <ccts:DictionaryEntryName>Identification Scheme Data. Uniform Resource.
4575 Identifier</ccts:DictionaryEntryName>
4576             <ccts:VersionID>1.0</ccts:VersionID>
4577             <ccts:DefinitionText>The Uniform Resource Identifier that identifies where the identification
4578 scheme data is located.</ccts:DefinitionText>
4579             <ccts:CardinalityText>0..1</ccts:CardinalityText>
4580             <ccts:PropertyTermName>>Uniform Resource</ccts:PropertyTermName>
4581             <ccts:RepresentationTermName>Identifier</ccts:RepresentationTermName>
4582             <ccts:UsageRuleText>May be used to identify the location of the identification scheme if
4583 available across the world wide web.</ccts:UsageRuleText>
4584         </xsd:documentation>
4585     </xsd:annotation>
4586 </xsd:attribute>
4587 <xsd:attribute name="identificationSchemeURI" type="xsd:anyURI" use="optional">
4588     <xsd:annotation>
4589         <xsd:documentation xml:lang="en">
4590             <ccts:UniqueID>UNDT000011-SC8</ccts:UniqueID>
4591             <ccts:CategoryCode>SC</ccts:CategoryCode>
4592             <ccts:DictionaryEntryName>Identification Scheme. Uniform Resource.
4593 Identifier</ccts:DictionaryEntryName>
4594             <ccts:VersionID>1.0</ccts:VersionID>
4595             <ccts:DefinitionText>The Uniform Resource Identifier that identifies where the identification
4596 scheme is located.</ccts:DefinitionText>
4597             <ccts:CardinalityText>0..1</ccts:CardinalityText>
4598             <ccts:PropertyTermName>>Uniform Resource</ccts:PropertyTermName>
4599             <ccts:RepresentationTermName>Identifier</ccts:RepresentationTermName>
4600             <ccts:UsageRuleText>May be used to identify the location of the schema of the identification
4601 scheme if available across the world wide web.</ccts:UsageRuleText>
4602         </xsd:documentation>
4603     </xsd:annotation>
4604 </xsd:attribute>
4605 </xsd:extension>
4606 </xsd:simpleContent>
4607 </xsd:complexType>
4608 <!-- ===== CCT: IndicatorType ----- -->
4609 <!-- ===== -->
4610 <xsd:complexType name="IndicatorType">
4611     <xsd:annotation>
4612         <xsd:documentation xml:lang="en">
4613             <ccts:UniqueID>UNDT000012</ccts:UniqueID>
4614             <ccts:CategoryCode>CCT</ccts:CategoryCode>
4615             <ccts:DictionaryEntryName>Indicator. Type</ccts:DictionaryEntryName>
4616             <ccts:VersionID>1.0</ccts:VersionID>
4617             <ccts:DefinitionText>A list of two mutually exclusive Boolean values that express the only possible states of
4618 a Property.</ccts:DefinitionText>
4619             <ccts:RepresentationTermName>Indicator</ccts:RepresentationTermName>
4620             <ccts:UsageRuleText>The enumeration is defined by the indicator format text SC.</ccts:UsageRuleText>
4621         </xsd:documentation>
4622     </xsd:annotation>
4623 <xsd:simpleContent>
4624     <xsd:extension base="xsd:string">
4625         <xsd:attribute name="indicatorFormatText" type="xsd:token" use="optional">
4626     <xsd:annotation>
4627         <xsd:documentation xml:lang="en">
4628             <ccts:UniqueID>UNDT000012-SC2</ccts:UniqueID>
4629             <ccts:CategoryCode>SC</ccts:CategoryCode>
4630             <ccts:DictionaryEntryName>Indicator. Format. Text</ccts:DictionaryEntryName>
4631             <ccts:VersionID>1.0</ccts:VersionID>
4632             <ccts:DefinitionText>Whether the indicator is numeric, textual or binary.</ccts:DefinitionText>
4633             <ccts:CardinalityText/>

```

```

4634         <ccts:PropertyTermName>Format</ccts:PropertyTermName>
4635         <ccts:RepresentationTermName>Text</ccts:RepresentationTermName>
4636         <ccts:UsageRuleText>If this SC is not used it is assumed that the format is True,False unless
4637 the format is included in the syntax implementation of the Content Component.</ccts:UsageRuleText>
4638     </xsd:documentation>
4639 </xsd:annotation>
4640 </xsd:attribute>
4641 </xsd:extension>
4642 </xsd:simpleContent>
4643 </xsd:complexType>
4644 <!-- ===== CCT: MeasureType ===== -->
4645 <!-- ===== -->
4646 <xsd:complexType name="MeasureType">
4647     <xsd:annotation>
4648         <xsd:documentation xml:lang="en">
4649             <ccts:UniqueID>UNDT000013</ccts:UniqueID>
4650             <ccts:CategoryCode>CCT</ccts:CategoryCode>
4651             <ccts:DictionaryEntryName>Measure. Type</ccts:DictionaryEntryName>
4652             <ccts:VersionID>1.0</ccts:VersionID>
4653             <ccts:DefinitionText>A numeric value determined by measuring an object along with the specified unit of
4654 measure.</ccts:DefinitionText>
4655             <ccts:RepresentationTermName>Measure</ccts:RepresentationTermName>
4656         </xsd:documentation>
4657     </xsd:annotation>
4658     <xsd:simpleContent>
4659         <xsd:extension base="xsd:decimal">
4660             <xsd:attribute name="measureUnitCode" type="xsd:token" use="optional">
4661                 <xsd:annotation>
4662                     <xsd:documentation xml:lang="en">
4663                         <ccts:UniqueID>UNDT000013-SC2</ccts:UniqueID>
4664                         <ccts:CategoryCode>SC</ccts:CategoryCode>
4665                         <ccts:DictionaryEntryName>Measure. Unit. Code</ccts:DictionaryEntryName>
4666                         <ccts:VersionID>1.0</ccts:VersionID>
4667                         <ccts:DefinitionText>The type of unit of measure.</ccts:DefinitionText>
4668                         <ccts:CardinalityText>0..1</ccts:CardinalityText>
4669                         <ccts:PropertyTermName>MeasureUnit</ccts:PropertyTermName>
4670                         <ccts:RepresentationTermName>Code</ccts:RepresentationTermName>
4671                         <ccts:UsageRuleText>Use the Common Codes of the UNECE Rec 20.</ccts:UsageRuleText>
4672                         <ccts:UsageRuleText>It is recommended that the measure unit code is provided in this SC
4673 rather than in a separately defined BBIE.</ccts:UsageRuleText>
4674                     </xsd:documentation>
4675                 </xsd:annotation>
4676             </xsd:attribute>
4677             <xsd:attribute name="measureUnitCodeListVersionID" type="xsd:token" use="optional">
4678                 <xsd:annotation>
4679                     <xsd:documentation xml:lang="en">
4680                         <ccts:UniqueID>UNDT000013-SC3</ccts:UniqueID>
4681                         <ccts:CategoryCode>SC</ccts:CategoryCode>
4682                         <ccts:DictionaryEntryName>Measure Unit. Code List VersionID.
4683 Identifier</ccts:DictionaryEntryName>
4684                         <ccts:VersionID>1.0</ccts:VersionID>
4685                         <ccts:DefinitionText>The version of the measure unit code list.</ccts:DefinitionText>
4686                         <ccts:CardinalityText>0..1</ccts:CardinalityText>
4687                         <ccts:PropertyTermName>Code List VersionID</ccts:PropertyTermName>
4688                         <ccts:RepresentationTermName>Identifier</ccts:RepresentationTermName>
4689                     </xsd:documentation>
4690                 </xsd:annotation>
4691             </xsd:attribute>
4692         </xsd:extension>
4693     </xsd:simpleContent>
4694 </xsd:complexType>
4695 <!-- ===== CCT: NumericType ===== -->
4696 <!-- ===== -->
4697 <xsd:complexType name="NumericType">
4698     <xsd:annotation>
4699         <xsd:documentation xml:lang="en">
4700             <ccts:UniqueID>UNDT000014</ccts:UniqueID>
4701             <ccts:CategoryCode>CCT</ccts:CategoryCode>
4702             <ccts:DictionaryEntryName>Numeric. Type</ccts:DictionaryEntryName>
4703             <ccts:VersionID>1.0</ccts:VersionID>

```

```

4704         <ccts:DefinitionText>Numeric information that is assigned or is determined by calculation, counting, or
4705 sequencing. It does not require a unit of quantity or unit of measure.</ccts:DefinitionText>
4706         <ccts:RepresentationTermName>Numeric</ccts:RepresentationTermName>
4707     </xsd:documentation>
4708 </xsd:annotation>
4709 <xsd:simpleContent>
4710     <xsd:extension base="xsd:decimal">
4711         <xsd:attribute name="numericFormatText" type="xsd:token" use="optional">
4712             <xsd:annotation>
4713                 <xsd:documentation xml:lang="en">
4714                     <ccts:UniqueID>UNDT000014-SC2</ccts:UniqueID>
4715                     <ccts:CategoryCode>SC</ccts:CategoryCode>
4716                     <ccts:DictionaryEntryName>Numeric. Format. Text</ccts:DictionaryEntryName>
4717                     <ccts:VersionID>1.0</ccts:VersionID>
4718                     <ccts:DefinitionText>Whether the number is an integer, decimal, real number or
4719 percentage.</ccts:DefinitionText>
4720                     <ccts:CardinalityText>0..1</ccts:CardinalityText>
4721                     <ccts:PropertyTermName>Format</ccts:PropertyTermName>
4722                     <ccts:RepresentationTermName>Text</ccts:RepresentationTermName>
4723                     <ccts:UsageRuleText>Not to be used as facets can be used instead.</ccts:UsageRuleText>
4724                 </xsd:documentation>
4725             </xsd:annotation>
4726         </xsd:attribute>
4727     </xsd:extension>
4728 </xsd:simpleContent>
4729 </xsd:complexType>
4730 <!-- ===== CCT: QuantityType ===== -->
4731 <!-- ===== -->
4732 <xsd:complexType name="QuantityType">
4733     <xsd:annotation>
4734         <xsd:documentation xml:lang="en">
4735             <ccts:UniqueID>UNDT000018</ccts:UniqueID>
4736             <ccts:CategoryCode>CCT</ccts:CategoryCode>
4737             <ccts:DictionaryEntryName>Quantity. Type</ccts:DictionaryEntryName>
4738             <ccts:VersionID>1.0</ccts:VersionID>
4739             <ccts:DefinitionText>A counted number of non-monetary units possibly including
4740 fractions.</ccts:DefinitionText>
4741             <ccts:RepresentationTermName>Quantity</ccts:RepresentationTermName>
4742         </xsd:documentation>
4743     </xsd:annotation>
4744     <xsd:simpleContent>
4745         <xsd:extension base="xsd:decimal">
4746             <xsd:attribute name="quantityUnitCode" type="xsd:token" use="optional">
4747                 <xsd:annotation>
4748                     <xsd:documentation xml:lang="en">
4749                         <ccts:UniqueID>UNDT000018-SC2</ccts:UniqueID>
4750                         <ccts:CategoryCode>SC</ccts:CategoryCode>
4751                         <ccts:DictionaryEntryName>Quantity. Unit. Code</ccts:DictionaryEntryName>
4752                         <ccts:VersionID>1.0</ccts:VersionID>
4753                         <ccts:DefinitionText>The unit of the quantity</ccts:DefinitionText>
4754                         <ccts:CardinalityText>0..1</ccts:CardinalityText>
4755                         <ccts:PropertyTermName>Unit</ccts:PropertyTermName>
4756                         <ccts:RepresentationTermName>Code</ccts:RepresentationTermName>
4757                         <ccts:UsageRuleText>If used, use the Common Codes of the UNECE Rec
4758 20</ccts:UsageRuleText>
4759                         <ccts:UsageRuleText>It is recommended that if a quantity unit code is provided it should be
4760 provided in this SC rather than in a separately defined BBIE.</ccts:UsageRuleText>
4761                         <ccts:UsageRuleText>If no code list is specified it is assumed to be UNECE Rec 20 (Common
4762 Codes).</ccts:UsageRuleText>
4763                     </xsd:documentation>
4764                 </xsd:annotation>
4765             </xsd:attribute>
4766             <xsd:attribute name="quantityUnitCodeListID" type="xsd:token" use="optional">
4767                 <xsd:annotation>
4768                     <xsd:documentation xml:lang="en">
4769                         <ccts:UniqueID>UNDT000018-SC3</ccts:UniqueID>
4770                         <ccts:CategoryCode>SC</ccts:CategoryCode>
4771                         <ccts:DictionaryEntryName>Quantity Unit. Code List. Identifier</ccts:DictionaryEntryName>
4772                         <ccts:VersionID>1.0</ccts:VersionID>
4773                         <ccts:DefinitionText>The quantity unit code list.</ccts:DefinitionText>
4774                         <ccts:CardinalityText>0..1</ccts:CardinalityText>

```

```

4775         <cts:PropertyTermName>Code List</cts:PropertyTermName>
4776         <cts:RepresentationTermName>Identifier</cts:RepresentationTermName>
4777     </xsd:documentation>
4778     </xsd:annotation>
4779 </xsd:attribute>
4780 <xsd:attribute name="quantityUnitCodeListAgencyID" type="xsd:token" use="optional">
4781     <xsd:annotation>
4782         <xsd:documentation xml:lang="en">
4783             <cts:UniqueID>UNDT000018-SC4</cts:UniqueID>
4784             <cts:CategoryCode>SC</cts:CategoryCode>
4785             <cts:DictionaryEntryName>Quantity Unit. Code List Agency.
4786 Identifier</cts:DictionaryEntryName>
4787             <cts:VersionID>1.0</cts:VersionID>
4788             <cts:DefinitionText>The identification of the agency that maintains the quantity unit code
4789 list</cts:DefinitionText>
4790             <cts:CardinalityText>0..1</cts:CardinalityText>
4791             <cts:PropertyTermName>Code List Agency</cts:PropertyTermName>
4792             <cts:RepresentationTermName>Identifier</cts:RepresentationTermName>
4793             <cts:UsageRuleText>Indicates the code list maintained by the agency which it is recommended
4794 is defined either by identifier or name.</cts:UsageRuleText>
4795         </xsd:documentation>
4796     </xsd:annotation>
4797 </xsd:attribute>
4798 <xsd:attribute name="quantityUnitCodeListAgencyNameText" type="xsd:token" use="optional">
4799     <xsd:annotation>
4800         <xsd:documentation xml:lang="en">
4801             <cts:UniqueID>UNDT000018-SC5</cts:UniqueID>
4802             <cts:CategoryCode>SC</cts:CategoryCode>
4803             <cts:DictionaryEntryName>Quantity Unit. Code List Agency Name.
4804 Text</cts:DictionaryEntryName>
4805             <cts:VersionID>1.0</cts:VersionID>
4806             <cts:DefinitionText>The name of the agency which maintains the quantity unit code
4807 list.</cts:DefinitionText>
4808             <cts:CardinalityText>0..1</cts:CardinalityText>
4809             <cts:PropertyTermName>Code List Agency Name</cts:PropertyTermName>
4810             <cts:RepresentationTermName>Text</cts:RepresentationTermName>
4811         </xsd:documentation>
4812     </xsd:annotation>
4813 </xsd:attribute>
4814 </xsd:extension>
4815 </xsd:simpleContent>
4816 </xsd:complexType>
4817 <!-- ===== CCT: TextType ===== -->
4818 <!-- ===== -->
4819 <xsd:complexType name="TextType">
4820     <xsd:annotation>
4821         <xsd:documentation xml:lang="en">
4822             <cts:UniqueID>UNDT000019</cts:UniqueID>
4823             <cts:CategoryCode>CCT</cts:CategoryCode>
4824             <cts:DictionaryEntryName>Text. Type</cts:DictionaryEntryName>
4825             <cts:VersionID>1.0</cts:VersionID>
4826             <cts:DefinitionText>A character string (i.e. a finite set of characters) generally in the form of words of a
4827 language.</cts:DefinitionText>
4828             <cts:RepresentationTermName>Text</cts:RepresentationTermName>
4829         </xsd:documentation>
4830     </xsd:annotation>
4831     <xsd:simpleContent>
4832         <xsd:extension base="xsd:string">
4833             <xsd:attribute name="languageID" type="xsd:language" use="optional">
4834                 <xsd:annotation>
4835                     <xsd:documentation xml:lang="en">
4836                         <cts:UniqueID>UNDT000019-SC2</cts:UniqueID>
4837                         <cts:CategoryCode>SC</cts:CategoryCode>
4838                         <cts:DictionaryEntryName>Language. Identifier</cts:DictionaryEntryName>
4839                         <cts:VersionID>1.0</cts:VersionID>
4840                         <cts:DefinitionText>The identifier of the language used in the content
4841 component.</cts:DefinitionText>
4842                         <cts:CardinalityText>0..1</cts:CardinalityText>
4843                         <cts:PropertyTermName>Language</cts:PropertyTermName>
4844                         <cts:RepresentationTermName>Identifier</cts:RepresentationTermName>

```

```

4845         <ccts:UsageRuleText>Identifies the language in which the code name is
4846 presented.</ccts:UsageRuleText>
4847     </xsd:documentation>
4848 </xsd:annotation>
4849 </xsd:attribute>
4850 <xsd:attribute name="languageLocaleID" type="xsd:token" use="optional">
4851 <xsd:annotation>
4852 <xsd:documentation xml:lang="en">
4853 <ccts:UniqueID>UNDT000019-SC3</ccts:UniqueID>
4854 <ccts:CategoryCode>SC</ccts:CategoryCode>
4855 <ccts:DictionaryEntryName> Language. Locale. Identifier</ccts:DictionaryEntryName>
4856 <ccts:VersionID>1.0</ccts:VersionID>
4857 <ccts:DefinitionText>The identification of the locale of the language.</ccts:DefinitionText>
4858 <ccts:CardinalityText>0..1</ccts:CardinalityText>
4859 <ccts:PropertyTermName>Locale</ccts:PropertyTermName>
4860 <ccts:RepresentationTermName>Identifier</ccts:RepresentationTermName>
4861 <ccts:UsageRuleText>Not recommended for use.</ccts:UsageRuleText>
4862 </xsd:documentation>
4863 </xsd:annotation>
4864 </xsd:attribute>
4865 </xsd:extension>
4866 </xsd:simpleContent>
4867 </xsd:complexType>
4868 </xsd:schema>
4869
4870

```

4871

Appendix G. Unqualified Data Type Schema Module

4872

4873

4874

4875

4876

4877

4878

4879

4880

4881

4882

4883

4884

4885

4886

4887

4888

4889

4890

4891

4892

4893

4894

4895

4896

4897

4898

4899

4900

4901

4902

4903

4904

4905

4906

4907

4908

4909

4910

4911

4912

4913

4914

4915

4916

4917

4918

4919

4920

4921

4922

4923

4924

4925

4926

4927

4928

4929

4930

4931

4932

4933

4934

4935

4936

4937

4938

```

<?xml version="1.0" encoding="UTF-8"?>
<!-- =====>
<!-- ===== UDT Unqualified Data Types Schema Module ===== -->
<!-- =====>
<!--
Module of      Unqualified Data Types,
Agency:       UN/CEFACT,
Version:      1.0
Last change:  02. August 2004

Copyright (C) UN/CEFACT (2004). All Rights Reserved.

This document and translations of it may be copied and furnished to others,
and derivative works that comment on or otherwise explain it or assist
in its implementation may be prepared, copied, published and distributed,
in whole or in part, without restriction of any kind, provided that the
above copyright notice and this paragraph are included on all such copies
and derivative works. However, this document itself may not be modified in
any way, such as by removing the copyright notice or references to
UN/CEFACT, except as needed for the purpose of developing UN/CEFACT
specifications, in which case the procedures for copyrights defined in the
UN/CEFACT Intellectual Property Rights document must be followed, or as
required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked
by UN/CEFACT or its successors or assigns.

This document and the information contained herein is provided on an "AS IS"
basis and UN/CEFACT DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING
BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL
NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR
FITNESS FOR A PARTICULAR PURPOSE.
-->
<xsd:schema targetNamespace="urn:un:unece:unefact:data:draft:UnqualifiedDataTypesSchemaModule:1:0"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:clm5639="urn:un:unece:unefact:codelist:draft:ISO_Language_Code:1988"
xmlns:clm54217="urn:un:unece:unefact:codelist:draft:ISO_Currency_Codes:2001"
xmlns:clmIANAMIMEMediaTypes="urn:un:unece:unefact:codelist:draft:IANA_MIME_Media_Types:2003"
xmlns:clm66411="urn:un:unece:unefact:codelist:draft:UNECE_Codes_for_Units_of_Measure:2001"
xmlns:udt="urn:un:unece:unefact:data:draft:UnqualifiedDataTypesSchemaModule:1:0" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <!-- ===== Imports ===== -->
  <!-- ===== Imports of Code Lists ===== -->
  <!-- =====>
  <xsd:import namespace="urn:un:unece:unefact:codelist:draft:UNECE_Codes_for_Units_of_Measure:2001"
schemaLocation="CodeList_UnitCode_UNECE_7_04.xsd"/>
  <xsd:import namespace="urn:un:unece:unefact:codelist:draft:IANA_MIME_Media_Types:2003"
schemaLocation="CodeList_MIMEMediaTypeCode_IANA_7_04.xsd"/>
  <xsd:import namespace="urn:un:unece:unefact:codelist:draft:ISO_Currency_Codes:2001"
schemaLocation="CodeList_CurrencyCode_ISO_7_04.xsd"/>
  <xsd:import namespace="urn:un:unece:unefact:codelist:draft:ISO_Language_Code:1988"
schemaLocation="CodeList_LanguageCode_ISO_7_04.xsd"/>
  <!-- ===== Type Definitions ===== -->
  <!-- ===== Primary RT: Amount. Type ===== -->
  <!-- =====>
  <xsd:complexType name="AmountType">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">
        <ccts:UniqueID>UDT000001</ccts:UniqueID>
        <ccts:CategoryCode>UDT</ccts:CategoryCode>
        <ccts:DictionaryEntryName>Amount. Type</ccts:DictionaryEntryName>
        <ccts:VersionID>1.0</ccts:VersionID>
        <ccts:DefinitionText>A number of monetary units specified in a currency where the unit of the currency is
explicit or implied.</ccts:DefinitionText>

```

```

4939         <ccts:RepresentationTermName>Amount</ccts:RepresentationTermName>
4940     </xsd:documentation>
4941 </xsd:annotation>
4942 <xsd:simpleContent>
4943     <xsd:extension base="xsd:decimal">
4944         <xsd:attribute name="amountCurrencyID" type="clm54217:CurrencyCodeContentType" use="required">
4945             <xsd:annotation>
4946                 <xsd:documentation xml:lang="en">
4947                     <ccts:UniqueID>UDT000001-SC2</ccts:UniqueID>
4948                     <ccts:CategoryCode>SC</ccts:CategoryCode>
4949                     <ccts:DictionaryEntryName>Amount. Currency. Identifier</ccts:DictionaryEntryName>
4950                     <ccts:VersionID>1.0</ccts:VersionID>
4951                     <ccts:DefinitionText>The currency of the amount.</ccts:DefinitionText>
4952                     <ccts:CardinalityText>1</ccts:CardinalityText>
4953                     <ccts:PropertyTermName>Currency</ccts:PropertyTermName>
4954                     <ccts:RepresentationTermName>Identifier</ccts:RepresentationTermName>
4955                     <ccts:UsageRuleText>Use 3 alpha codes UNECE Rec 9.</ccts:UsageRuleText>
4956                 </xsd:documentation>
4957             </xsd:annotation>
4958         </xsd:attribute>
4959     </xsd:extension>
4960 </xsd:simpleContent>
4961 </xsd:complexType>
4962 <!-- ===== Primary RT: Binary Object. Type ===== -->
4963 <!-- ===== -->
4964 <xsd:complexType name="BinaryObjectType">
4965     <xsd:annotation>
4966         <xsd:documentation xml:lang="en">
4967             <ccts:UniqueID>UDT000002</ccts:UniqueID>
4968             <ccts:CategoryCode>UDT</ccts:CategoryCode>
4969             <ccts:DictionaryEntryName>Binary Object. Type</ccts:DictionaryEntryName>
4970             <ccts:VersionID>1.0</ccts:VersionID>
4971             <ccts:DefinitionText>A set of finite-length sequences of binary octets.</ccts:DefinitionText>
4972             <ccts:RepresentationTermName>Binary Object</ccts:RepresentationTermName>
4973         </xsd:documentation>
4974     </xsd:annotation>
4975     <xsd:simpleContent>
4976         <xsd:extension base="xsd:base64Binary">
4977             <xsd:attribute name="binaryObjectFormatText" type="xsd:token" use="optional">
4978                 <xsd:annotation>
4979                     <xsd:documentation xml:lang="en">
4980                         <ccts:UniqueID>UDT000002-SC2</ccts:UniqueID>
4981                         <ccts:CategoryCode>SC</ccts:CategoryCode>
4982                         <ccts:DictionaryEntryName>Binary Object. Format. Text</ccts:DictionaryEntryName>
4983                         <ccts:VersionID>1.0</ccts:VersionID>
4984                         <ccts:DefinitionText>The format of the binary content.</ccts:DefinitionText>
4985                         <ccts:CardinalityText>0..1</ccts:CardinalityText>
4986                         <ccts:PropertyTermName>Format</ccts:PropertyTermName>
4987                         <ccts:RepresentationTermName>Text</ccts:RepresentationTermName>
4988                     </xsd:documentation>
4989                 </xsd:annotation>
4990             </xsd:attribute>
4991             <xsd:attribute name="binaryObjectMimeCode"
4992 type="clmIANAMIMEMediaTypes:BinaryObjectMimeCodeContentType" use="required">
4993                 <xsd:annotation>
4994                     <xsd:documentation xml:lang="en">
4995                         <ccts:UniqueID>UDT000002-SC3</ccts:UniqueID>
4996                         <ccts:CategoryCode>SC</ccts:CategoryCode>
4997                         <ccts:DictionaryEntryName>Binary Object. Mime. Code</ccts:DictionaryEntryName>
4998                         <ccts:VersionID>1.0</ccts:VersionID>
4999                         <ccts:DefinitionText>The mime type of the binary object.</ccts:DefinitionText>
5000                         <ccts:CardinalityText>1</ccts:CardinalityText>
5001                         <ccts:PropertyTermName>Mime</ccts:PropertyTermName>
5002                         <ccts:RepresentationTermName>Code</ccts:RepresentationTermName>
5003                     </xsd:documentation>
5004                 </xsd:annotation>
5005             </xsd:attribute>
5006             <xsd:attribute name="binaryObjectEncodingCode" type="xsd:token" use="optional">
5007                 <xsd:annotation>
5008                     <xsd:documentation xml:lang="en">
5009                         <ccts:UniqueID>UDT000002-SC4</ccts:UniqueID>

```



```

5010         <ccts:CategoryCode>SC</ccts:CategoryCode>
5011         <ccts:DictionaryEntryName>Binary Object. Encoding. Code</ccts:DictionaryEntryName>
5012         <ccts:VersionID>1.0</ccts:VersionID>
5013         <ccts:DefinitionText>Specifies the decoding algorithm of the binary object.</ccts:DefinitionText>
5014         <ccts:CardinalityText>0..1</ccts:CardinalityText>
5015         <ccts:PropertyTermName>Encoding</ccts:PropertyTermName>
5016         <ccts:RepresentationTermName>Code</ccts:RepresentationTermName>
5017     </xsd:documentation>
5018 </xsd:annotation>
5019 </xsd:attribute>
5020 <xsd:attribute name="binaryObjectCharacterSetCode" type="xsd:token" use="optional">
5021     <xsd:annotation>
5022         <xsd:documentation xml:lang="en">
5023             <ccts:UniqueID>UDT000002-SC5</ccts:UniqueID>
5024             <ccts:CategoryCode>SC</ccts:CategoryCode>
5025             <ccts:DictionaryEntryName>Binary Object. Character Set. Code</ccts:DictionaryEntryName>
5026             <ccts:VersionID>1.0</ccts:VersionID>
5027             <ccts:DefinitionText>The character set of the binary object if the mime type is
5028 text.</ccts:DefinitionText>
5029             <ccts:CardinalityText>0..1</ccts:CardinalityText>
5030             <ccts:PropertyTermName>Character Set</ccts:PropertyTermName>
5031             <ccts:RepresentationTermName>Code</ccts:RepresentationTermName>
5032         </xsd:documentation>
5033     </xsd:annotation>
5034 </xsd:attribute>
5035 <xsd:attribute name="binaryObjectURI" type="xsd:anyURI" use="optional">
5036     <xsd:annotation>
5037         <xsd:documentation xml:lang="en">
5038             <ccts:UniqueID>UDT000002-SC6</ccts:UniqueID>
5039             <ccts:CategoryCode>SC</ccts:CategoryCode>
5040             <ccts:DictionaryEntryName>Binary Object. Uniform Resource.
5041 Identifier</ccts:DictionaryEntryName>
5042             <ccts:VersionID>1.0</ccts:VersionID>
5043             <ccts:DefinitionText>The Uniform Resource Identifier that identifies where the binary object is
5044 located.i</ccts:DefinitionText>
5045             <ccts:CardinalityText>0..1</ccts:CardinalityText>
5046             <ccts:PropertyTermName>Uniform Resource</ccts:PropertyTermName>
5047             <ccts:RepresentationTermName>Identifier</ccts:RepresentationTermName>
5048         </xsd:documentation>
5049     </xsd:annotation>
5050 </xsd:attribute>
5051 <xsd:attribute name="binaryObjectFilenameText" type="xsd:token" use="optional">
5052     <xsd:annotation>
5053         <xsd:documentation xml:lang="en">
5054             <ccts:UniqueID>UDT000002-SC7</ccts:UniqueID>
5055             <ccts:CategoryCode>SC</ccts:CategoryCode>
5056             <ccts:DictionaryEntryName>Binary Object. Filename.Text</ccts:DictionaryEntryName>
5057             <ccts:VersionID>1.0</ccts:VersionID>
5058             <ccts:DefinitionText>The filename of the binary object.</ccts:DefinitionText>
5059             <ccts:CardinalityText>0..1</ccts:CardinalityText>
5060             <ccts:PropertyTermName>Filename</ccts:PropertyTermName>
5061             <ccts:RepresentationTermName>Text</ccts:RepresentationTermName>
5062         </xsd:documentation>
5063     </xsd:annotation>
5064 </xsd:attribute>
5065 </xsd:extension>
5066 </xsd:simpleContent>
5067 </xsd:complexType>
5068 <!-- ===== Secondary RT: Graphic. Type          ===== -->
5069 <!-- =====>
5070 <xsd:complexType name="GraphicType">
5071     <xsd:annotation>
5072         <xsd:documentation xml:lang="en">
5073             <ccts:UniqueID>UDT000003</ccts:UniqueID>
5074             <ccts:CategoryCode>UDT</ccts:CategoryCode>
5075             <ccts:DictionaryEntryName>Graphic. Type</ccts:DictionaryEntryName>
5076             <ccts:VersionID>1.0</ccts:VersionID>
5077             <ccts:DefinitionText>A diagram, graph, mathematical curves, or similar representation.</ccts:DefinitionText>
5078             <ccts:RepresentationTermName>Graphic</ccts:RepresentationTermName>
5079         </xsd:documentation>
5080     </xsd:annotation>

```

```

5081     <xsd:simpleContent>
5082         <xsd:extension base="xsd:base64Binary">
5083             <xsd:attribute name="graphicFormatText" type="xsd:token" use="optional">
5084                 <xsd:annotation>
5085                     <xsd:documentation xml:lang="en">
5086                         <ccts:UniqueID>UDT000003-SC2</ccts:UniqueID>
5087                         <ccts:CategoryCode>SC</ccts:CategoryCode>
5088                         <ccts:DictionaryEntryName>Graphic. Format. Text</ccts:DictionaryEntryName>
5089                         <ccts:VersionID>1.0</ccts:VersionID>
5090                         <ccts:DefinitionText>The format of the binary content.</ccts:DefinitionText>
5091                         <ccts:CardinalityText>0..1</ccts:CardinalityText>
5092                         <ccts:PropertyTermName>Format</ccts:PropertyTermName>
5093                         <ccts:RepresentationTermName>Text</ccts:RepresentationTermName>
5094                     </xsd:documentation>
5095                 </xsd:annotation>
5096             </xsd:attribute>
5097             <xsd:attribute name="graphicMimeCode"
5098 type="clmIANAMIMEMediaTypes:BinaryObjectMimeCodeContentType" use="required">
5099                 <xsd:annotation>
5100                     <xsd:documentation xml:lang="en">
5101                         <ccts:UniqueID>UDT000003-SC3</ccts:UniqueID>
5102                         <ccts:CategoryCode>SC</ccts:CategoryCode>
5103                         <ccts:DictionaryEntryName>Graphic. Mime. Code</ccts:DictionaryEntryName>
5104                         <ccts:VersionID>1.0</ccts:VersionID>
5105                         <ccts:DefinitionText>The mime type of the binary object.</ccts:DefinitionText>
5106                         <ccts:CardinalityText>1</ccts:CardinalityText>
5107                         <ccts:PropertyTermName>Mime</ccts:PropertyTermName>
5108                         <ccts:RepresentationTermName>Code</ccts:RepresentationTermName>
5109                     </xsd:documentation>
5110                 </xsd:annotation>
5111             </xsd:attribute>
5112             <xsd:attribute name="graphicEncodingCode" type="xsd:token" use="optional">
5113                 <xsd:annotation>
5114                     <xsd:documentation xml:lang="en">
5115                         <ccts:UniqueID>UDT000003-SC4</ccts:UniqueID>
5116                         <ccts:CategoryCode>SC</ccts:CategoryCode>
5117                         <ccts:DictionaryEntryName>Graphic. Encoding. Code</ccts:DictionaryEntryName>
5118                         <ccts:VersionID>1.0</ccts:VersionID>
5119                         <ccts:DefinitionText>Specifies the decoding algorithm of the binary object.</ccts:DefinitionText>
5120                         <ccts:CardinalityText>0..1</ccts:CardinalityText>
5121                         <ccts:PropertyTermName>Encoding</ccts:PropertyTermName>
5122                         <ccts:RepresentationTermName>Code</ccts:RepresentationTermName>
5123                     </xsd:documentation>
5124                 </xsd:annotation>
5125             </xsd:attribute>
5126             <xsd:attribute name="graphicURI" type="xsd:anyURI" use="optional">
5127                 <xsd:annotation>
5128                     <xsd:documentation xml:lang="en">
5129                         <ccts:UniqueID>UDT000003-SC6</ccts:UniqueID>
5130                         <ccts:CategoryCode>SC</ccts:CategoryCode>
5131                         <ccts:DictionaryEntryName>Graphic. Uniform Resource. Identifier</ccts:DictionaryEntryName>
5132                         <ccts:VersionID>1.0</ccts:VersionID>
5133                         <ccts:DefinitionText>The Uniform Resource Identifier that identifies where the binary object is
5134 located.</ccts:DefinitionText>
5135                         <ccts:CardinalityText>0..1</ccts:CardinalityText>
5136                         <ccts:PropertyTermName>Uniform Resource</ccts:PropertyTermName>
5137                         <ccts:RepresentationTermName>Identifier</ccts:RepresentationTermName>
5138                     </xsd:documentation>
5139                 </xsd:annotation>
5140             </xsd:attribute>
5141             <xsd:attribute name="graphicFilenameText" type="xsd:token" use="optional">
5142                 <xsd:annotation>
5143                     <xsd:documentation xml:lang="en">
5144                         <ccts:UniqueID>UDT000003-SC7</ccts:UniqueID>
5145                         <ccts:CategoryCode>SC</ccts:CategoryCode>
5146                         <ccts:DictionaryEntryName>Graphic. Filename. Text</ccts:DictionaryEntryName>
5147                         <ccts:VersionID>1.0</ccts:VersionID>
5148                         <ccts:DefinitionText>The filename of the binary object.</ccts:DefinitionText>
5149                         <ccts:CardinalityText>0..1</ccts:CardinalityText>
5150                         <ccts:PropertyTermName>Filename</ccts:PropertyTermName>
5151                         <ccts:RepresentationTermName>Text</ccts:RepresentationTermName>

```

```

5152         </xsd:documentation>
5153     </xsd:annotation>
5154 </xsd:attribute>
5155 </xsd:extension>
5156 </xsd:simpleContent>
5157 </xsd:complexType>
5158 <!-- ===== Secondary RT: Picture. Type ===== -->
5159 <!-- =====>
5160 <xsd:complexType name="PictureType">
5161     <xsd:annotation>
5162         <xsd:documentation xml:lang="en">
5163             <ccts:UniqueID>UDT000004</ccts:UniqueID>
5164             <ccts:CategoryCode>UDT</ccts:CategoryCode>
5165             <ccts:DictionaryEntryName>Picture. Type</ccts:DictionaryEntryName>
5166             <ccts:VersionID>1.0</ccts:VersionID>
5167             <ccts:DefinitionText>A diagram, graph, mathematical curves, or similar representation.</ccts:DefinitionText>
5168             <ccts:RepresentationTermName>Picture</ccts:RepresentationTermName>
5169         </xsd:documentation>
5170     </xsd:annotation>
5171 <xsd:simpleContent>
5172     <xsd:extension base="xsd:base64Binary">
5173         <xsd:attribute name="pictureFormatText" type="xsd:token" use="optional">
5174             <xsd:annotation>
5175                 <xsd:documentation xml:lang="en">
5176                     <ccts:UniqueID>UDT000004-SC2</ccts:UniqueID>
5177                     <ccts:CategoryCode>SC</ccts:CategoryCode>
5178                     <ccts:DictionaryEntryName>Picture. Format. Text</ccts:DictionaryEntryName>
5179                     <ccts:VersionID>1.0</ccts:VersionID>
5180                     <ccts:DefinitionText>The format of the binary content.</ccts:DefinitionText>
5181                     <ccts:CardinalityText>0..1</ccts:CardinalityText>
5182                     <ccts:PropertyTermName>Format</ccts:PropertyTermName>
5183                     <ccts:RepresentationTermName>Text</ccts:RepresentationTermName>
5184                 </xsd:documentation>
5185             </xsd:annotation>
5186         </xsd:attribute>
5187         <xsd:attribute name="pictureMimeCode"
5188 type="clmIANAMIMEMediaTypes:BinaryObjectMimeCodeContentType" use="required">
5189             <xsd:annotation>
5190                 <xsd:documentation xml:lang="en">
5191                     <ccts:UniqueID>UDT000004-SC3</ccts:UniqueID>
5192                     <ccts:CategoryCode>SC</ccts:CategoryCode>
5193                     <ccts:DictionaryEntryName>Picture. Mime. Code</ccts:DictionaryEntryName>
5194                     <ccts:VersionID>1.0</ccts:VersionID>
5195                     <ccts:DefinitionText>The mime type of the binary object.</ccts:DefinitionText>
5196                     <ccts:CardinalityText>1</ccts:CardinalityText>
5197                     <ccts:PropertyTermName>Mime</ccts:PropertyTermName>
5198                     <ccts:RepresentationTermName>Code</ccts:RepresentationTermName>
5199                 </xsd:documentation>
5200             </xsd:annotation>
5201         </xsd:attribute>
5202         <xsd:attribute name="pictureEncodingCode" type="xsd:token" use="optional">
5203             <xsd:annotation>
5204                 <xsd:documentation xml:lang="en">
5205                     <ccts:UniqueID>UDT000004-SC4</ccts:UniqueID>
5206                     <ccts:CategoryCode>SC</ccts:CategoryCode>
5207                     <ccts:DictionaryEntryName>Picture. Encoding. Code</ccts:DictionaryEntryName>
5208                     <ccts:VersionID>1.0</ccts:VersionID>
5209                     <ccts:DefinitionText>Specifies the decoding algorithm of the binary object.</ccts:DefinitionText>
5210                     <ccts:CardinalityText>0..1</ccts:CardinalityText>
5211                     <ccts:PropertyTermName>Encoding</ccts:PropertyTermName>
5212                     <ccts:RepresentationTermName>Code</ccts:RepresentationTermName>
5213                 </xsd:documentation>
5214             </xsd:annotation>
5215         </xsd:attribute>
5216         <xsd:attribute name="pictureURI" type="xsd:anyURI" use="optional">
5217             <xsd:annotation>
5218                 <xsd:documentation xml:lang="en">
5219                     <ccts:UniqueID>UDT000004-SC6</ccts:UniqueID>
5220                     <ccts:CategoryCode>SC</ccts:CategoryCode>
5221                     <ccts:DictionaryEntryName>Picture. Uniform Resource. Identifier</ccts:DictionaryEntryName>
5222                     <ccts:VersionID>1.0</ccts:VersionID>

```

```

5223         <ccts:DefinitionText>The Uniform Resource Identifier that identifies where the binary object is
5224 located.i</ccts:DefinitionText>
5225         <ccts:CardinalityText>0..1</ccts:CardinalityText>
5226         <ccts:PropertyTermName>Uniform Resource</ccts:PropertyTermName>
5227         <ccts:RepresentationTermName>Identifier</ccts:RepresentationTermName>
5228         </xsd:documentation>
5229     </xsd:annotation>
5230 </xsd:attribute>
5231 <xsd:attribute name="pictureFilenameText" type="xsd:token" use="optional">
5232     <xsd:annotation>
5233         <xsd:documentation xml:lang="en">
5234             <ccts:UniqueID>UDT000004-SC7</ccts:UniqueID>
5235             <ccts:CategoryCode>SC</ccts:CategoryCode>
5236             <ccts:DictionaryEntryName>Picture. Filename. Text</ccts:DictionaryEntryName>
5237             <ccts:VersionID>1.0</ccts:VersionID>
5238             <ccts:DefinitionText>The filename of the binary object.</ccts:DefinitionText>
5239             <ccts:CardinalityText>0..1</ccts:CardinalityText>
5240             <ccts:PropertyTermName>Filename</ccts:PropertyTermName>
5241             <ccts:RepresentationTermName>Text</ccts:RepresentationTermName>
5242         </xsd:documentation>
5243     </xsd:annotation>
5244 </xsd:attribute>
5245 </xsd:extension>
5246 </xsd:simpleContent>
5247 </xsd:complexType>
5248 <!-- ===== Secondary RT: Sound. Type          ===== -->
5249 <!-- =====>
5250 <xsd:complexType name="SoundType">
5251     <xsd:annotation>
5252         <xsd:documentation xml:lang="en">
5253             <ccts:UniqueID>UDT000005</ccts:UniqueID>
5254             <ccts:CategoryCode>UDT</ccts:CategoryCode>
5255             <ccts:DictionaryEntryName>Sound. Type</ccts:DictionaryEntryName>
5256             <ccts:VersionID>1.0</ccts:VersionID>
5257             <ccts:DefinitionText>A diagram, graph, mathematical curves, or similar representation.</ccts:DefinitionText>
5258             <ccts:RepresentationTermName>Sound</ccts:RepresentationTermName>
5259         </xsd:documentation>
5260     </xsd:annotation>
5261     <xsd:simpleContent>
5262         <xsd:extension base="xsd:base64Binary">
5263             <xsd:attribute name="soundFormatText" type="xsd:token" use="optional">
5264                 <xsd:annotation>
5265                     <xsd:documentation xml:lang="en">
5266                         <ccts:UniqueID>UDT000005-SC2</ccts:UniqueID>
5267                         <ccts:CategoryCode>SC</ccts:CategoryCode>
5268                         <ccts:DictionaryEntryName>Sound. Format. Text</ccts:DictionaryEntryName>
5269                         <ccts:VersionID>1.0</ccts:VersionID>
5270                         <ccts:DefinitionText>The format of the binary content.</ccts:DefinitionText>
5271                         <ccts:CardinalityText>0..1</ccts:CardinalityText>
5272                         <ccts:PropertyTermName>Format</ccts:PropertyTermName>
5273                         <ccts:RepresentationTermName>Text</ccts:RepresentationTermName>
5274                     </xsd:documentation>
5275                 </xsd:annotation>
5276             </xsd:attribute>
5277             <xsd:attribute name="soundMimeCode"
5278 type="clmIANAMIMEMediaTypes:BinaryObjectMimeTypeCodeContent" use="required">
5279                 <xsd:annotation>
5280                     <xsd:documentation xml:lang="en">
5281                         <ccts:UniqueID>UDT000005-SC3</ccts:UniqueID>
5282                         <ccts:CategoryCode>SC</ccts:CategoryCode>
5283                         <ccts:DictionaryEntryName>Sound. Mime. Code</ccts:DictionaryEntryName>
5284                         <ccts:VersionID>1.0</ccts:VersionID>
5285                         <ccts:DefinitionText>The mime type of the binary object.</ccts:DefinitionText>
5286                         <ccts:CardinalityText>1</ccts:CardinalityText>
5287                         <ccts:PropertyTermName>Mime</ccts:PropertyTermName>
5288                         <ccts:RepresentationTermName>Code</ccts:RepresentationTermName>
5289                     </xsd:documentation>
5290                 </xsd:annotation>
5291             </xsd:attribute>
5292             <xsd:attribute name="soundEncodingCode" type="xsd:token" use="optional">
5293                 <xsd:annotation>

```

```

5294         <xsd:documentation xml:lang="en">
5295             <ccts:UniqueID>UDT000005-SC4</ccts:UniqueID>
5296             <ccts:CategoryCode>SC</ccts:CategoryCode>
5297             <ccts:DictionaryEntryName>Sound. Encoding. Code</ccts:DictionaryEntryName>
5298             <ccts:VersionID>1.0</ccts:VersionID>
5299             <ccts:DefinitionText>Specifies the decoding algorithm of the binary object.</ccts:DefinitionText>
5300             <ccts:CardinalityText>0..1</ccts:CardinalityText>
5301             <ccts:PropertyTermName>Encoding</ccts:PropertyTermName>
5302             <ccts:RepresentationTermName>Code</ccts:RepresentationTermName>
5303         </xsd:documentation>
5304     </xsd:annotation>
5305 </xsd:attribute>
5306 <xsd:attribute name="soundURI" type="xsd:anyURI" use="optional">
5307     <xsd:annotation>
5308         <xsd:documentation xml:lang="en">
5309             <ccts:UniqueID>UDT000005-SC6</ccts:UniqueID>
5310             <ccts:CategoryCode>SC</ccts:CategoryCode>
5311             <ccts:DictionaryEntryName>Sound. Uniform Resource. Identifier</ccts:DictionaryEntryName>
5312             <ccts:VersionID>1.0</ccts:VersionID>
5313             <ccts:DefinitionText>The Uniform Resource Identifier that identifies where the binary object is
5314 located.i</ccts:DefinitionText>
5315             <ccts:CardinalityText>0..1</ccts:CardinalityText>
5316             <ccts:PropertyTermName>Uniform Resource</ccts:PropertyTermName>
5317             <ccts:RepresentationTermName>Identifier</ccts:RepresentationTermName>
5318         </xsd:documentation>
5319     </xsd:annotation>
5320 </xsd:attribute>
5321 <xsd:attribute name="soundFilenameText" type="xsd:token" use="optional">
5322     <xsd:annotation>
5323         <xsd:documentation xml:lang="en">
5324             <ccts:UniqueID>UDT000005-SC7</ccts:UniqueID>
5325             <ccts:CategoryCode>SC</ccts:CategoryCode>
5326             <ccts:DictionaryEntryName>Sound. Filename.Text</ccts:DictionaryEntryName>
5327             <ccts:VersionID>1.0</ccts:VersionID>
5328             <ccts:DefinitionText>The filename of the binary object.</ccts:DefinitionText>
5329             <ccts:CardinalityText>0..1</ccts:CardinalityText>
5330             <ccts:PropertyTermName>Filename</ccts:PropertyTermName>
5331             <ccts:RepresentationTermName>Text</ccts:RepresentationTermName>
5332         </xsd:documentation>
5333     </xsd:annotation>
5334 </xsd:attribute>
5335 </xsd:extension>
5336 </xsd:simpleContent>
5337 </xsd:complexType>
5338 <!-- ===== Secondary RT: Video. Type                ===== -->
5339 <!-- =====>
5340 <xsd:complexType name="VideoType">
5341     <xsd:annotation>
5342         <xsd:documentation xml:lang="en">
5343             <ccts:UniqueID>UDT000006</ccts:UniqueID>
5344             <ccts:CategoryCode>UDT</ccts:CategoryCode>
5345             <ccts:DictionaryEntryName>Video. Type</ccts:DictionaryEntryName>
5346             <ccts:VersionID>1.0</ccts:VersionID>
5347             <ccts:DefinitionText>A diagram, graph, mathematical curves, or similar representation.</ccts:DefinitionText>
5348             <ccts:RepresentationTermName>Graphic</ccts:RepresentationTermName>
5349         </xsd:documentation>
5350     </xsd:annotation>
5351 <xsd:simpleContent>
5352     <xsd:extension base="xsd:base64Binary">
5353         <xsd:attribute name="videoFormatText" type="xsd:token" use="optional">
5354             <xsd:annotation>
5355                 <xsd:documentation xml:lang="en">
5356                     <ccts:UniqueID>UDT000006-SC2</ccts:UniqueID>
5357                     <ccts:CategoryCode>SC</ccts:CategoryCode>
5358                     <ccts:DictionaryEntryName>Video. Format. Text</ccts:DictionaryEntryName>
5359                     <ccts:VersionID>1.0</ccts:VersionID>
5360                     <ccts:DefinitionText>The format of the binary content.</ccts:DefinitionText>
5361                     <ccts:CardinalityText>0..1</ccts:CardinalityText>
5362                     <ccts:PropertyTermName>Format</ccts:PropertyTermName>
5363                     <ccts:RepresentationTermName>Text</ccts:RepresentationTermName>
5364                 </xsd:documentation>

```

```

5365         </xsd:annotation>
5366     </xsd:attribute>
5367     <xsd:attribute name="videoMimeType"
5368 type="clm:ANAMIMEMediaTypes:BinaryObjectMimeType" use="required">
5369         <xsd:annotation>
5370             <xsd:documentation xml:lang="en">
5371                 <ccts:UniqueID>UDT000006-SC3</ccts:UniqueID>
5372                 <ccts:CategoryCode>SC</ccts:CategoryCode>
5373                 <ccts:DictionaryEntryName>Video. Mime. Code</ccts:DictionaryEntryName>
5374                 <ccts:VersionID>1.0</ccts:VersionID>
5375                 <ccts:DefinitionText>The mime type of the binary object.</ccts:DefinitionText>
5376                 <ccts:CardinalityText>1</ccts:CardinalityText>
5377                 <ccts:PropertyTermName>Mime</ccts:PropertyTermName>
5378                 <ccts:RepresentationTermName>Code</ccts:RepresentationTermName>
5379             </xsd:documentation>
5380         </xsd:annotation>
5381     </xsd:attribute>
5382     <xsd:attribute name="videoEncodingCode" type="xsd:token" use="optional">
5383         <xsd:annotation>
5384             <xsd:documentation xml:lang="en">
5385                 <ccts:UniqueID>UDT000006-SC4</ccts:UniqueID>
5386                 <ccts:CategoryCode>SC</ccts:CategoryCode>
5387                 <ccts:DictionaryEntryName>Video. Encoding. Code</ccts:DictionaryEntryName>
5388                 <ccts:VersionID>1.0</ccts:VersionID>
5389                 <ccts:DefinitionText>Specifies the decoding algorithm of the binary object.</ccts:DefinitionText>
5390                 <ccts:CardinalityText>0..1</ccts:CardinalityText>
5391                 <ccts:PropertyTermName>Encoding</ccts:PropertyTermName>
5392                 <ccts:RepresentationTermName>Code</ccts:RepresentationTermName>
5393             </xsd:documentation>
5394         </xsd:annotation>
5395     </xsd:attribute>
5396     <xsd:attribute name="videoURI" type="xsd:anyURI" use="optional">
5397         <xsd:annotation>
5398             <xsd:documentation xml:lang="en">
5399                 <ccts:UniqueID>UDT000006-SC6</ccts:UniqueID>
5400                 <ccts:CategoryCode>SC</ccts:CategoryCode>
5401                 <ccts:DictionaryEntryName>Video. Uniform Resource. Identifier</ccts:DictionaryEntryName>
5402                 <ccts:VersionID>1.0</ccts:VersionID>
5403                 <ccts:DefinitionText>The Uniform Resource Identifier that identifies where the binary object is
5404 located.i</ccts:DefinitionText>
5405                 <ccts:CardinalityText>0..1</ccts:CardinalityText>
5406                 <ccts:PropertyTermName>Uniform Resource</ccts:PropertyTermName>
5407                 <ccts:RepresentationTermName>Identifier</ccts:RepresentationTermName>
5408             </xsd:documentation>
5409         </xsd:annotation>
5410     </xsd:attribute>
5411     <xsd:attribute name="videoFilenameText" type="xsd:token" use="optional">
5412         <xsd:annotation>
5413             <xsd:documentation xml:lang="en">
5414                 <ccts:UniqueID>UDT000006-SC7</ccts:UniqueID>
5415                 <ccts:CategoryCode>SC</ccts:CategoryCode>
5416                 <ccts:DictionaryEntryName>video. Filename.Text</ccts:DictionaryEntryName>
5417                 <ccts:VersionID>1.0</ccts:VersionID>
5418                 <ccts:DefinitionText>The filename of the binary object.</ccts:DefinitionText>
5419                 <ccts:CardinalityText>0..1</ccts:CardinalityText>
5420                 <ccts:PropertyTermName>Filename</ccts:PropertyTermName>
5421                 <ccts:RepresentationTermName>Text</ccts:RepresentationTermName>
5422             </xsd:documentation>
5423         </xsd:annotation>
5424     </xsd:attribute>
5425     </xsd:extension>
5426 </xsd:simpleContent>
5427 </xsd:complexType>
5428 <!-- ===== Primary RT: Code. Type ===== -->
5429 <!-- =====>
5430 <xsd:complexType name="CodeType">
5431     <xsd:annotation>
5432         <xsd:documentation xml:lang="en">
5433             <ccts:UniqueID>UDT000007</ccts:UniqueID>
5434             <ccts:CategoryCode>UDT</ccts:CategoryCode>
5435             <ccts:DictionaryEntryName>Code. Type</ccts:DictionaryEntryName>

```

5436 <ccts:VersionID>1.0</ccts:VersionID>
5437 <ccts:DefinitionText>A character string (letters, figures, or symbols) that for brevity and/or language
5438 independence may be used to represent or replace a definitive value or text of an attribute together with relevant
5439 supplementary information.</ccts:DefinitionText>
5440 <ccts:RepresentationTermName>Code</ccts:RepresentationTermName>
5441 </xsd:documentation>
5442 </xsd:annotation>
5443 <xsd:simpleContent>
5444 <xsd:extension base="xsd:token">
5445 <xsd:attribute name="codeNameText" type="xsd:token" use="optional">
5446 <xsd:annotation>
5447 <xsd:documentation xml:lang="en">
5448 <ccts:UniqueID>UDT000007-SC7</ccts:UniqueID>
5449 <ccts:CategoryCode>SC</ccts:CategoryCode>
5450 <ccts:DictionaryEntryName>Code. Name. Text</ccts:DictionaryEntryName>
5451 <ccts:VersionID>1.0</ccts:VersionID>
5452 <ccts:DefinitionText>The textual equivalent of the code content
5453 component.</ccts:DefinitionText>
5454 <ccts:CardinalityText>0..1</ccts:CardinalityText>
5455 <ccts:PropertyTermName>Name</ccts:PropertyTermName>
5456 <ccts:RepresentationTermName>Text</ccts:RepresentationTermName>
5457 <ccts:UsageRuleText> It is recommended that the name of a code is provided in this SC rather
5458 than in a separately defined BBIE.</ccts:UsageRuleText>
5459 </xsd:documentation>
5460 </xsd:annotation>
5461 </xsd:attribute>
5462 <xsd:attribute name="codeLanguageID" type="xsd:language" use="optional">
5463 <xsd:annotation>
5464 <xsd:documentation xml:lang="en">
5465 <ccts:UniqueID>UDT000007-SC8</ccts:UniqueID>
5466 <ccts:CategoryCode>SC</ccts:CategoryCode>
5467 <ccts:DictionaryEntryName>Code. Language. Identifier</ccts:DictionaryEntryName>
5468 <ccts:VersionID>1.0</ccts:VersionID>
5469 <ccts:DefinitionText>The identifier of the language used in the code name.</ccts:DefinitionText>
5470 <ccts:CardinalityText>0..1</ccts:CardinalityText>
5471 <ccts:PropertyTermName>Language</ccts:PropertyTermName>
5472 <ccts:RepresentationTermName>Identifier</ccts:RepresentationTermName>
5473 <ccts:UsageRuleText>Identifies the language in which the code name is
5474 presented.</ccts:UsageRuleText>
5475 </xsd:documentation>
5476 </xsd:annotation>
5477 </xsd:attribute>
5478 <xsd:attribute name="codeListURI" type="xsd:anyURI" use="optional">
5479 <xsd:annotation>
5480 <xsd:documentation xml:lang="en">
5481 <ccts:UniqueID>UDT000007-SC9</ccts:UniqueID>
5482 <ccts:CategoryCode>SC</ccts:CategoryCode>
5483 <ccts:DictionaryEntryName>Code List. Uniform Resource.
5484 Identifier</ccts:DictionaryEntryName>
5485 <ccts:VersionID>1.0</ccts:VersionID>
5486 <ccts:DefinitionText>The Uniform Resource Identifier that identifies where the code list is
5487 located.</ccts:DefinitionText>
5488 <ccts:CardinalityText>0..1</ccts:CardinalityText>
5489 <ccts:PropertyTermName>Uniform Resource</ccts:PropertyTermName>
5490 <ccts:RepresentationTermName>Identifier</ccts:RepresentationTermName>
5491 <ccts:UsageRuleText>May be used to identify the location of the code list if available across the
5492 world wide web</ccts:UsageRuleText>
5493 </xsd:documentation>
5494 </xsd:annotation>
5495 </xsd:attribute>
5496 <xsd:attribute name="codeListSchemeURI" type="xsd:anyURI" use="optional">
5497 <xsd:annotation>
5498 <xsd:documentation xml:lang="en">
5499 <ccts:UniqueID>UDT000007-SC9</ccts:UniqueID>
5500 <ccts:CategoryCode>SC</ccts:CategoryCode>
5501 <ccts:DictionaryEntryName>Code List Scheme. Uniform Resource.
5502 Identifier</ccts:DictionaryEntryName>
5503 <ccts:VersionID>1.0</ccts:VersionID>
5504 <ccts:DefinitionText>The Uniform Resource identifier that identifies where the code list scheme
5505 is located.</ccts:DefinitionText>
5506 <ccts:CardinalityText>0..1</ccts:CardinalityText>

```

5507         <ccts:PropertyTermName>Uniform Resource</ccts:PropertyTermName>
5508         <ccts:RepresentationTermName>Identifier</ccts:RepresentationTermName>
5509         <ccts:UsageRuleText>May be used to identify the location of the schema of the code list if
5510 available across the world wide web</ccts:UsageRuleText>
5511     </xsd:documentation>
5512 </xsd:annotation>
5513 </xsd:attribute>
5514 </xsd:extension>
5515 </xsd:simpleContent>
5516 </xsd:complexType>
5517 <!-- ===== Primary RT: Date Time. Type ===== -->
5518 <!-- ===== -->
5519 <xsd:simpleType name="DateTimeType">
5520     <xsd:annotation>
5521         <xsd:documentation xml:lang="en">
5522             <ccts:UniqueID>UDT000008</ccts:UniqueID>
5523             <ccts:CategoryCode>UDT</ccts:CategoryCode>
5524             <ccts:DictionaryEntryName>Date Time. Type</ccts:DictionaryEntryName>
5525             <ccts:VersionID>1.0</ccts:VersionID>
5526             <ccts:DefinitionText>A particular point in the progression of time together with the relevant supplementary
5527 information.</ccts:DefinitionText>
5528             <ccts:RepresentationTermName>Date Time</ccts:RepresentationTermName>
5529             <ccts:UsageRuleText>This lexical representation is the [ISO 8601] extended format CCYY-MM-
5530 DDThh:mm:ss.</ccts:UsageRuleText>
5531         </xsd:documentation>
5532     </xsd:annotation>
5533     <xsd:restriction base="xsd:dateTime"/>
5534 </xsd:simpleType>
5535 <!-- ===== Secondary RT: Date. Type ===== -->
5536 <!-- ===== -->
5537 <xsd:simpleType name="DateType">
5538     <xsd:annotation>
5539         <xsd:documentation xml:lang="en">
5540             <ccts:UniqueID>UDT000009</ccts:UniqueID>
5541             <ccts:CategoryCode>UDT</ccts:CategoryCode>
5542             <ccts:DictionaryEntryName>Date. Type</ccts:DictionaryEntryName>
5543             <ccts:VersionID>1.0</ccts:VersionID>
5544             <ccts:DefinitionText>One calendar day according the Gregorian calendar.</ccts:DefinitionText>
5545             <ccts:RepresentationTermName>Date</ccts:RepresentationTermName>
5546             <ccts:UsageRuleText>The lexical representation for date is the reduced (right truncated) lexical
5547 representation for dateTime: CCYY-MM-DD.</ccts:UsageRuleText>
5548         </xsd:documentation>
5549     </xsd:annotation>
5550     <xsd:restriction base="xsd:date"/>
5551 </xsd:simpleType>
5552 <!-- ===== Secondary RT: Time. Type ===== -->
5553 <!-- ===== -->
5554 <xsd:simpleType name="TimeType">
5555     <xsd:annotation>
5556         <xsd:documentation xml:lang="en">
5557             <ccts:UniqueID>UDT000010</ccts:UniqueID>
5558             <ccts:CategoryCode>UDT</ccts:CategoryCode>
5559             <ccts:DictionaryEntryName>Time. Type</ccts:DictionaryEntryName>
5560             <ccts:VersionID>1.0</ccts:VersionID>
5561             <ccts:DefinitionText>The instance of time that occurs every day.</ccts:DefinitionText>
5562             <ccts:RepresentationTermName>Time</ccts:RepresentationTermName>
5563             <ccts:UsageRuleText>The lexical representation for time is the left truncated lexical representation for
5564 dateTime: hh:mm:ss.sss with optional following time zone indicator.</ccts:UsageRuleText>
5565         </xsd:documentation>
5566     </xsd:annotation>
5567     <xsd:restriction base="xsd:time"/>
5568 </xsd:simpleType>
5569 <!-- ===== Primary RT: Identifier. Type ===== -->
5570 <!-- ===== -->
5571 <xsd:complexType name="IdentifierType">
5572     <xsd:annotation>
5573         <xsd:documentation xml:lang="en">
5574             <ccts:UniqueID>UDT000011</ccts:UniqueID>
5575             <ccts:CategoryCode>UDT</ccts:CategoryCode>
5576             <ccts:DictionaryEntryName>Identifier. Type</ccts:DictionaryEntryName>
5577             <ccts:VersionID>1.0</ccts:VersionID>

```



```

5578         <ccts:DefinitionText>A character string to identify and distinguish uniquely, one instance of an object in an
5579 identification scheme from all other objects in the same scheme together with relevant supplementary
5580 information.</ccts:DefinitionText>
5581         <ccts:RepresentationTermName>Identifier</ccts:RepresentationTermName>
5582     </xsd:documentation>
5583 </xsd:annotation>
5584 <xsd:simpleContent>
5585     <xsd:extension base="xsd:token">
5586     <xsd:attribute name="identificationSchemeDataURI" type="xsd:anyURI" use="optional">
5587     <xsd:annotation>
5588         <xsd:documentation xml:lang="en">
5589             <ccts:UniqueID>UDT0000011-SC7</ccts:UniqueID>
5590             <ccts:CategoryCode>SC</ccts:CategoryCode>
5591             <ccts:DictionaryEntryName>Identification Scheme Data. Uniform Resource.
5592 Identifier</ccts:DictionaryEntryName>
5593             <ccts:VersionID>1.0</ccts:VersionID>
5594             <ccts:DefinitionText>The Uniform Resource Identifier that identifies where the identification
5595 scheme data is located.</ccts:DefinitionText>
5596             <ccts:CardinalityText>0..1</ccts:CardinalityText>
5597             <ccts:PropertyTermName>>Uniform Resource</ccts:PropertyTermName>
5598             <ccts:RepresentationTermName>Identifier</ccts:RepresentationTermName>
5599             <ccts:UsageRuleText>May be used to identify the location of the identification scheme if
5600 available across the world wide web.</ccts:UsageRuleText>
5601         </xsd:documentation>
5602     </xsd:annotation>
5603 </xsd:attribute>
5604 <xsd:attribute name="identificationSchemeURI" type="xsd:anyURI" use="optional">
5605     <xsd:annotation>
5606         <xsd:documentation xml:lang="en">
5607             <ccts:UniqueID>UDT0000011-SC8</ccts:UniqueID>
5608             <ccts:CategoryCode>SC</ccts:CategoryCode>
5609             <ccts:DictionaryEntryName>Identification Scheme. Uniform Resource.
5610 Identifier</ccts:DictionaryEntryName>
5611             <ccts:VersionID>1.0</ccts:VersionID>
5612             <ccts:DefinitionText>The Uniform Resource Identifier that identifies where the identification
5613 scheme is located.</ccts:DefinitionText>
5614             <ccts:CardinalityText>0..1</ccts:CardinalityText>
5615             <ccts:PropertyTermName>Uniform Resource</ccts:PropertyTermName>
5616             <ccts:RepresentationTermName>Identifier</ccts:RepresentationTermName>
5617             <ccts:UsageRuleText>May be used to identify the location of the schema of the identification
5618 scheme if available across the world wide web.</ccts:UsageRuleText>
5619         </xsd:documentation>
5620     </xsd:annotation>
5621 </xsd:attribute>
5622 </xsd:extension>
5623 </xsd:simpleContent>
5624 </xsd:complexType>
5625 <!-- ===== Primary RT: Indicator. Type ===== -->
5626 <!-- =====>
5627 <xsd:simpleType name="IndicatorType">
5628     <xsd:annotation>
5629         <xsd:documentation xml:lang="en">
5630             <ccts:UniqueID>UDT0000012</ccts:UniqueID>
5631             <ccts:CategoryCode>UDT</ccts:CategoryCode>
5632             <ccts:DictionaryEntryName>Indicator. Type</ccts:DictionaryEntryName>
5633             <ccts:VersionID>1.0</ccts:VersionID>
5634             <ccts:DefinitionText>A list of two mutually exclusive Boolean values that express the only possible states of
5635 a property.</ccts:DefinitionText>
5636             <ccts:RepresentationTermName>Indicator</ccts:RepresentationTermName>
5637         </xsd:documentation>
5638     </xsd:annotation>
5639     <xsd:restriction base="xsd:boolean">
5640         <xsd:pattern value="false"/>
5641         <xsd:pattern value="true"/>
5642     </xsd:restriction>
5643 </xsd:simpleType>
5644 <!-- ===== Primary RT: Measure. Type ===== -->
5645 <!-- =====>
5646 <xsd:complexType name="MeasureType">
5647     <xsd:annotation>
5648         <xsd:documentation xml:lang="en">

```

```

5649     <ccts:UniqueID>UDT0000013</ccts:UniqueID>
5650     <ccts:CategoryCode>UDT</ccts:CategoryCode>
5651     <ccts:DictionaryEntryName>Measure. Type</ccts:DictionaryEntryName>
5652     <ccts:VersionID>1.0</ccts:VersionID>
5653     <ccts:DefinitionText>A numeric value determined by measuring an object along with the specified unit of
5654 measure.</ccts:DefinitionText>
5655     <ccts:RepresentationTermName>Measure</ccts:RepresentationTermName>
5656     <ccts:PropertyTermName>Type</ccts:PropertyTermName>
5657     </xsd:documentation>
5658 </xsd:annotation>
5659 <xsd:simpleContent>
5660   <xsd:extension base="xsd:decimal">
5661     <xsd:attribute name="measureUnitCode" type="clm66411:UnitCodeContentType" use="required">
5662       <xsd:annotation>
5663         <xsd:documentation xml:lang="en">
5664           <ccts:UniqueID>UDT0000013-SC2</ccts:UniqueID>
5665           <ccts:CategoryCode>SC</ccts:CategoryCode>
5666           <ccts:DictionaryEntryName>Measure. Unit. Code</ccts:DictionaryEntryName>
5667           <ccts:VersionID>1.0</ccts:VersionID>
5668           <ccts:DefinitionText>The type of unit of measure.</ccts:DefinitionText>
5669           <ccts:CardinalityText>0..1</ccts:CardinalityText>
5670           <ccts:PropertyTermName>MeasureUnit</ccts:PropertyTermName>
5671           <ccts:RepresentationTermName>Code</ccts:RepresentationTermName>
5672           <ccts:UsageRuleText>Use the Common Codes of the UNECE Rec 20.</ccts:UsageRuleText>
5673           <ccts:UsageRuleText>It is recommended that the measure unit code is provided in this SC
5674 rather than in a separately defined BBIE.</ccts:UsageRuleText>
5675         </xsd:documentation>
5676       </xsd:annotation>
5677     </xsd:attribute>
5678   </xsd:extension>
5679 </xsd:simpleContent>
5680 </xsd:complexType>
5681 <!-- ===== Primary RT: Numeric. Type ===== -->
5682 <!-- ===== -->
5683 <xsd:simpleType name="NumericType">
5684   <xsd:annotation>
5685     <xsd:documentation xml:lang="en">
5686       <ccts:UniqueID>UDT0000014</ccts:UniqueID>
5687       <ccts:CategoryCode>UDT</ccts:CategoryCode>
5688       <ccts:DictionaryEntryName>Numeric. Type</ccts:DictionaryEntryName>
5689       <ccts:VersionID>1.0</ccts:VersionID>
5690       <ccts:DefinitionText>Numeric information that is assigned or is determined by calculation, counting, or
5691 sequencing. It does not require a unit of quantity or unit of measure.</ccts:DefinitionText>
5692       <ccts:RepresentationTermName>Numeric</ccts:RepresentationTermName>
5693     </xsd:documentation>
5694   </xsd:annotation>
5695   <xsd:restriction base="xsd:decimal"/>
5696 </xsd:simpleType>
5697 <!-- ===== Secondary RT: Value. Type ===== -->
5698 <!-- ===== -->
5699 <xsd:simpleType name="ValueType">
5700   <xsd:annotation>
5701     <xsd:documentation xml:lang="en">
5702       <ccts:UniqueID>UDT0000015</ccts:UniqueID>
5703       <ccts:CategoryCode>UDT</ccts:CategoryCode>
5704       <ccts:DictionaryEntryName>Value. Type</ccts:DictionaryEntryName>
5705       <ccts:DefinitionText>Numeric information that is assigned or is determined by calculation, counting, or
5706 sequencing. It does not require a unit of quantity or unit of measure.</ccts:DefinitionText>
5707       <ccts:RepresentationTermName>Value</ccts:RepresentationTermName>
5708     </xsd:documentation>
5709   </xsd:annotation>
5710   <xsd:restriction base="xsd:decimal"/>
5711 </xsd:simpleType>
5712 <!-- ===== Secondary RT: Percent. Type ===== -->
5713 <!-- ===== -->
5714 <xsd:simpleType name="PercentType">
5715   <xsd:annotation>
5716     <xsd:documentation xml:lang="en">
5717       <ccts:UniqueID>UDT0000016</ccts:UniqueID>
5718       <ccts:CategoryCode>UDT</ccts:CategoryCode>
5719       <ccts:DictionaryEntryName>Percent. Type</ccts:DictionaryEntryName>

```

```

5720         <ccts:DefinitionText>Numeric information that is assigned or is determined by calculation, counting, or
5721 sequencing. It does not require a unit of quantity or unit of measure.</ccts:DefinitionText>
5722         <ccts:RepresentationTermName>Percent</ccts:RepresentationTermName>
5723     </xsd:documentation>
5724 </xsd:annotation>
5725     <xsd:restriction base="xsd:decimal"/>
5726 </xsd:simpleType>
5727 <!-- ===== Secondary RT: Rate. Type                ===== -->
5728 <!-- ===== -->
5729 <xsd:simpleType name="RateType">
5730     <xsd:annotation>
5731         <xsd:documentation xml:lang="en">
5732             <ccts:UniqueID>UDT0000017</ccts:UniqueID>
5733             <ccts:CategoryCode>UDT</ccts:CategoryCode>
5734             <ccts:DictionaryEntryName>Rate. Type</ccts:DictionaryEntryName>
5735             <ccts:DefinitionText>Numeric information that is assigned or is determined by calculation, counting, or
5736 sequencing. It does not require a unit of quantity or unit of measure.</ccts:DefinitionText>
5737             <ccts:RepresentationTermName>Rate</ccts:RepresentationTermName>
5738         </xsd:documentation>
5739     </xsd:annotation>
5740     <xsd:restriction base="xsd:decimal"/>
5741 </xsd:simpleType>
5742 <!-- ===== Primary RT: Quantity. Type                ===== -->
5743 <!-- ===== -->
5744 <xsd:complexType name="QuantityType">
5745     <xsd:annotation>
5746         <xsd:documentation xml:lang="en">
5747             <ccts:UniqueID>UDT0000018</ccts:UniqueID>
5748             <ccts:CategoryCode>UDT</ccts:CategoryCode>
5749             <ccts:DictionaryEntryName>Quantity. Type</ccts:DictionaryEntryName>
5750             <ccts:VersionID>1.0</ccts:VersionID>
5751             <ccts:DefinitionText>A counted number of non-monetary units possibly including
5752 fractions.</ccts:DefinitionText>
5753             <ccts:RepresentationTermName>Quantity</ccts:RepresentationTermName>
5754         </xsd:documentation>
5755     </xsd:annotation>
5756     <xsd:simpleContent>
5757         <xsd:extension base="xsd:decimal">
5758             <xsd:attribute name="unitCode" type="clm66411:UnitCodeContentType" use="optional">
5759                 <xsd:annotation>
5760                     <xsd:documentation xml:lang="en">
5761                         <ccts:UniqueID>UDT0000018-SC2</ccts:UniqueID>
5762                         <ccts:CategoryCode>SC</ccts:CategoryCode>
5763                         <ccts:DictionaryEntryName>Quantity. Unit. Code</ccts:DictionaryEntryName>
5764                         <ccts:VersionID>1.0</ccts:VersionID>
5765                         <ccts:DefinitionText>The unit of the quantity</ccts:DefinitionText>
5766                         <ccts:CardinalityText>0..1</ccts:CardinalityText>
5767                         <ccts:PropertyTermName>Unit</ccts:PropertyTermName>
5768                         <ccts:RepresentationTermName>Code</ccts:RepresentationTermName>
5769                         <ccts:UsageRuleText>It is recommended that if a quantity unit code is provided it should be
5770 provided in this SC rather than in a separately defined BBIE.</ccts:UsageRuleText>
5771                         <ccts:UsageRuleText>If no code list is specified it is assumed to be UNECE Rec 20 (Common
5772 Codes).</ccts:UsageRuleText>
5773                     </xsd:documentation>
5774                 </xsd:annotation>
5775                 </xsd:attribute>
5776             </xsd:extension>
5777         </xsd:simpleContent>
5778     </xsd:complexType>
5779 <!-- ===== Primary RT: Text.Type                ===== -->
5780 <!-- ===== -->
5781 <xsd:complexType name="TextType">
5782     <xsd:annotation>
5783         <xsd:documentation xml:lang="en">
5784             <ccts:UniqueID>UDT0000019</ccts:UniqueID>
5785             <ccts:CategoryCode>UDT</ccts:CategoryCode>
5786             <ccts:DictionaryEntryName>Text. Type</ccts:DictionaryEntryName>
5787             <ccts:VersionID>1.0</ccts:VersionID>
5788             <ccts:DefinitionText>A character string (i.e. a finite set of characters) generally in the form of words of a
5789 language.</ccts:DefinitionText>
5790             <ccts:RepresentationTermName>Text</ccts:RepresentationTermName>

```

```

5791     </xsd:documentation>
5792 </xsd:annotation>
5793 <xsd:simpleContent>
5794     <xsd:extension base="xsd:string">
5795         <xsd:attribute name="languageID" type="clm5639:LanguageCodeContentType" use="optional">
5796             <xsd:annotation>
5797                 <xsd:documentation xml:lang="en">
5798                     <ccts:UniqueID>UDT0000019-SC2</ccts:UniqueID>
5799                     <ccts:CategoryCode>SC</ccts:CategoryCode>
5800                     <ccts:DictionaryEntryName>Language. Identifier</ccts:DictionaryEntryName>
5801                     <ccts:VersionID>1.0</ccts:VersionID>
5802                     <ccts:DefinitionText>The identifier of the language used in the content
5803 component.</ccts:DefinitionText>
5804                     <ccts:CardinalityText>0..1</ccts:CardinalityText>
5805                     <ccts:PropertyTermName>Language</ccts:PropertyTermName>
5806                     <ccts:RepresentationTermName>Identifier</ccts:RepresentationTermName>
5807                     <ccts:UsageRuleText>Identifies the language in which the text is
5808 presented.</ccts:UsageRuleText>
5809                 </xsd:documentation>
5810             </xsd:annotation>
5811         </xsd:attribute>
5812     <xsd:attribute name="languageLocaleID" type="xsd:token" use="optional">
5813         <xsd:annotation>
5814             <xsd:documentation xml:lang="en">
5815                 <ccts:UniqueID>UDT0000019-SC3</ccts:UniqueID>
5816                 <ccts:CategoryCode>SC</ccts:CategoryCode>
5817                 <ccts:DictionaryEntryName>Language. Locale. Identifier</ccts:DictionaryEntryName>
5818                 <ccts:VersionID>1.0</ccts:VersionID>
5819                 <ccts:DefinitionText>The identification of the locale of the language.</ccts:DefinitionText>
5820                 <ccts:CardinalityText>0..1</ccts:CardinalityText>
5821                 <ccts:PropertyTermName>Locale</ccts:PropertyTermName>
5822                 <ccts:RepresentationTermName>Identifier</ccts:RepresentationTermName>
5823                 <ccts:UsageRuleText>Not recommended for use.</ccts:UsageRuleText>
5824             </xsd:documentation>
5825         </xsd:annotation>
5826     </xsd:attribute>
5827 </xsd:extension>
5828 </xsd:simpleContent>
5829 </xsd:complexType>
5830 <!-- ===== Secondary RT: Name. Type ===== -->
5831 <!-- ===== -->
5832 <xsd:complexType name="NameType">
5833     <xsd:annotation>
5834         <xsd:documentation xml:lang="en">
5835             <ccts:UniqueID>UDT0000020</ccts:UniqueID>
5836             <ccts:CategoryCode>UDT</ccts:CategoryCode>
5837             <ccts:DictionaryEntryName>Name. Type</ccts:DictionaryEntryName>
5838             <ccts:VersionID>1.0</ccts:VersionID>
5839             <ccts:DefinitionText>A character string that consitutes the distinctive designation of a person, place, thing or
5840 concept.</ccts:DefinitionText>
5841             <ccts:RepresentationTermName>Name</ccts:RepresentationTermName>
5842         </xsd:documentation>
5843     </xsd:annotation>
5844 <xsd:simpleContent>
5845     <xsd:extension base="xsd:string">
5846         <xsd:attribute name="languageID" type="clm5639:LanguageCodeContentType" use="optional">
5847             <xsd:annotation>
5848                 <xsd:documentation xml:lang="en">
5849                     <ccts:UniqueID>UDT0000020-SC2</ccts:UniqueID>
5850                     <ccts:CategoryCode>SC</ccts:CategoryCode>
5851                     <ccts:DictionaryEntryName>Language. Identifier</ccts:DictionaryEntryName>
5852                     <ccts:VersionID>1.0</ccts:VersionID>
5853                     <ccts:DefinitionText>The identifier of the language used in the content
5854 component.</ccts:DefinitionText>
5855                     <ccts:CardinalityText>0..1</ccts:CardinalityText>
5856                     <ccts:PropertyTermName>Language</ccts:PropertyTermName>
5857                     <ccts:RepresentationTermName>Identifier</ccts:RepresentationTermName>
5858                     <ccts:UsageRuleText>Identifies the language in which the text is
5859 presented.</ccts:UsageRuleText>
5860                 </xsd:documentation>
5861             </xsd:annotation>

```

```

5862     </xsd:attribute>
5863     <xsd:attribute name="languageLocaleID" type="xsd:token" use="optional">
5864         <xsd:annotation>
5865             <xsd:documentation xml:lang="en">
5866                 <ccts:UniqueID>UDT0000020-SC3</ccts:UniqueID>
5867                 <ccts:CategoryCode>SC</ccts:CategoryCode>
5868                 <ccts:DictionaryEntryName> Language. Locale. Identifier</ccts:DictionaryEntryName>
5869                 <ccts:VersionID>1.0</ccts:VersionID>
5870                 <ccts:DefinitionText>The identification of the locale of the language.</ccts:DefinitionText>
5871                 <ccts:CardinalityText>0..1</ccts:CardinalityText>
5872                 <ccts:PropertyTermName>Locale</ccts:PropertyTermName>
5873                 <ccts:RepresentationTermName>Identifier</ccts:RepresentationTermName>
5874                 <ccts:UsageRuleText>Not recommended for use.</ccts:UsageRuleText>
5875             </xsd:documentation>
5876         </xsd:annotation>
5877     </xsd:attribute>
5878 </xsd:extension>
5879 </xsd:simpleContent>
5880 </xsd:complexType>
5881 </xsd:schema>
5882

```

5883 **Copyright Statement**

5884 Copyright © UN/CEFACT 2002. All Rights Reserved.

5885 This document and translations of it may be copied and furnished to others, and
5886 derivative works that comment on or otherwise explain it or assist in its implementation
5887 may be prepared, copied, published and distributed, in whole or in part, without
5888 restriction of any kind, provided that the above copyright notice and this paragraph are
5889 included on all such copies and derivative works. However, this document itself may not
5890 be modified in any way, such as by removing the copyright notice or references to
5891 UN/CEFACT except as required to translate it into languages other than English.

5892 The limited permissions granted above are perpetual and will not be revoked by
5893 UN/CEFACT or its successors or assigns.

5894 This document and the information contained herein is provided on an "AS IS" basis and
5895 UN/CEFACT DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED,
5896 INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE
5897 INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED
5898 WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR
5899 PURPOSE.

5900