UNITED NATIONS ECONOMIC COMMISSION FOR EUROPE

United Nations Centre for Trade Facilitation and Electronic Business

1

2

3

4

5

6

7

8 **UN/CEFACT**

9 **Core Components Business Document Assembly**

10 **Technical Specification**

11

12

13

14

15 **Version 1.1**

16 **31 August 2021**

17

18

19

20   **Abstract**

21   This document describes how to construct syntax independent Business
22   Documents based on UN/CEFACT Core Components and defines the basic
23   structure of Business Documents in a syntax and technology neutral way.

24   This document offers guidance to experts involved in the development and
25   standardization of UN/CEFACT standard *Business Document Types*. A *Business*
26   *Document Type* is a container of artifacts that describe the information
27   exchanged in a *Business Interaction*. The CCBDA specification can be employed
28   wherever business information is being shared or exchanged amongst and
29   between enterprises, governmental agencies and/or other organizations in an
30   open environment. This environment can be of a worldwide scope or restricted to
31   a specific *Business Context* (such as an industry or region).

32   The CCBDA specification is developed to identify how to construct *Business*
33   *Documents* from *Core Component* constructs (BIEs) in accordance with a
34   Business Requirements Specification (BRS) and a Requirements Specification
35   Mapping (RSM). CCBDA supports assembling CCTS based data models into
36   syntax independent business information exchange models that may
37   subsequently be rendered as syntax specific business information exchanges.

38   CCBDA forms the basis for standards development work of business analysts,
39   business users and information technology specialists supplying the content of
40   and implementing applications that will employ the UN/CEFACT Core
41   Component Document Library (CCDL).

42

Table of Contents

67

68

1   Status of this document

This Technical Specification has being developed in accordance with the UN/CEFACT/TRADE/22 Open Development Process for Technical Specifications and approved for publication by the UN/CEFACT Bureau.

2   Project Team Participants

We would like to recognize the following for their significant participation to the development of this document.

Project Team Leader: Sue Probert

Lead Editors: Mary Kay Blantz and Chris Hassler

Editing Team Members: Fred van Blommestein, Hidekazu Enjo, Hisanao Sugamata, Michel Bormans, Sylvia Webb, Satya Prasad Sahu, Gerhard Heemskerk

Contributors: Andreas Schultz, Audun Vennesland, Edmund Gray, Francois Vuilleumier, Klaus-Dieter Naujok, Michael Dill, Sander Huizinga

85   2.1   Legal Disclaimer

86   The views and specification expressed in this document are those of the authors
87   and are not necessarily those of their employers. The authors and their
88   employers specifically disclaim responsibility for any problems arising from
89   correct or incorrect implementation or use of this technical specification.

90   2.2   Contact Information

91   Lead Editor: Mary Kay Blantz, mblantz@sbcglobal.net

92   Editors: Hisanao Sugamata, hsedi0111@gmail.com

Gerhard Heemskerk, Gerhard.Heemskerk@kpnmail.nl

Hidekazu Enjo, enjouh@ciedi.jp92

3   Introduction

3.1 Scope and Focus

This Core Components Business Document Assembly Technical Specification (CCBDA) can be employed wherever business information is being shared or exchanged amongst and between enterprises, governmental agencies and/or other organizations in an open environment. This environment can be of a worldwide scope or restricted to a specific *Business Context* (such as an industry or region). Interoperability supported by this specification covers both interactive and batch exchanges of business data between applications. These applications may use Internet and Web based information exchanges as well as traditional Electronic Data Interchange (EDI) systems. CCBDA compliant models may serve as the computer readable part of an EDI Document Implementation Guideline. The exchanges may be service oriented as in Web Services or be Peer-to-peer.

This specification builds on the UN/CEFACT Modeling Methodology (UMM). UMM describes how inter organizational business information systems are designed by means of various views (see Annex A). The CCBDA specification is an elaboration of the UMM Business Information View, and defines how information is structured. CCBDA forms the basis for standards development work of business analysts, business users and information technology specialists supplying the content of and implementing applications that will employ the UN/CEFACT Core Component Message Library (CCML).

This specification does not specify a transmission protocol or any mechanisms to circumvent communication failure. Signalling (acknowledgements of receipt or acceptance) is outside the scope of this specification as are application processing considerations.

The way the information on envelopes, headers and information entities is represented in some syntax, such as XML or UN/EDIFACT, is outside the scope of this specification.

3.2 Audience

The audience of this technical specification includes but not limited to developers and implementers of e-business systems.

3.3 Structure of this document

Section 4 provides an overview of *Business Document Assembly.*

Section 5 provides rules for how a *Message Assembly* is named, defined and structured.

128 Section 6 describes how *Message Assemblies* and the *Business Information*
129 *Entities* they are associated with are defined in a specific *Business Context*.

130 Section 7 explains the use of *Constraints*.

131 Section 8 provides a list of terms and their definitions.

132 Sections 3, 5, 6, 7 and 8 are normative. All other sections are informative.

133 Implementations of this technical specification will be considered to be in full
134 compliance with this technical specification if they comply with the content of the
135 normative sections.
136

3.4   Notation

138 *Italics* – All words appearing in italics, when not titles or used for emphasis, are
139 special terms defined in Section 8.

140 `Courier` – All words appearing in bolded courier font are rules.

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

3.5   Related Documents

146 • ISO/TS 15000-5:2005: ebXML Core Components Technical Specification,
147   Version 2.01 (CCTS);
148 • UN/CEFACT Modelling Methodology Base Module Version 2.0;
149 • UN/CEFACT Modelling Methodology Foundation Module Version 2.0;
150 • UN/CEFACT Business Document Header Technical Specification;
151 • OMG Unified Modelling Language [UML] Specification, Version 1.4;
152 • OMG Object Constraint Language Specification [OCL] Version 1.1;
153 • UN/CEFACT Business Requirements Specification (BRS) Documentation
154   Template, Version 2.1 (out for public review);
155 • Requirements Specification Mapping (RSM) Documentation Template and
156   Conformity Rules Version 2.1 (out for public review);
157 • Key words for use in RFCs to Indicate Requirement Levels - Internet
158   Engineering Task Force, Request For Comments 2119, March 1997
159   http://www.ietf.org/rfc/rfc2119.txt?number=2119

160

161 **4   Overview**

162   This specification provides a specific methodology for assembling higher level
163   semantic *Business Documents* from *Business Information Entities*. *Business*
164   *Document Types* may be reusable and stored in Core Component Message
165   Libraries.

166   This technical specification will focus on the business information in a *Business*
167   *Document*, and will not include the specification of any protocol envelope needed
168   to transmit this information. The methodology makes use of *Business Information*
169   *Entities* as specified in the UN/CEFACT Core Component Technical Specification
170   v2.01 (CCTS) and ISO/TS 15000-5 (ebCCTS).

171   This technical specification defines the structure of *Business Documents* in an
172   abstract, syntax neutral way, as a data model (a UML Class Diagram) that can
173   be realised in various concrete syntaxes, such as XML (Extensible Markup
174   Language, a W3C Recommendation) and UN/EDIFACT (ISO 9735).
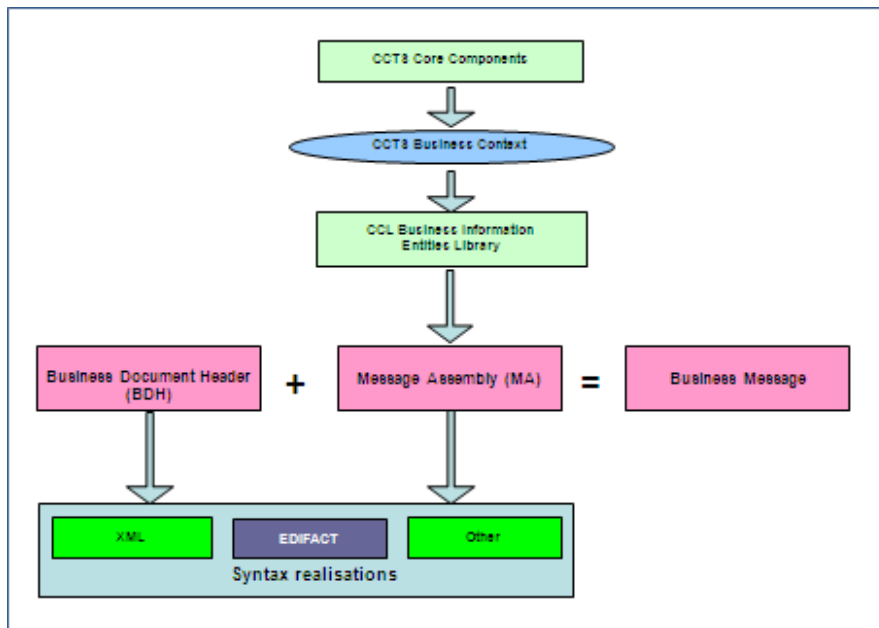
175

176   **Figure 4–1. Business Message Structure**

177   Fig 4.1 describes the components of a CCBDA document assembly and how
178   they combine with a business document header to form a complete business
179   message.

180   A *Business Message* is a container of artifacts that describe the information
181   exchanged in a *Business Interaction*. A *Business Message* consists ofan
182   Exchange Header Envelope (XHE), and one or more *Message Assemblies*
183   (MAs). The XHE includes the identification of the sender and receiver, document
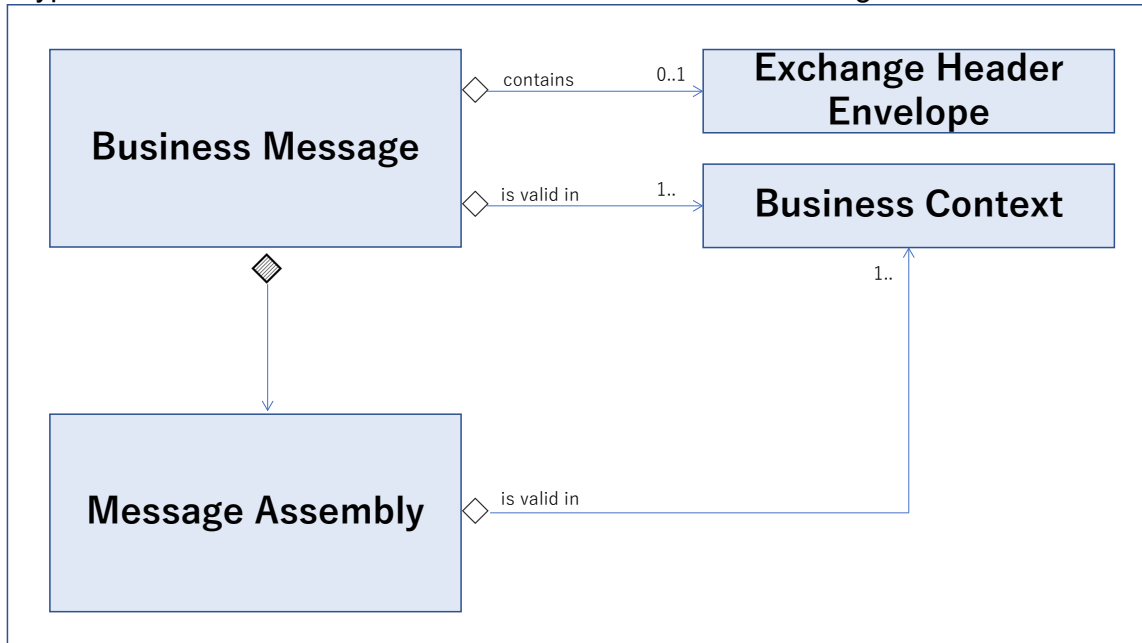184   type etc. An MA is a collection of business data to be exchanged.

Figure 4–2. Message Assembly Conceptual Overview

185   An MA is the logical root of a message structure. An MA consists of one or more
186   ASMAs each of which is an association to an MBIE.

187   MBIEs may have associations to other MBIEs through *Message Association*
188   *Business Information Entities* (ASMBIEs). MBIEs inherit their name, structure
189   and definition from *Aggregate Business Information Entities* (ABIEs). All MBIE
190   properties (ASMBIEs and MBBIEs) inherit their (meta) properties such as
191   *Names, Definitions* and *Property Terms* from their corresponding property
192   (ASBIEs and BBIEs) in the inherited ABIE. The cardinality and values of the
193   ABIE properties (BBIEs and ASBIEs) that the MBIE inherits from the ABIE may
194   be restricted.

```
        ┌─────────────────────────┐
        │   <<InformationEntity>>  │
        │                          │
        │     Message Assembly     │
        └─────────────────────────┘
                    │ ASMA
                    │
        1..*        │      0..*
        ┌─────────────────────────┐
        │   <<InformationEntity>>  │        ASMBIE
        │     Message Business     │
        │    Information Entity    │
        └─────────────────────────┘
                    │
                           0..*
        ┌─────────────────────────┐
        │   Aggregate Business     │        ASBIE
        │    Information Entity    │
        └─────────────────────────┘
```

195
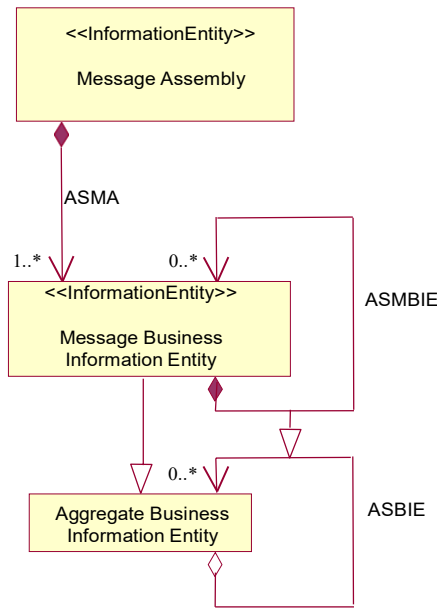
Figure 4–3. CCBDA Conceptual Overview

196

197   *Syntax-neutral Business Messages* are implemented in concrete syntaxes and
198   technical protocols.

199   MAs and ASMAs are not CCTS *Aggregate Business Information Entities* (ABIE)
200   and do not require corresponding *Aggregate Core Components* (ACC). ASMAs
201   are the properties of an MA. *MAs* may be based on more generic *MAs*, much like
202   *Aggregate Business Information Entities* may be based on more generic
203   *Aggregate Business Information Entities*

204   A *Business Message* is exchanged in the course of a *Business Process*. The
205   *Business Context* in which a *Business Message* is valid must be specified. The
206   *Business Context* may be a (specific step in a) *Business Process* plus a business
207   sector and/or a geographical area (amongst others).

208   This section defines the characteristics of an assembled *Business Document as*:

209   • **Atomic**:
210   This means that no constituent element (part), of a *Business Document,*
211   has any business meaning on its own, and cannot, in isolation, change the
212   state of any *Business Entity*. In other words before taking any action
213   whatsoever, the entire *Business Document* must be processed as it is only
214   then that it will be known exactly what actions are required.
215   • **Consistent**:
216   A *Business Document* in its entirety has business meaning that can
217   change the state of *Business Entities*. In other words a *Business*

218   *Document*, that affects the state of a *Business Entity*, uniquely identifies
219   the end state of that *Business Entity*..
220   • **Isolated**:
221   Each *Business Document* has an individual business meaning. A
222   *Business Document* changes the state of *Business Entities* independent of
223   other *Business Documents*. In other words, a Business Document must
224   not have a dependency on any other Business Document.
225   • **Durable**:
226   Once a *Business Document* has been completely processed, the states of
227   all of the *Business Entities* that have been changed, will remain changed
228   until further changes are introduced by a later Business Document. In
229   other words they can only be changed again (or even reverted) by means
230   of a new *Business Document*.
231   • **Idempotent**:
232   Processing multiple copies of the same *Business Document* or processing
233   a Business Document more than once has the same effect on *Business*
234   *Entity* states as processing just one copy. For example, if a *Business*
235   *Document* asks to "increment stock of product A with 1 unit" then
236   processing 10 copies of this same *Business Document* will still only result
237   in one increment of one unit of product A. It is important that a copy be
238   recognized as a copy, and not as another original *Business Document*.

239   A collection of business information to be exchanged between *Business Partners*
240   can form a single business document assembly, multiple business document
241   assemblies or be part of another business document assembly.

242   The characteristics described above may be used to determine whether a
243   collection of information, to be communicated with a *Business Partner*, is 1) a
244   complete *Message Assembly*, or 2) needs to be divided into (distinct / separate)
245   *Message Assemblies*, or 3) needs to be combined with other collections to form a
246   (single) *Message Assembly as follows:*

247   1) Whenever a collection has all of the characteristics listed, then this
248   collection constitutes a complete *Message Assembly*.
249   2) Whenever a part of such a collection has all of the characteristics listed,
250   then this part of the collection constitutes a separate *Message Assembly*.
251   3) Whenever a collection only has these characteristics in combination with
252   one or more other collections, then the collections should be combined to
253   form one *Message Assembly*.

254

5   Business Document Structure

5.1 Message Business Information Entities (MBIEs)

An MA is the logical root of a message structure. An MA consists of one or more ASMAs each of which is an association to an MBIE.

[R01]   Each MA must have one or more ASMAs each of which associates to an MBIE.

An MA may be a subset of another defined MA as long as the context of the subset MA is restricted compared to the parent MA.

[R02]   If an MA is defined as a subset of another MA then the context values of the subset MA must include at least one restricted context value compared to the context values of the parent MA.

[R03]   When MAs are restricted no additional MBIEs may be added.

An MBIE defines which properties of an ABIE are to be included. The content model of an MBIE may be the same as, or may be a subset of, the content model of an ABIE.

An MBIE inherits its name, structure and definition from the ABIE on which it is based.

MBIEs are defined within the scope of an MA and do not exist outside that scope. Identical MBIEs may exist in other MAs but they are not reuses as such.

The structure of an MBIE (e.g. the repetition of its properties i.e. its MBBIEs and ASMBIEs) may be restricted with regard to the underlying ABIE (*Constraints* may also apply; see section 7: Constraints). The following rules apply:

[R04]   If an MBIE is a restricted form of an ABIE then the restrictions must be specified.

[R05]   An MBIE inherits its structure from the ABIE on which it is based. The set of properties of an MBIE may be a subset of the set of properties of the ABIE on which it is based.

[R06]   An MBIE must contain all mandatory properties (BBIEs and/or ASBIEs) of the ABIE on which it is based. These become MBBIEs and/or ASMBIEs.

290 MBIEs have properties which are either basic properties (MBBIEs) or association
291 properties (ASMBIEs).

292 **[R07]   An MBIE must be identified by means of a unique**
293 **   Identifier.**

294 **[R08]   An MBIE may have zero or more associations with**
295 **   other *MBIEs* through *ASMBIEs*.**

296 **[R09]   An MBIE may have zero or more MBBIEs derived from**
297 **   the ABIE on which it is based.**

298 **[R10]   Each property of an MBIE must be derived from the**
299 **   ABIE on which it is based and an MBIE must contain at**
300 **   least one property**

301 **[R11]   An MBIE property may be a restriction of its**
302 **   inherited ABIE property in any or all of the followings**
303 **   ways:**

304 **   a. A used optional property may be made mandatory**

305 **   b. An optional property may be not used**

306 **   c. A used optional or mandatory property may specify a**
307 **   lower number of maximum occurrences but not lower than**
308 **   the minimum occurrences.**

309 **[R12]   The maximum repetition of an MBIE property must**
310 **   not exceed the maximum repetition that is defined for**
311 **   the ABIE.**

312 MBBIEs have Message Data Types (MDTs) that may be a restriction of the
313 underlying qDT.

314 **[R13]   Restrictions may not be applied to uDTs.**

315 **[R14]   If an MDT is a restricted form of a qDT then all**
316 **   restrictions must be specified.**

317 **[R15]   An MDT must not add supplementary components to a**
318 **   Data Type that do not already exist.**

319 **[R16]   An MDT may reduce the maximum cardinality of a**
320 **   supplementary component from 1 to 0. The minimum**
321 **   number of supplementary components is 0.**

322 **[R17]   An MDT may restrict the value domain of a**
323 **   supplementary component or the content component to be**
324 **   more restrictive than the base qDT.**

325 An example of a restriction of a value domain would be a restricted set of codes
326 for a code list.

327   **[R18]   An MDT may restrict the values of the facets of a**
328   **   supplementary component or the content component to be**
329   **   more restrictive than the base qDT.**

330
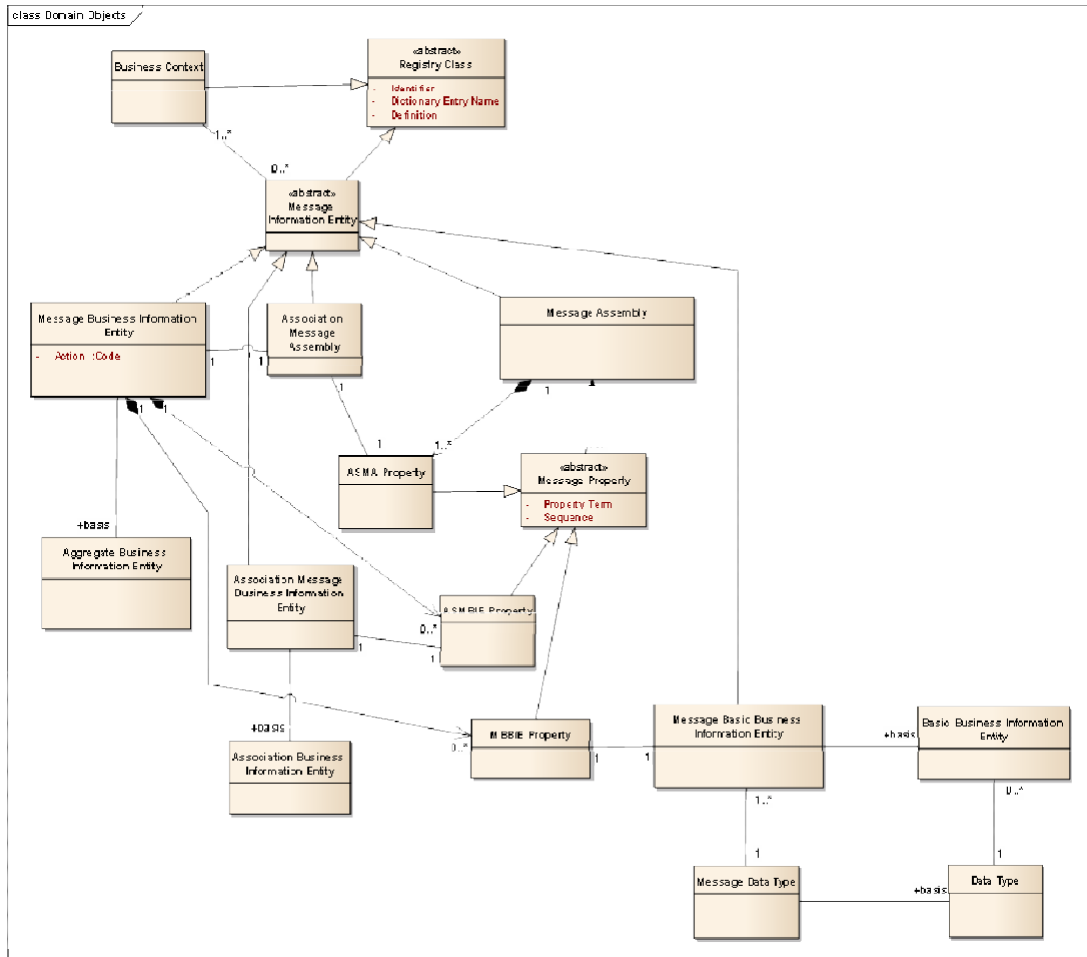
331

332   **Figure 5–1. CCBDA Metamodel**

333   MAs, MBIEs, ASMBIEs MBBIEs, and MDTs within a Business Message that are
334   stored in libraries and registries possess a cohesive set of metadata as defined
335   in CCTS.

336   *MBIEs* have the same set of metadata that ABIEs have.

337   ASMBIEs have the same set of metadata as ASBIEs.

338   An ASMA is an association, without any metadata.

339   Note: The *action code* can define how information defined by the MBIE is to be
340   acted upon (e.g. Create / Refer / Update / Delete).

5.2 Names and Definitions

[R19]   The Dictionary Entry Name of an MA shall consist
of a meaningful Object Class Term and optionally
preceded by an additional Qualifier Term(s) (ref rules
B26 and B27 in CCTS 2.01) to represent its specific
Business Context, followed by a dot, a space
character, and the term Message. The Object Class Term
may consist of more than one word. In all other
respects the naming of the MA should follow the ABIE
naming rules.

[R20] An MBIE inherits its dictionary entry name and definition
from the ABIE on which it is based.

[R21]   The Dictionary Entry Name of an MBIE, ASMBIE,
MBBIE, or MDT shall be unique within its Business
Message.

[R22]   The Dictionary Entry Name of an ASMA follows all
the naming rules for an ASBIE except that the Property
Term is optional.

[R23]   The definition for an ASMA follows the same rules
as for an ASBIE.

5.3 Sequencing

ASMAs and the properties of MBIEs may be assigned an optional *Sequence Number* that may be used in syntax renderings (or in model representations) to present them in some order in the *MA*. However, sequencing does not change the semantics.

[R24]   Sequencing the properties of MABIE should keep the
order of the properties of the derived ABIE. Each MBIE may
be assigned a *Sequence Number* that is used for
presentation of the MA structure or for
representation in a specific syntax.

5.4 Business Document Header

A Business Message contains a *Business Document Header*. A *Business Document Header* MUST contain the attributes as defined by the UN/CEFACT Business Document Header (BDH).

6   Context

A *Business Document* is exchanged in a certain *Context*. CCTS defines eight context categories or context dimensions: Business Process, Product Classification, Industry Classification, Geopolitical, Official Constraints, Business Process Role, Supporting Role and System Capabilities.

The structure of a *Business Document* is dependent on the specific step in the *Business Process* or *Business Process Activity* where the *Business Document* is used. That step further narrows the context.

[R25]   The context of the MA must apply to all of the objects within the MA structure.

7   Constraints

Each *Message Business Information Entity* or *Message Assembly* may have *Constraints* associated with it.

Constraints are used to restrict the content model or business process to satisfy specific business requirements where the constraint is the formal expression of the requirements.

Constraints may affect repetition and cardinality, element values, or a mix (e.g. "if the Delivery Term="FOB", a Delivery Term Location must be present").

A constraint can be structured or unstructured. An unstructured constraint will be expressed as free form text. A structured constraint is a constraint that is expressed in a formal constraint language such as the UML Object Constraint Language (OCL) or Object Management Group (OMG) Semantics of Business Vocabulary and Usage Rules (SBVR).

Constraint condition types may also be specified such as invariant, pre-condition or post-condition. The constraint condition type value is taken from a constraint type code list.

[R26]   If an MBIE contains any constraints then each constraint must contain one or more of the following:

- **the text of the constraint**
- **a reference identifier to a constraint defined in an external list of constraints if applicable**
- **a code defining the type of the constraint condition**
- **in the case of structured constraints, a code indicating the constraint language in which the constraint is expressed**

410   **[R27]**   An unstructured constraint shall have or refer to
411   a free form text expression that fully details the
412   business requirements that it is addressing.
413

417   **[R28]**   A structured constraint shall have or refer to a
418   formal constraint
419   language expression.
420

8   Definition of Terms

Aggregate Business Information Entity (ABIE) – A collection of related pieces of business information that together convey a distinct business meaning in a specific business context. Expressed in modelling terms, it is the representation of an object class, in a specific business context.

Association Business Information Entity (ASBIE) – A business information entity that represents a complex business characteristic of a specific object class in a specific business context. It has a unique business semantic definition. An Association Business Information Entity represents an Association Business Information Entity property and is therefore associated to an Aggregate Business Information Entity, which describes its structure. An Association Business Information Entity is derived from an Association Core Component.

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

Association Core Component (ASCC) – A Core Component which constitutes a complex business characteristic of a specific Aggregate Core Component that represents an Object Class. It has a unique Business Semantic definition. An Association Core Component represents an Association Core Component Property and is associated to an Aggregate Core Component, which describes its structure.

Association Message Business Information Entity – An association between two Message Business Information Entities. An Association Message Business Information Entity is based on an Association Business Information Entity.

Attribute – A named value or relationship that exists for some or all instances of some entity and is directly associated with that instance.

Business Partner - A business partner is an organization type, an organizational unit type or a person type that participates in a business process.

Business Context – The formal description of a specific business circumstance as identified by the values of a set of context categories, allowing different business circumstances to be uniquely distinguished.

Business Information Entity (BIE) – A piece of business data or a group of pieces of business data with a unique business semantic definition. A business information entity can be a Basic Business Information Entity (BBIE), an Association Business Information Entity (ASBIE), or an Aggregate Business Information Entity (ABIE).

458   **Business Interaction** – Exchange of Business Messages between Business
459   Partners in a Business Context. Business interaction involves communication by
460   one Business Partner to another of a change to at least one BIE of a Business
461   Document.
462   **Business Document** – A collection of information that is exchanged in a step of
463   a Business Process. A Business Document updates information on Business
464   Entities, such as Products, Contracts, Locations, etc. Exchanging a Business
465   Document synchronizes the knowledge on the states of those Business Entities
466   among the Business Partners involved.
467   **Business Message –** A business message (also known as an information
468   envelope) has exactly one business document header (BDH) which serves for
469   identification purposes of technical sender and receiver, document type etc. A
470   business document header is defined in the Business Document Header
471   specification of UN/CEFACT. The body of a business message consists of
472   exactly one element, which is of type message assembly (MA). This single
473   message assembly serves as the root element of a business document definition
474   and is connected to the information envelope using a standard UML aggregation.
475   Message assemblies are used to aggregate different aggregate business
476   information entities (ABIE) to a specific business document. Association
477   message assemblies (ASMA) are used to connect different message assemblies
478   to each other and to connect aggregate business information entities to message
479   assemblies. ABIEs, MAs, and ASMAs are part of the UML Profile for Core
480   Components.
481   **Business Process** – The business process as described using the UN/CEFACT
482   Catalogue of Common business processes.
483   **Business Semantic(s)** – A precise meaning of a concept from a business
484   perspective.
485   **Classification Scheme** – This is an officially supported scheme to describe a
486   given context category.
487   **Constraint –** a constraint is one or more conditions expressed as a business rule
488   used to restrict a content model or business process to satisfy a specific
489   business requirement where the constraint is the formal expression of the
490   requirement.
491   **Context** – Defines the circumstances in which a business process may be used.
492   This is specified by a set of context categories known as business context.
493   **Context Category** – A group of one or more related values used to express a
494   characteristic of a business circumstance.
495   **Core Component Message Library (CCML) –** The library of all Message
496   Assemblies and their constituent parts published by an organization, such as the
497   UN/CEFACT CCML.
498   **Definition** – The unique semantic meaning of a concept, business document,
499   core component, business information entity, business context or data type.
500   **Dictionary –** A collection of Dictionary Entry Names for CCTS conformant
501   artefacts for a specific library.
502   **Document Assembly** – (1) The process whereby Business Information Entities
503   are assembled into a usable document for exchanging business information. (2)

504 A grouping of Message Business Information Entities which does not change the
505 semantics of the MBIEs.
506 **Information Envelope (also known as Business Message)** - An information
507 envelope has exactly one business document header (BDH) which serves for
508 identification purposes of technical sender and receiver, document type etc. A
509 business document header is defined in the Business Document Header
510 specification of UN/CEFACT. The body of an information envelope consists of
511 exactly one element, which is of type message assembly (MA). This single
512 message assembly serves as the root element of a business document definition
513 and is connected to the information envelope using a standard UML aggregation.
514 Message assemblies are used to aggregate different aggregate business
515 information entities (ABIE) to a specific business document. Association
516 message assemblies (ASMA) are used to connect different message assemblies
517 to each other and to connect aggregate business information entities to message
518 assemblies. ABIEs, MAs, and ASMAs are part of the UML Profile for Core
519 Component
520 **Message Business Information Entity** – A part of a Business Document that
521 inherits its definition and structure from an ABIE to represent information to be
522 updated.
523 **Naming Convention** – The set of rules that together comprise how the
524 dictionary entry name for artefacts is constructed.
525 **Object Class** – The logical data grouping (in a logical data model) to which a
526 data element belongs.
527 **Object Class Term** – A component of the name of a core component or
528 business information entity which represents the object class to which it belongs.
529 **Property Term** – A semantically meaningful name for the characteristic of the
530 Object Class that is represented by the core component property. It shall serve
531 as basis for the DEN of the basic and Association Core Components that
532 represents this core component property.
533 **Qualifier Term** – A word or group of words that help define and differentiate an
534 item (e.g. a business information entity or a business data type) from its
535 associated items (e.g. from a core component, a core data type, another
536 business information entity or another business data type).
537 **Registry –** An information system that manages and references artifacts that are
538 stored in a repository. The term registry implies a combination of
539 registry/repository.
540 **Registry class** – The formal definition of all the common information necessary
541 to be recorded in the registry by a registry artefact – core component, a business
542 information entity, a data type or a business context.
543 **Representation Term** – The type of valid values for a Basic Core Component or
544 Basic Business Information Entity.
545 **Sequence Number –** A number identifying the sequence of an item within a
546 collection of related items.
547 **Unique Identifier** – The identifier that references a registry class instance in a
548 universally unique and unambiguous way.

549 **Usage Rules** – Usage rules describe a constraint that describes specific
550 conditions that are applicable to a component in the model.
551 **Version** – An indication of the evolution over time of an instance of a core
552 component, data type, business context, or business information entity.
553 **Web Service** - Web services describes a standardized way of integrating Web-
554 based applications using the XML, SOAP, WSDL and UDDI open standards over
555 an Internet protocol backbone.
556

557 ## Intellectual Property Disclaimer

558 ECE draws attention to the possibility that the practice or implementation of its
559 outputs (which include but are not limited to Recommendations, norms,
560 standards, guidelines and technical specifications) may involve the use of a
561 claimed intellectual property right.

562 Each output is based on the contributions of participants in the UN/CEFACT
563 process, who have agreed to waive enforcement property rights
564 pursuant to the UN/CEFACT ECE/TRADE/C/CEFACT/2010/20/Rev.2
565 available at http://www.unece.org/cefact/ cf_docs.html or from the ECE
566 secretariat).

567 ECE takes no position concerning the evidence, validity or applicability of any
568 claimed intellectual property right or any other right that might be claimed by any
569 third parties related to the implementation of its outputs. ECE makes no
570 representation that it has made any investigation or effort to evaluate any such
571 rights.

572 Implementers of UN/CEFACT outputs are cautioned that any third-party
573 intellectual property rights claims related to their use of a UN/CEFACT output will
574 be their responsibility and are urged to ensure that their use of UN/CEFACT
575 outputs does not infringe on an intellectual property right of a third party.
576 ECE does not accept any liability for any possible infringement of a claimed
577 intellectual property right or any other right that might be claimed to relate to the
578 implementation of any of its outputs.

579

580   # Annex A : Non-normative - for information only

581   ## Relation between CCBDA and UMM

582   UMM defines how to analyze an inter-organizational business process to design
583   the Process Choreography. One of the views, defined in the UMM, is the
584   Business Information View (BIV). The UN/CEFACT CCBDA Technical
585   Specification, in conjunction with the UN/CEFACT CCTS, may be used to model
586   that BIV.

587   UMM also defines a Business Entity View (BEV). In the BEV, the *Business*
588   *Entities* are identified that are affected by the process. *Business Entities* are
589   represented in the Core Components Library (CCL) as *Aggregate Business*
590   *Information Entities* (ABIEs). In the course of the business process the
591   information on *Business Entities* changes and is updated by means of business
592   documents, defined as *Message Assembles* (MA). An MA consists of *Message*
593   *Business Information Entities* (MBIEs).

594   The choreography may define the states and the lifecycles of the *Business*
595   *Entities* and the mechanism to synchronize those states between business
596   partners (who fulfill Authorized Roles). Business Entity State synchronization
597   means that some business entities may be instantiated in a business transaction,
598   while others may only be changed or deleted, e.g., in an Order Confirmation
599   document, order lines might be deleted or changed, but not added. This can be
600   defined in the Action code, which is an optional attribute of an MBIE.

601   Business Context, as defined in the UN/CEFACT Context Methodology (UCM),
602   defines what information needs to be exchanged to synchronize the *Business*
603   *Entity* states. Context is applied in two steps. First, context is applied to Core
604   Components to turn these CC's into *Business Information Entities* (BIE's) that
605   reflect a particular business environment (e.g., the ordering of office supplies).
606   The step from CC to BIE is described in UN/CEFACT CCTS.

607   Second, BIE's are customized to the specific requirements of a particular
608   information exchange in a particular business process (e.g., office supplies order
609   confirmation). This step is described in the UN/CEFACT CCBDA Technical
610   Specification.

611   The second step of context application could be realized by adding constraints to
612   the business document root (the MA). These constraints may be stated in a
613   formal constraint language such as the OMG Object Constraint Language. In
614   many cases cardinalities of BIE's are affected by this second step (e.g., in an
615   order confirmation, order lines may only contain an identifier and no product
616   specification), and it is desirable to present the restricted cardinalities graphically
617   instead of as rule statements. Therefore, instead of using ABIE's directly,
618   *Message Business Information Entities* (MBIE's) are used. MBIE's are based on

619   ABIE's in the same way ABIE's are based on *Aggregate Core Components*
620   (ACC's). The cardinalities of MBIE's may be restricted with regard to the ABIE
621   they are based on and they may graphically be presented in a UML Class
622   diagram.

623