



Creating A Single Global Electronic Market

Automated Negotiation of Collaboration- Protocol Agreements Specification Version 0.10

OASIS ebXML Collaboration Protocol Profile and Agreement Technical Committee

Date TBD

Status of this Document

This document specifies an ebXML SPECIFICATION for the eBusiness community.

Distribution of this document is unlimited.

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

This version:

URL TBD

Errata for this version:

URL TBD

Previous version:

URL TBD

30 **1 Automated Negotiation Sub Team Members**

- 31 Martin Sachs, Cyclone Commerce, team lead
- 32 Keith Babo, Sun Microsystems
- 33 Jamie Clark, OASIS staff
- 34 Brian S. Hayes, Extended Thinking (individual member)
- 35 Neelakantan Kartha, Sterling Commerce
- 36 Kevin Liu, SAP
- 37 Heiko Ludwig, IBM
- 38 Monica Martin, Sun Microsystems
- 39 Dale Moberg, Cyclone Commerce
- 40 Himagiri Mukkamala, Sybase
- 41 Peter Ogden, Cyclone Commerce
- 42 Yukinori Saito, ECOM
- 43 Nikola Stojanovic, Encoda Systems
- 44 Jean Zheng, Vitria

45 **2 Table of Contents**

46	Status of this Document.....	1
47	1 Automated Negotiation Sub Team Members	2
48	2 Table of Contents.....	3
49	3 Introduction	5
50	3.1 Summary of Contents of Document	5
51	3.2 Definition and Scope of this Specification	5
52	3.3 Document Conventions	5
53	3.4 Versioning of the Specification, Schema, and Related Documents.....	6
54	3.5 Definitions	7
55	3.6 Audience.....	7
56	3.7 Assumptions	7
57	3.8 Related Documents.....	7
58	3.9 Acknowledgments	7
59	4 Design Objectives.....	8
60	5 System Overview.....	9
61	5.1 Main Components of CPA Negotiation.....	9
62	5.2 Overview of CPA Negotiation.....	11
63	5.3 Pre-Conditions for Negotiation.....	13
64	5.4 CPP and NDD Formation and Editing.....	14
65	5.5 Discovery of CPPs and CPA Templates.....	14
66	5.6 Negotiation through an Intermediary.....	14
67	6 CPA Template	16
68	6.1 CPA Template and Draft CPA.....	16
69	6.2 Advantages of Starting Negotiation with a CPA Template	16
70	6.3 CPA Template composition.....	17
71	7 CPP and CPA Template Content.....	18
72	7.1 Validation of CPP and CPA Template.....	18
73	7.2 Preference Order.....	18
74	7.3 Conflicts between two Parties' Preferences.....	18
75	7.4 CPA Period of Validity.....	18
76	8 Negotiation CPA (NCPA)	19
77	8.1 Document Exchange.....	19
78	8.2 Transport.....	20
79	8.3 Packaging	20
80	8.4 Security.....	20
81	8.5 Explanation of NCPA Example.....	20
82	9 Negotiability of CPA Elements and Attributes.....	21
83	9.1 Enumerations.....	21
84	9.2 CollaborationRole element and its child elements.....	21
85	9.3 Elements or Attributes whose Cardinality Includes Zero	22
86	9.4 Values.....	22
87	9.5 Items that are Referred to	22
88	9.6 Transport Endpoints	22
89	9.7 Security.....	22
90	9.7.1 Trust Anchor and Certificate Alignment	23
91	9.8 Discussion of Various Elements and Attributes	25
92	10 Negotiation Descriptor Document	28
93	10.1 Use of NDD.....	28
94	10.2 General Principles of Contents of NDD	28
95	10.3 Composition of an NDD for a CPA Template.....	29
96	10.4 Explanation of Contents of NDD.....	30
97	11 Negotiation Messages.....	32

98	11.1 Negotiation Message Structure.....	32
99	11.1.1 NegotiationMessage element.....	33
100	11.1.2 NCPA element.....	34
101	11.1.3 CPATemplateId element.....	34
102	11.1.4 cpa:SecurityDetails element.....	34
103	11.1.5 InitiatingParty element.....	34
104	11.1.6 RespondingParty element.....	35
105	11.1.7 BPSSBusinessDocumentName element.....	35
106	11.1.8 BusinessDocuments element.....	36
107	11.1.9 NegotiationContent element.....	36
108	11.1.10 ResponseToURL element.....	41
109	11.1.11 Offer-CounterOffer-Acceptance-Time.....	41
110	11.2 CPA ID, Negotiation Dialog ID, Unique Business Message ID, and InResponseTo.....	41
111	11.3 Offer and Counter Offer.....	41
112	11.4 CPA Offer Rejected.....	41
113	11.5 CPA Offer accepted.....	42
114	12 Negotiation Protocol.....	44
115	12.1 General Principles of Negotiation Protocol.....	44
116	12.2 CPA Identifier.....	44
117	12.3 Negotiation-Dialog Identifier.....	44
118	12.4 Offer Identifier.....	44
119	12.5 Negotiation Status.....	45
120	12.6 ebXML Conversation.....	45
121	12.7 Negotiation CPA.....	45
122	12.8 Initial Offer.....	45
123	12.9 Simultaneous Initial Offers.....	46
124	12.10 Offer and Counter Offer.....	46
125	12.10.1 Responses to Offer and Counter Offer.....	47
126	12.10.2 Offer-Counter Offer Acceptance Time.....	47
127	12.11 Conclusion of Negotiation.....	47
128	12.12 Signing the CPA.....	48
129	12.13 BPSS Instance Document for Automated Negotiation.....	49
130	12.13.1 Offer-Counter-Offer Choreography.....	50
131	12.13.2 Final CPA exchange.....	51
132	12.13.3 Negotiation Business Signals.....	53
133	12.13.4 State Diagrams.....	53
134	13 Negotiation Algorithm.....	56
135	13.1 CPPs and NDDs.....	56
136	14 References.....	57
137	15 Conformance.....	58
138	15.1 NDD and Negotiation Messages.....	58
139	15.2 NCPA Instance Document.....	58
140	15.3 Negotiation BPSS Instance Document.....	58
141	15.4 Negotiation Business Signals.....	59
142	16 Disclaimer.....	60
143	17 Contact Information.....	61
144	Appendix A XML Schema for Negotiation Descriptor Document (Normative).....	63
145	Appendix B XML Schema for Negotiation Messages (Normative).....	67
146	Appendix C Negotiation CPA Example (Non-Normative).....	70
147	Appendix D Negotiation BPSS Instance Document (Normative).....	83
148	Appendix E Instance Documents for Business Signals.....	90
149	Appendix F Example of NDD Instance Document (Non-Normative).....	91
150	Appendix G Examples of Negotiation-Message Instance Documents (Non-Normative).....	95
151	Appendix H Glossary of Terms.....	97

152 **3 Introduction**

153 **3.1 Summary of Contents of Document**

154 This document contains a specification for automatically negotiating the contents of an ebXML
155 *Collaboration Protocol Agreement (CPA)*[ebCPP]. This specification is a component of the suite
156 of ebXML specifications.

157 This document is organized as follows:

- 158 • Section 3 introduces the specification and discusses various procedural matters
- 159 • Section 4 summarizes the design objectives.
- 160 • Section 5 is a system-level overview.
- 161 • Section 6 discusses the *CPA Template*.
- 162 • Section 7 discusses content of *CPPs* and *CPA Templates* with respect to negotiation.
- 163 • Section 8 gives the rules for constructing a *Negotiation CPA*, the *CPA* that governs the
164 *Negotiation Protocol*.
- 165 • Section 9 discusses negotiability of elements and attributes in the *CPA*.
- 166 • Section 10 defines and discusses the *Negotiation Descriptor Document (NDD)* that is used to
167 describe offers and counter offers.
- 168 • Section 11 defines the contents of the *Negotiation Messages*.
- 169 • Section 12 defines the *Negotiation Protocol* including the ebXML *Business Process*
170 *Specification Schema*[ebBPSS] instance *Document* that is used to describe the Negotiation
171 Transactions and their choreography.
- 172 • Section 13 discusses negotiation algorithms.
- 173 • The appendices include XML Schemas for the *NDD* and *Negotiation Messages*, the
174 negotiation BPSS instance *Document*, examples of an *NDD* instance *Document* and
175 negotiation *Message* instance *Documents*, non-normative aspects of *CPA* composition, and a
176 glossary of terms.
- 177

178 **3.2 Definition and Scope of this Specification**

179 The goal of this specification is to define a means of automatically negotiating the contents of a
180 *CPA*. The focus is on negotiating both long-term partner relationships and spontaneous (perhaps
181 for a single business exchange) relationships. Automated negotiation of *CPAs* is a critical
182 element of spontaneous e-commerce since it will enable business to be conducted with minimal
183 delay, as soon as two potential trading partners discover each other. Automated negotiation also
184 will enhance the ability of an enterprise to maintain large numbers of partner relationships. It will
185 reduce the need for manual intervention in maintaining those relationships, thereby simplifying
186 life-cycle management of the relationships.

187
188 This specification defines the rules for automated negotiation of *CPAs*. It defines the *Negotiation*
189 *Protocol* and the contents of the *Documents* that are part of the *Negotiation Protocol*.

190 **3.3 Document Conventions**

191 Terms in *Italics* are defined in Appendix H or in the glossary of the CPPA specification[ebCPP].
192 Terms listed in ***Bold Italics*** represent the element and/or attribute content of the XML *CPP*,

193 *CPA*, or related definitions.

194

195 In this specification, the term “item”, when used in the context of an *NDD* or counter offer
196 *Message* denotes an element, attribute, or subtree that is negotiable.

197

198 The term “BPSS instance *Document*” refers to an XML document that is an instance *Document*
199 of the XML schema of the *Business Process Specification Schema*[ebBPSS] ebXML
200 specification.

201

202 In this specification, indented paragraphs beginning with "NOTE:" provide non-normative
203 explanations or suggestions that are not mandated by the specification.

204

205 References to external documents are represented with BLOCK text enclosed in brackets, e.g.
206 [RFC2396]. The references are listed in Section 14.

207

208 The keywords MUST, MUST NOT, REQUIRED, SHALL, SHALL NOT, SHOULD, SHOULD
209 NOT, RECOMMENDED, MAY, and OPTIONAL, when they appear in this document, are to be
210 interpreted as described in [RFC 2119].

211

212 NOTE: Vendors SHOULD carefully consider support of elements with cardinalities (0 or
213 1) or (0 or more). Support of such an element means that the element is processed
214 appropriately for its defined function and not just recognized and ignored. A given *Party*
215 might use these elements in some *CPPs*, *CPAs*, negotiation *Messages*, or *NDDs* and not in
216 others. Some of these elements define parameters or operating modes and SHOULD be
217 implemented by all vendors. It might be appropriate to implement elective elements that
218 represent major run-time functions, such as various alternative communication protocols or
219 security functions, by means of plug-ins so that a given *Party* MAY acquire only the
220 needed functions rather than having to install all of them.

221

222 By convention, values of [XML] attributes are generally enclosed in quotation marks; however
223 those quotation marks are not part of the values themselves.

224

224 **3.4 Versioning of the Specification, Schema, and Related Documents**

225

225 Whenever this specification is modified, it SHALL be given a new version number.

226

227 It is anticipated that during the review period, errors and inconsistencies in the specification and
228 in the schemas may be detected and have to be corrected. All known errors in the specification as
229 well as necessary changes to the schema will be summarized in an errata page found at

230

231 TBD [http://www.oasis-](http://www.oasis-open.org/committees/ebxml-cppa-negotiation/documents/cppa-negotiation-Errata.shtml)
232 [open.org/committees/ebxml-cppa-negotiation/documents/cppa-](http://www.oasis-open.org/committees/ebxml-cppa-negotiation/documents/cppa-negotiation-Errata.shtml)
233 [negotiation-Errata.shtml](http://www.oasis-open.org/committees/ebxml-cppa-negotiation/documents/cppa-negotiation-Errata.shtml)

234

234 The specification when approved SHALL carry a version number of “1_0”. At that time, the
235 schemas SHALL have a version number of “1_0a” and the suffix letter after “1_0” will be
236 advanced as necessary when bug fixes to the schemas have to be introduced. Such versions of
237 the schemas SHALL be found at the following URLs:

238

Automated Negotiation Specification

Copyright © OASIS, 2003. All Rights Reserved

Negotiation.spec.01Nov03.doc

Deleted: <#>Versioning of the Specification, Schema, and Related Documents¶

Whenever this specification is modified, it SHALL be given a new version number. ¶

¶ It is anticipated that during the review period, errors and inconsistencies in the specification and in the schemas may be detected and have to be corrected. All known errors in the specification as well as necessary changes to the schema will be summarized in an errata page found at¶

¶ **TBD** <http://www.oasis-open.org/committees/ebxml-cppa-negotiation/documents/cppa-negotiation-Errata.shtml>¶

¶ The specification when approved SHALL carry a version number of “1_0”. At that time, the schemas SHALL have a version number of “1_0a” and the suffix letter after “1_0” will be advanced as necessary when bug fixes to the schemas have to be introduced. Such versions of the schemas SHALL be found under the directory¶

¶ **TBD** <http://www.oasis-open.org/committees/ebxml-cppa/schema/NegotiationMessage.xsd>¶

¶ **TBD** <http://www.oasis-open.org/committees/ebxml-cppa/schema/NDD.xsd>¶

¶ In addition, the latest version of the schemas SHALL always be found at¶

¶ **TBD** http://www.oasis-open.org/schema/2_0.xsd¶

¶ These URIs are also the namespace URIs used for the schemas defined by this specification. The latest versions will then be directly resolvable from the namespace URI.¶

¶ **TBD SHOULD WE HAVE A VERSION ATTRIBUTE IN THE SCHEMAS?¶**

The value of the version attribute of the Schema element in a given version of the schemas SHALL be equal to the version of the schema.¶

Formatted: Bullets and Numbering

239 [TBD http://www.oasis-open.org/committees/ebxml-cppa/schema/NegotiationMessage.xsd](http://www.oasis-open.org/committees/ebxml-cppa/schema/NegotiationMessage.xsd)

240

241 [TBD http://www.oasis-open.org/committees/ebxml-cppa/schema/NDD.xsd](http://www.oasis-open.org/committees/ebxml-cppa/schema/NDD.xsd)

242

243 [In addition, the latest version of the schemas SHALL always be found at](#)

244

245 [TBD http://www.oasis-open.org/schema/2_0.xsd](http://www.oasis-open.org/schema/2_0.xsd)

246

247 [These URIs are also the namespace URIs used for the schemas defined by this specification. The latest versions will then be directly resolvable from the namespace URI.](#)

248

249 [***SHOULD WE HAVE A VERSION ATTRIBUTE IN THE SCHEMAS?***](#)

250 [The value of the version attribute of the Schema element in a given version of the schemas](#)

251 [SHALL be equal to the version of the schema.](#)

252

253 **3.5 Definitions**

254 Technical terms related to the subject of this specification are defined in Appendix H.

255 Technical terms related to *Collaboration Protocol Profiles* and *Agreements* and to the overall
256 vocabulary of ebXML are defined in [ebCPP].

257 **3.6 Audience**

258 One target audience for this specification is implementers of ebXML services and other
259 designers and developers of middleware and application software that is to be used for
260 conducting electronic *Business*. Another target audience is the people in each enterprise who are
261 responsible for creating *CPPs* and *CPAs*.

262 **3.7 Assumptions**

263 It is expected that the reader has an understanding of XML and is familiar with the ebXML
264 CPPA specification[ebCPP].

265 **3.8 Related Documents**

266 Related documents include ebXML specifications on the following topics:

- 267 • ebXML Collaboration Protocol Profile and Agreement Specification[ebCPP]
- 268 • ebXML Business Process Specification Schema[ebBPSS]
- 269 • ebXML Message Service Specification[ebMS]

270

271 See Section 14 for the complete list of references.

272 **3.9 Acknowledgments**

- 273 • To Duane Nickull, XML Global, for his ebXML Automatic CPA Negotiation proposal, Feb,
274 14, 2001.

- 275 • To The ebXML *Business Process* Team, for its automated contract negotiation pattern in
276 [bpPATT].

277 **4 Design Objectives**

278 This specification defines the protocol, *Messages*, and *Business Documents* associated with
 279 automatically negotiating the contents of a *CPA*. It does NOT define negotiation algorithms in
 280 detail. The negotiation algorithm is part of the private process at each *Party* and MAY embody
 281 private or proprietary strategies. This specification does define the rules that ensure
 282 interoperability between two *Parties'* negotiation algorithms.

283
 284 Following are the objectives for the design of this specification.

- 285 • The design is based on negotiating the contents of a *CPA* starting with a *CPA Template* (draft
 286 *CPA*) that one prospective trading partner sends to the other as an initial offer. See Section
 287 6.2 for a discussion of *CPA Template* and draft *CPA*. A *CPA Template* contains elements and
 288 attributes that need to be negotiated with a prospective trading partner. A *Party* can publish a
 289 *CPA Template* in a *Registry* or can create one from its *CPP* and the prospective trading
 290 partner's *CPP*.
- 291 • The specification defines the *Negotiation Protocol* transactions and choreography by means
 292 of an ebXML *Business Process Specification Schema*[ebBPSS] instance *Document*.
- 293 • The *Negotiation Protocol* is governed by a *Negotiation CPA* (*NCPA*). The *NCPA* is a
 294 standard ebXML *CPA* that defines a minimal set of function that all *Parties* can be expected
 295 to support without *Parties* having to negotiate the *NCPA* before negotiating the *CPA* for their
 296 *Business Collaboration*.
- 297 • Avoid requiring changes to the CPPA and BPSS specifications, at least for version 1 of the
 298 negotiation spec.
- 299 • Use deterministic algorithms
- 300 • The *Negotiation Process* SHOULD converge rapidly.
 - 301 ♦ The process SHOULD either succeed or fail.
 - 302 ♦ The process SHOULD invoke human intervention on failure
 - 303 ♦ The design SHOULD avoid deadlock such as iterative loops that don't advance the state
 304 of the negotiation. An example is reiteration over the same offer or counter offer that was
 305 previously rejected by either or both parties.
 - 306 ▪ The specification SHOULD state rules that avoid such iterative loops even if it is
 307 decided that automatic detection of loops is out of scope for version 1.
- 308 • It MUST be absolutely clear at any point in the negotiation which *Party* (i.e., only one *Party*)
 309 has the initiative to send the next request (counter offer).
 - 310 ♦ The design SHOULD avoid race conditions in which both parties simultaneously send an
 311 a counter offer. The choreography should make this an error condition.

312
 313
 314 NOTE: It is probably not possible to avoid or detect the case where two *Parties* send
 315 each other initial offers. This condition should be recognized by people.

- 316 • The design SHOULD minimize the amount of state that has to be saved.
- 317 • Offer rejection semantics SHOULD be strong; rejection SHOULD not be a tactical
 318 maneuver.

320 **5 System Overview**

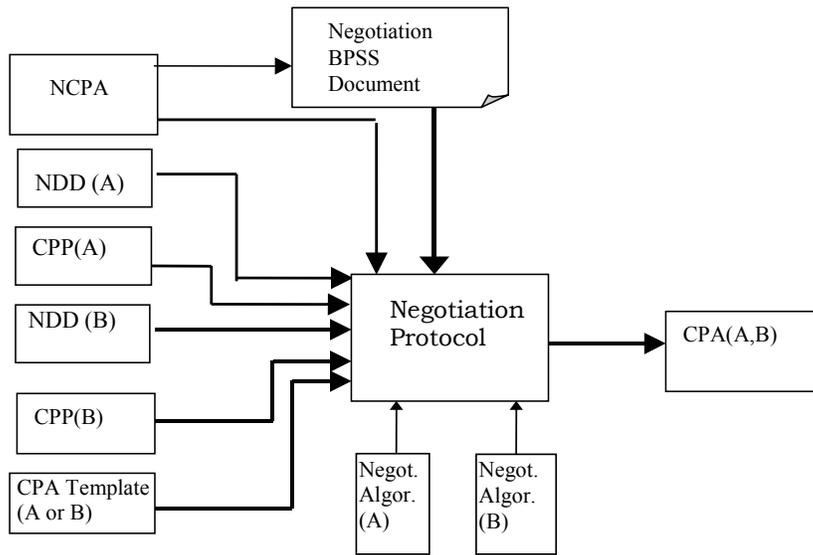
321 The *CPA Negotiation Protocol* begins when one *Party* makes an initial offer to a second *Party*.
 322 The initial offer consists of a *CPA Template* and a *Negotiation Descriptor Document (NDD)* that
 323 describes what is negotiable in the *CPA Template*. [See the discussion in Section 6.2.](#)

324
 325 In the *CPA Negotiation Protocol*, a *CPA Template* is verified as suitable for both *Parties* and
 326 modified until a suitable *CPA* is constructed. It might also be discovered that agreement cannot
 327 be reached until one *Party* (or both) acquires additional software capabilities. The term “*CPA*
 328 *Template*” was chosen to emphasize its use as the starting point for *CPA* negotiation. In general,
 329 a *CPA Template* constitutes a proposal about an overall binding of a *Business Process* to a
 330 delivery agreement with some items left open; negotiation is then used to arrive at detailed
 331 values for the open items in order to achieve a final agreement. The *NDD* identifies what items
 332 have to be negotiated and defines ranges or sets of acceptable values for those items.

334 **5.1 Main Components of CPA Negotiation**

335 Figure 1 illustrates the main components of *CPA* negotiation.

336



337
 338

339 **Figure 1. Components of CPA Negotiation**

340 The following entities are shown in the figure:

- 341 • NCPA: The Negotiation CPA controls the Negotiation Protocol.

- 344 • Negotiation BPSS Instance *Document*: An ebXML *Business Process Specification*
 345 *Schema*[ebBPSS] instance *Document* that is used to define the negotiation collaborative
 346 protocol. This BPSS instance *Document* is referenced from an *NCPA*.
 347 • *CPP*: Parties A and B publish their *CPPs* in an ebXML *Registry*[ebRS] or otherwise
 348 exchange them when they discover each other.
 349 • *CPA Template*: A *CPA* in which some items remained to be filled in by one or the other
 350 *Party*, or negotiated between them.
 351 • *NDD*: The *Negotiation Descriptor Document*, a *Document* associated with a *CPP* or a *CPA*
 352 *Template* that defines what is negotiable, ranges of numeric values, etc. The *NDD* is used in
 353 the *Negotiation Protocol*. [Each Party might have a separate NDD that refers to its own CPP](#)
 354 [and represents its own viewpoint of what is negotiable, acceptable ranges of values, etc.](#)
 355 • *Negotiation Messages*: The *Messages* used to exchange offer and counter-offer information
 356 between negotiating *Parties*.
 357 • *Negotiation Protocol*: The collaborative protocol that produces a negotiated *CPA*. Although
 358 shown as a single box in this figure, the *Negotiation Protocol* is executed between the two
 359 *Parties* or between each *Party* and an intermediary.
 360 • Negotiation algorithm: The negotiation algorithm is the private process at each negotiating
 361 *Party* that implements that *Party's* private negotiation strategy. Note that the *Negotiation*
 362 *Protocol* is distinct from the negotiation algorithm. The former is the public protocol,
 363 captured by the BPSS instance *Document*. Each *Party* uses its negotiation algorithm, in
 364 conjunction with the *CPA Template*, *CPPs* and the *NDD*, to arrive at an offer or counter offer
 365 in the *Negotiation Protocol*. The negotiation algorithm is out of scope for version 1 of this
 366 specification. See Section 13 for additional discussion.

367 Two *Parties* can negotiate a *CPA* as follows. First, they publish their *CPPs* in an ebXML
 368 *Registry*, or similar *Registry*, so that potential trading partners can discover them. A *Party* MAY
 369 publish an *NDD* along with the *CPP*. This *NDD* describes what is negotiable in the *CPP*.
 370

371 When *Party B* discovers *Party A* as a potential trading partner, *Party B* composes a *CPA*
 372 *Template* from its own *CPP* and *Party A's* *CPP*. If *Party A* published an *NDD* along with its
 373 *CPP*, *Party B* MAY use the information in *Party A's* *NDD* along with its own *NDD* in
 374 composing the *NDD* for the initial offer.
 375

376 Alternatively, *Party A* MAY publish a *CPA Template* and *NDD*. In that case, *Party B* creates an
 377 initial offer by filling in basic information about itself (e.g. its *Party* ID and transport endpoint
 378 address). It then creates a new *NDD* by adding its own negotiability information to that from
 379 *Party A's* *NDD*.
 380

381 In order to negotiate, *Parties A* and *B* MUST obey the rules defined by an *NCPA*. The *NCPA*
 382 MAY be virtual or real. "Virtual" means that the two *Parties* MUST configure their systems to
 383 [conform to a published default NCPA. This avoids the need to negotiate the details of an NCPA](#)
 384 [before negotiating a CPA.](#) The *Parties* can obtain each other's endpoint address by out-of-band
 385 means such as phone, fax, or discovery in an ebXML or other registry. [See Section 12.7 for](#)
 386 [additional discussion.](#)
 387

Deleted: conform to the *NCPA* defined in Appendix C but there might not be a physical *NCPA* document

388 [NOTE: It is the intention of the OASIS CPPA negotiation subcommittee to design one or](#)
 389

390 | [a family of suitable default NCPAs.](#)

391

392 “Real” means that the *Parties* create and deploy a specific *NCPA* using, for example, the
393 following procedure:

394

395 1. *Parties* A and B publish *NCPA Templates* (that they are willing to abide by) in a *Registry*.

396 They are *NCPA Templates* (as opposed to *NCPAs*) because some information (such as the

397 prospective trading partner’s *Party* ID and endpoint address) is missing from an *NCPA*

398 template. In many cases, a *Party*’s *NCPA Templates* might differ from each other only with
399 regard to which of several negotiation BPSS instance *Documents* they refer to.

400 2. *Party* B discovers *Party* A and wants to conduct trade.

401 3. *Party* B chooses an *NCPA* template of *Party* A that it can live with (say, by looking at the
402 BPSS instance *Document* pointed to by this *NCPA* template).

403 4. *Party* B then fills in this *NCPA* template with its own name, endpoint address, etc. (so that
404 now it becomes an *NCPA*) and sends it [to Party B](#).

405 5. *Party* B then starts the *Negotiation Protocol* by sending an initial offer to *Party* A. The initial
406 offer consists of a *CPA* template and the corresponding *NDD*.

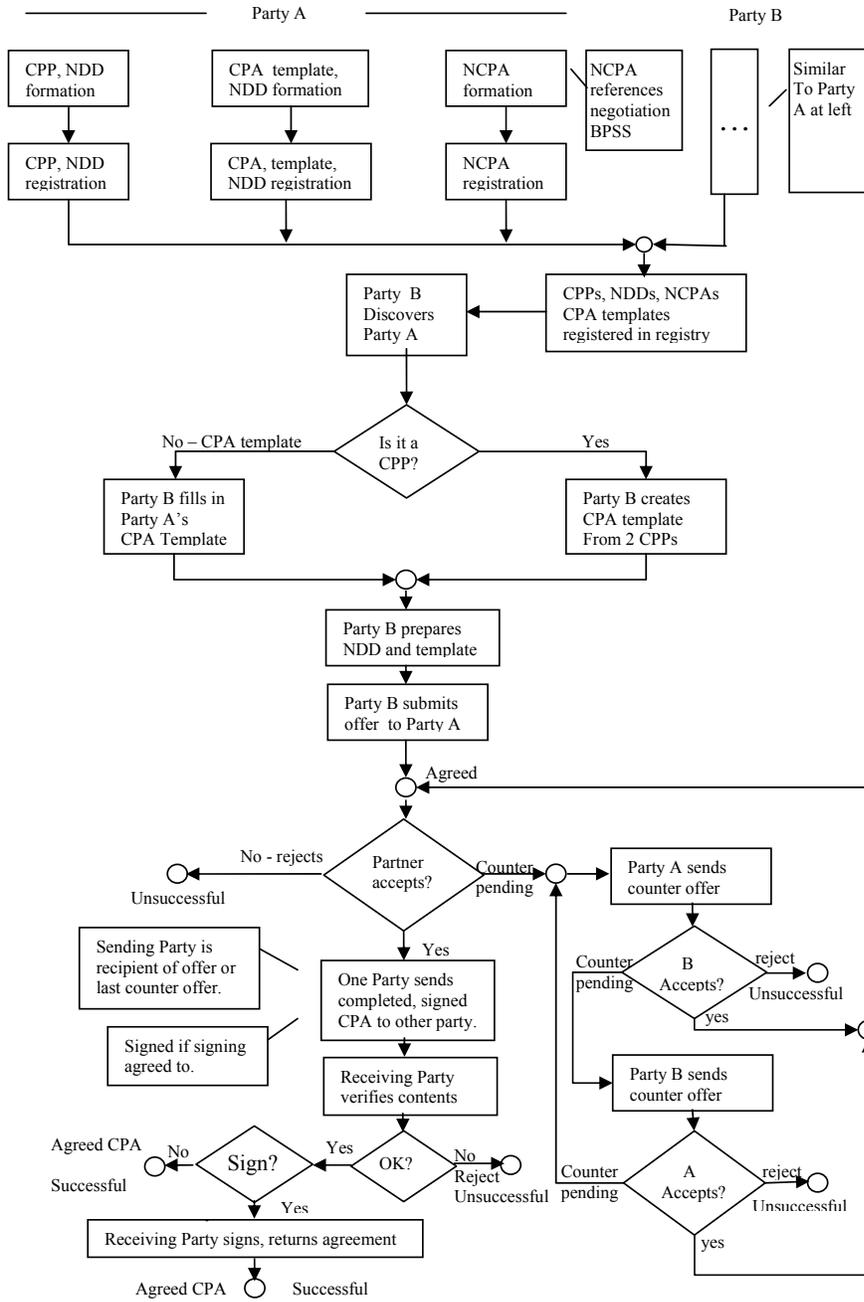
407

408 The two *Parties* can then perform the *Negotiation Protocol*, exchanging counter offers until they
409 create an agreed *CPA*. They are then ready to do electronic *Business*.

410

411 **5.2 Overview of CPA Negotiation**

412 Figure 2 is a high-level view of the *Negotiation Process*. Following are some details of the
413 *Negotiation Process* illustrated in Figure 2.



414
415
416

Figure 2. Negotiation Process

- 417 • Initial inputs:
- 418 ♦ *CPPs* and the associated *NDDs* of two prospective partners or a *CPA Template* and *NDD*
- 419 that one partner provides to a prospective partner.
- 420 • For the case of the *CPA Template* and *NDD*, the *CPA Template* might be generated
- 421 by one of the *Parties*, might be a copy of a *CPA* used by someone else that is almost
- 422 exactly what is needed, or might be supplied by a third-*Party* negotiation service.
- 423 ♦ Proposed Process-Specification *Document* (BPSS instance document)
- 424 • The *Parties* can negotiate about which BPSS instance *Document* to use based on the
- 425 name of the BPSS instance *Document* (i.e. syntactic negotiation) but not over the
- 426 details within a given BPSS instance *Document* (semantic negotiation).
- 427 • One *Party* prepares a *CPA Template* and an *NDD* that describes what is negotiable and
- 428 submits the *CPA Template* and *NDD* to the other *Party* as an initial offer.
- 429 • The two *Parties* then exchange counter offers until they arrive at a mutually acceptable *CPA*.
- 430 Offer and counter-offer information is in *Negotiation Messages* exchanged using negotiation
- 431 *Business Transactions* defined in the *NCPA* and BPSS instance *Document*.
- 432 • Result of negotiation:
- 433 ♦ A successful result is a *CPA* that is ready to sign and use, possibly subject to human
- 434 approval.
- 435 ♦ An unsuccessful result means that agreement was not possible on some items in the *CPA*.
- 436 Possibly, further human interaction could resolve the incompatibilities.
- 437 • Concluding negotiation
- 438 ♦ The *Party* that received the last counter offer builds the complete *CPA* by filling in
- 439 details such as its *Party* ID and transport endpoint address and sends it to the other *Party*.
- 440 (If it is the case that no counter offers were received during the *Negotiation Protocol*, that
- 441 is, if the *Party* that received the initial offer accepted it without sending a counter offer,
- 442 that *Party* builds the complete *CPA* by filling in details such as its *Party* ID and transport
- 443 endpoint address and then sends it to the other *Party*)
- 444 • If it was agreed that the *CPA* is to be signed, the *Party* that sends the final *CPA* signs
- 445 it before sending it.
- 446 ♦ The other *Party* verifies the contents of the completed *CPA* including, perhaps validation
- 447 of the first *Party*'s signature. If these tests are successful, that *Party* signs the new *CPA*
- 448 (if signing was agreed to) and returns it to the first *Party*.
- 449 ♦ The two *Parties* now deploy the new *CPA* and begin doing business.
- 450

451 5.3 Pre-Conditions for Negotiation

452 This section discusses conditions that MUST be met before negotiation. If these conditions are

453 not met, a successful outcome is unlikely. The discussions relate to *CPPs* or a *CPA Template* as

454 appropriate

456 The two partners MUST agree on what *Negotiation Protocol* to follow, i.e. what *NCPA* to use

457 for negotiation. (The *NCPA* identifies the *Negotiation BPSS Instance Document* to be used.)

459 There MUST be a minimum level of matching (i.e. compatibility) between two *CPPs*.

- 460 • There MUST be at least one transport protocol in common.
- 461 • There MUST be a minimum level of compatibility between at least one *DocumentExchange*

462 | [element in each CPP.](#)

463

464

Deleted: See Section 7 for related information.¶

465 | **5.4 CPP and NDD Formation and Editing**

466 | **NOTE:** These are pre-discovery steps that are out of scope for the negotiation specification,
467 they are included here in the interest of completeness. Following is a non-normative list of
468 the elements of *CPP* and *NDD* formation.

- 469 • *CPP* template (a prototype CPP that could be used for creating CPPs).
 - 470 ♦ Supplied with software installation (configured options)
 - 471 ♦ Edited to reflect preferences
- 472 • *NDD* formation.
 - 473 ♦ Although *NDD* formation is out of scope, the *NDD* schema is a key component of this
474 specification.
- 475 • Tool for custom *CPP* formation
- 476 • Tool for *CPA* and *CPA Template* formation.
- 477 • Tool for *NDD* formation
- 478 • Service(s) for supplying *CPPs* or *CPA Templates*
 - 479 ♦ UDDI advertised, SOAP, ebXML, simple HTTP GET, and so on.
- 480 • ebXML *Registry* submission (publication)

481
482 In principle, a *Party* SHOULD be able to publish both a *CPP* and a *CPA Template*. However,
483 this would lead to a problem that a given prospective trading partner might find either one. If
484 a *Party* intends that some prospective trading partners negotiate with a *CPP* while other are
485 expected to accept a *CPA Template*, then the *Party* SHOULD probably publish only the *CPP*
486 and decide whether to send a *CPA Template* based on its knowledge of who the prospective
487 trading partner is.

488 | **5.5 Discovery of CPPs and CPA Templates**

489 | **NOTE:** The discovery process is out of scope for the negotiation specification; [this](#)
490 [discussion](#) is included here in the interest of completeness. Following are some points
491 concerning the discovery process.

Deleted: it

- 492 • The minimum requirement is to be able to perform an HTTP GET of a CPP from a URL
493 obtained by means outside the scope of this specification.
- 494 • UDDI ebXML *Registry* bootstrap. This permits CPPs to be advertised in either UDDI or
495 the ebXML Registry.
- 496 • Search and retrieval in ebXML *Registry* or similar *Registry*.
- 497 • Well-known address of the *Registry*.
- 498 • A *Registry* might provide [added](#) function, perhaps as value-added services, [such as](#),
 - 499 ♦ Notification of *CPP* expirations,
 - 500 ♦ Acceptance filled-out *CPA Templates*.

Deleted: Can a

Deleted: further

Deleted: ?

Deleted: ?

Deleted: ?

501 | **5.6 Negotiation through an Intermediary**

502 | **NOTE:** Negotiation through an intermediary (negotiation broker) is out of scope for this
503 version of the specification. A *Message*-forwarding intermediary that is not aware of the
504 purpose of the *Messages* can be used if it conforms to the manner in which [ebMS] supports

505 intermediaries.

506 **6 CPA Template**

507 This section provides an overview of the use of a *CPA Template*.

508 **6.1 CPA Template and Draft CPA**

509 The *Negotiation Protocol* defined in this specification is based on the use of a *CPA* that is
510 incomplete in that [it contains items](#) that are negotiable or MUST be filled in by the *Party* that
511 receives an initial offer. Negotiable items can have “dummy” values that will later be replaced by
512 the agreed values arrived at during the *Negotiation Process*. Such an incomplete *CPA* can be
513 categorized as a *CPA Template* or a draft *CPA*.

514
515 A *CPA Template* will normally contain dummy values for the *Party*-specific values of the *Party*
516 to which the *CPA Template* is being sent as well as dummy values for other items that the
517 offering *Party* considers negotiable. A draft *CPA*, on the other hand, will typically have been
518 formed by pruning and combining *CPPs* of each of the *Parties* in the *Negotiation Process*, and
519 so can contain all “real” values. That is, using a *CPA Template* will typically require a counter
520 proposal while using a draft *CPA*, the *Party* making the initial offer might only be asking for
521 approval of the draft *CPA* rather than offering to negotiate some items. For convenience, both
522 kinds of *Document*, though having different origins, will be referred to by the term “*CPA*
523 *Template*” because the process of negotiation proceeds the same way for either *CPA Templates*
524 or draft *CPAs*. Sensible use of *CPA Templates* requires that the dummy values be indicated as
525 negotiable and that acceptance does not occur until the dummy values have been replaced. In this
526 specification, the *NDD* is the means of indicating what is negotiable.

527
528 A *CPA Template* can encompass a wide range of negotiating possibilities. At one end of the
529 range, it might amount simply to a take-it-or-leave-it offer, its *NDD* indicating only those items
530 that MUST be filled in to customize it to the other *Party*. At the other end of the spectrum, its
531 *NDD* might indicate that virtually everything is negotiable.

532
533 In the simplest case, the accompanying *NDD* might be very simple and would simply indicate
534 which elements and attributes need to be completed by the prospective trading partner, such as
535 *Party* ID and transport endpoint address. For this case, the *NDD* facilitates identifying the items
536 to be filled in, avoiding the need to label the items to be filled in within the *CPA Template* and
537 the need to parse the *CPA Template* to find those items.

538 **6.2 Advantages of Starting Negotiation with a CPA Template**

539 If negotiation is performed with the two *Parties*’ *CPPs* and an *NDD* for each, everything in the
540 *CPPs* is potentially negotiable and has to be considered during the *Negotiation Process*. The
541 process of composing a *CPA Template* from two *CPPs* will often narrow down the amount of
542 negotiation relative to the negotiation possibilities expressed in the *NDDs* that accompany the
543 *CPPs*. The reason is that many of the differences between the two *CPPs* can be “mechanically”
544 resolved by finding compatible choices and matching values of some elements or attributes. For
545 example, there might be only one transport protocol that is common to the two *Parties*. After the
546 *CPA Template* is constructed, a new *NDD* MUST be constructed that includes only the items in
547 the *CPA Template* that remain to be negotiated.

548

549 The result is that the non-controversial aspects of the agreement are recorded in the *CPA*
550 *Template* before negotiation starts. This simplifies the *Negotiation Process* by removing from
551 consideration all subjects that were resolved during the composition process. The *Negotiation*
552 *Process* operates on a smaller set of items and will converge rapidly. In addition, the process of
553 composing the new *NDD* will uncover any incompatibilities between the *Parties* before the start
554 of the *Negotiation Process*. The two *Parties* can either resolve those incompatibilities by human
555 to human contact or conclude that no resolution is possible, without having first to go through a
556 fruitless *Negotiation Process*.

557 **6.3 CPA Template composition**

558 Composition of a *CPA Template* is the same as composing any *CPA* from two *CPPs*. Appendix
559 E, “*CPA Composition (Non-Normative)*”, of [ebCPP] contains a detailed discussion of *CPA*
560 composition from two *CPPs*.

561 **7 CPP and CPA Template Content**

562 This section discusses content of the *CPP* and *CPA Template* from the viewpoint of
563 negotiability.

564 **7.1 Validation of CPP and CPA Template**

565 The rules discussed below ensure that the negotiable *CPP* or *CPA Template* can be validated by
566 an XML parser while not appearing to constrain negotiability.

567
568 In general, since the negotiability details are provided in the *NDD*, it SHOULD be acceptable to
569 include any valid arbitrary value or choice for a negotiable item in the pre-negotiation *CPP* or
570 *CPA Template*. In other words, the *NDD* overrides what is in the pre-negotiation *CPP* or *CPA*
571 *Template* for all negotiable items.

- 572
- 573 • Numerical values: Any valid value can be stated for a negotiable item in the pre-negotiation
- 574 *CPP* or *CPA Template*.
- 575 • Cardinality: All acceptable choices that are to be negotiated MUST appear in the pre-
- 576 negotiation *CPP* or *CPA Template*.

577 **7.2 Preference Order**

578 Enumerations MUST always be stated in preference order (highest preference first). In most
579 cases, preference order is REQUIRED by the CPPA specification[ebCPP]. Following are
580 examples:

- 581 • **PartyId** elements under the same **PartyInfo** element.
- 582 • CanSend and CanReceive elements under the ServiceBinding element (NEED TO VERIFY
583 THIS)
- 584 • AccessAuthentication elements under the same TransportSender element
- 585 • EncryptionAlgorithm elements under the same TransportClientSecurity or
586 TransportServerSecurity element.
- 587 • **TransportProtocol** elements under the same **Transport** element
- 588 • **AnchorCertificate** elements under the same **Certificate** element

589 **7.3 Conflicts between two Parties' Preferences**

590 When composing a *CPA Template* from its and another *Party's CPP*, a *Party* might encounter
591 unresolvable conflicts. For example, *Party 1* might allow alternative elements X and Y with a
592 preference for X while *Party 2* might allow elements X and Y with a preference for Y. In cases
593 like these, the choice can be left open in the *CPA Template* and negotiated later.

594 **7.4 CPA Period of Validity**

595 The values of the **Start** and **End** elements in the *CPA Template* SHOULD be consistent with
596 each other (start time MUST precede end time) and SHOULD be consistent with the expiration
597 times of all the certificates. It is preferable that the *CPA* expire before any of its certificates
598 expire. All of these times are negotiable but it will simplify matters if the times in the *CPA*
599 *Template* are mutually consistent. If the **Start** and **End** elements do not appear in the *CPPs*; they
600 MUST be added when the *CPA Template* is composed from the *CPPs*.

601 8 Negotiation CPA (NCPA)

602 CHANGES MAY BE NEEDED TO THE EXISTING TEXT IN THIS SECTION AND TO
 603 THE NCPA INSTANCE DOCUMENT TO HARMONIZE IT WITH THE LATEST
 604 VERSIONS OF THE NEGOTIATION MESSAGES AND BPSS INSTANCE DOCUMENT.

605
 606 THIS SECTION NEEDS TO BE REVIEWED IN DETAIL TO SEE WHAT ADDITIONAL
 607 MATERIAL IS NEEDED.

608
 609 The purpose of this section is to:

- 610 • Explain how to construct the *Negotiation CPA* such that it does not have to be negotiated;
- 611 • Explain the negotiation aspects of the *NCPA*. Principally, these aspects are the elements that
- 612 define the interface between a *CPA* and the BPSS instance *Document*, i.e., the
- 613 ***CollaborationRole***, ***ProcessSpecification***, and ***Role*** elements.

614
 615 In general, an *NCPA* SHOULD be the simplest possible *CPA* that conforms to the [ebCPP]
 616 schema. With the possible exception of selection of a negotiation BPSS instance *Document* and
 617 *Party*-specific information such as *Party* name, *Party* ID, and endpoint address, it SHOULD be
 618 possible for any pair of *Parties* to use it.

619
 620 *Message* exchanges are asynchronous. The ***ResponseToURL*** element in the *Negotiation*
 621 *Message* provides the URL for a response to the message.

622
 623 The *NCPA* defines the interactions between two *Parties* that are negotiating the contents of a
 624 *CPA*. It identifies the BPSS instance *Document* that defines the negotiation choreography. An
 625 example of an *NCPA* is in Appendix C.

626
 627 The following are minimalist requirements on the contents of the *NCPA* that help avoid the need
 628 to negotiate the *NCPA*. Depending on the particular function, negotiation can be avoided either
 629 by mandating choices or values in this specification or by mandating that a function with
 630 cardinality that includes zero be omitted.

Deleted: negotiation

631 8.1 Document Exchange

632 The following rules eliminate the need for negotiating the *Document*-exchange specifications for
 633 the *NCPA*:

- 634 • Omit the following child elements of the ***ebXMLSenderBinding*** and
 635 ***ebXMLReceiverBinding*** elements: ***ReliableMessaging***, ***PersistDuration***,
 636 ***xxxNonRepudiation***, and ***xxxDigitalEnvelope***. This means that reliable *Messaging* and
 637 *Message* security are not used.
- 638 • In the ***MessagingCharacteristics*** elements, specify the value “never” for the attributes
 639 ***ackRequested***, ***ackSignatureRequested***, and ***duplicateElimination*** (they are used only with
 640 reliable *Messaging*). For the ***actor*** attribute, specify either of the permitted values; this
 641 attribute is ignored when the value of the ***ackRequested*** attribute is “never”.

642
 643 NOTE: the negotiation subteam plans to define the following capability: *Messaging* could be

644 specified to use basic SOAP or W3C XML Protocol (when available). In this context, “basic”
645 means that values or choices that normally have to be negotiated will either be omitted or
646 will be given fixed values by this specification.

647 **8.2 Transport**

- 648 • Use HTTP PUT or POST to send a proposed CPA to a URL.
- 649 • The response to an offer or counter offer is always synchronous. This avoids the need for the
650 responder to know the URL for a response.

651
652 [For more details, see the \[ebMS\] appendices that discuss synchronous exchanges.](#)

Deleted: S

Deleted: B that discuss synchronous exchanges

653 **8.3 Packaging**

654 **COMPLETION OF THE PACKAGING DEFINITION (E.G. SIMPLEPART**
 655 **DEFINITIONS) AWAITED COMPLETION OF THE NDD AND NEGOTIATION**
 656 **MESSAGE SCHEMAS. THE PACKAGING DEFINITION SHOULD BE COMPLETED**
 657 **NOW IN CONFORMANCE WITH THE CURENT NDD AND NEGOTIATION MESSAGE**
 658 **SCHEMAS.**

Deleted: S

660 **8.4 Security**

661 [Security of the negotiation protocol is not defined in this version of the specification. Negotiating](#)
 662 [Parties can use SSL with basic authentication to obtain a degree of security.](#)

664 **8.5 Explanation of NCPA Example**

665 The [instance document for the NCPA example](#) is in Appendix C.

Deleted: text of

666 **THE CURRENT NCPA EXAMPLE INSTANCE DOCUMENT IS UNCHANGED FROM**
 667 **THE JANUARY, 2003 DRAFT. IT MAY HAVE TO BE CHANGED TO CONFORM TO**
 668 **THE BPSS INSTANCE DOCUMENT IN THIS DRAFT, ESPECIALLY WITH REGARD TO**
 669 **THE CHANGES THAT HAVE BEEN MADE TO THE BPSS INSTANCE DOCUMENT IN**
 670 **THE AREA OF CONCLUSION OF NEGOTIATION.**

671 **EXPLANATORY TEXT IS NEEDED HERE.**

674 **9 Negotiability of CPA Elements and Attributes**

675 This section discusses the negotiability of the different elements and attributes in the *CPA* and is
676 concerned mostly with composing a *CPA* from two *CPPs*. It focuses on those cases that involve
677 special considerations.

678 **9.1 Enumerations**

679 There are several cases of enumerations:

- 680 • Some enumerations are laid out in the *CPP* instance *Documents* (e.g. certificates).
- 681 • Some enumerations are laid out in the CPPA schema itself.
- 682 • Some enumerations might be defined only in the text of the CPPA specification and would
683 have to be put into the *NDD* schema.
- 684 • Some enumerations are not listed in full anywhere (e.g. the W3C forms of encryption
685 algorithm name)
- 686 • Some might be defined elsewhere, perhaps as a set of URIs.

687
688 In some cases, especially those that are defined in the CPPA schema, only the items in an
689 enumeration that are acceptable to the *Party* that is preparing the *NDD* instance *Document* have
690 to be listed in the *NDD*. An example is the versions of the specification that are acceptable to the
691 *Party*.

692
693 The CPPA schema itself is input to the *Negotiation Process*. Therefore, enumerations that are
694 defined in full in the CPPA schema don't necessarily have to be defined in full in the *NDD*
695 schema.

696 **9.2 CollaborationRole element and its child elements**

697 The normal case is that the two *CPPs* are being composed into a *CPA Template* specify the same
698 BPSS instance *Document*. In version 1 of this specification, the contents of the BPSS instance
699 *Document* cannot be negotiated using the negotiation functions defined in this specification.
700 Two prospective trading partners SHOULD agree on the same BPSS instance *Document* and
701 assignment of *Roles* before beginning to negotiate the *CPA*. In many cases, agreement will be
702 established by the fact that the two prospective trading partners have compatible
703 **CollaborationRole** subtrees in their *CPPs*. The following considerations relate to establishing
704 compatible **CollaborationRole** subtrees.

- 705 • If both *CPPs* specify the same *Role* (e.g. both specify "buyer"), the situation cannot be
706 resolved automatically. Human contact is needed and one *CPP* MUST be changed to specify
707 the other *Role*.
- 708 • If both *CPPs* specify both *Roles* (i.e. two **CollaborationRole** elements with opposite *Roles*),
709 this cannot be resolved automatically. Human contact is needed and the two *Parties* MUST
710 agree on which *Party* plays which *Role*.
- 711 • If *CPP* A specifies one *Role* and *CPP* B specifies both *Roles*, chose the *Role* in *CPP* B which
712 is opposite to the *Role* specified in *CPP* A.
- 713 • If both *CPPs* specify more than one BPSS instance *Document* but there is only one in
714 common to the two *Parties*, use that one.
- 715 • If both *CPPs* specify more than one BPSS instance *Document* that is in common to both of

716 them, human contact is needed to decide whether all the common ones are to be used in the
717 *Business Collaboration* or which one is to be used.

718
719 From the viewpoint of *CPA* composition and negotiation, the best practice is to include only one
720 BPSS instance *Document* in each *CPP*.

721
722 NOTE: A *Party* can describe the *Business Collaboration* using any desired alternative to
723 the ebXML *Business Process Specification Schema*. When an alternative *Business-*
724 *Collaboration* description is used, the *Parties* to a *CPA* MUST agree on how to interpret
725 the *Business-Collaboration* description and how to interpret the elements in the *CPA* that
726 reference information in the *Business-Collaboration* description. The affected elements
727 in the *CPA* are the **Role** element, the **CanSend** and **CanReceive** elements, the
728 **ActionContext** element, and some attributes of the **BusinessTransactionCharacteristics**
729 element. The two *Parties* also have to come to a common understanding of how to
730 negotiate the negotiable elements and attributes whose interpretations are changed by the
731 use of the alternative *Business Collaboration* description.

732

733 9.3 Elements or Attributes whose Cardinality Includes Zero

734 Regarding elements or attributes whose cardinalities include zero (omission), the main
735 negotiable thing is “presence or absence”. However, if it is agreed to include (one or more of)
736 that element or attribute, it is then necessary to negotiate the value (or child elements in the case
737 of an element) of each one that is included. **PersistDuration** is an example. If the two parties
738 agree to include it, they then have to negotiate its value.

739 9.4 Values

740 For negotiating values, the negotiation depends on the type of value. It could be a range of
741 values, a step size, members of an enumeration, etc. The type information is in the CPPA
742 schema and might not have to be repeated in the *NDD*.

743 9.5 Items that are Referred to

744 If some element, *A*, refers to another element, *B*, by means of an IDREF and element *A* is
745 negotiated, then there is also a need to negotiate what element *A* refers to (element *B*). Element
746 *B* must be inserted into the *CPA*-under-construction. For example, an anchor certificate
747 reference points to a trust anchor that must be added to the *CPA*.

748 9.6 Transport Endpoints

749 Any *Party* can define whatever endpoints it chooses. There might be issues of matching
750 endpoint characteristics. One example is the endpoint type. *Parties* might need to negotiate
751 what endpoint types are used.

Deleted: Transport endpoints are not really negotiable since

752 9.7 Security

753 Negotiation of security agreements for collaboration can involve decisions to add or forego
754 particular security features. For example, it might not be feasible to configure software for PKI
755 based authentication, and the decision could be made to use weaker forms of basic (username
756 and password) authentication combined with a transport data confidentiality option, such as SSL-
757 3.

758 [NDDs might indicate a willingness to change security features at the higher](#)
 759 [BusinessTransactionCharacteristics level as well as details related to the type of security to be](#)
 760 [used \(its strength, allowed algorithms, and so forth.\)](#)
 761

762
 763 [Finally, it might simply not be feasible to arrange for implementing certain security features that](#)
 764 [are regarded as essential for the collaboration contract. Until compatible software is obtained, or](#)
 765 [other security contracts are made, negotiation could fail to arrive at a mutually acceptable CPA.](#)
 766

767 [It should be noted that negotiation on certificates might require human input. A Party's](#)
 768 [unwillingness to handle the proposed trust model is a reason for failure of the negotiation.](#)
 769

769 **9.7.1 Trust Anchor and Certificate Alignment**

770 [This section discusses the kinds of negotiation that might take place for aligning SecurityDetails](#)
 771 [and TrustAnchors with various Certificates.](#)
 772

773 [The need for "aligning" TrustAnchors with various Certificates arises from the validation](#)
 774 [process for certificates. In this process, a certificate is examined to see whether the association of](#)
 775 [name and the distributed public key is to be trusted in one of its particular uses, such as for](#)
 776 [checking a digital signature or for encrypting a symmetric key as part of key exchange. One](#)
 777 [aspect of this validation process checks a certificate to see whether its issuer \(or iteratively, an](#)
 778 [issuer of its issuer\) is itself trusted as vouching for the certificate being a public key of the named](#)
 779 [entity. It does this by checking the signature on the subject certificate against the issuer's](#)
 780 [certificate. The certificates of the trusted issuers are called trust anchors. \(For a self-signed](#)
 781 [certificate, where the issuer and subject are the same, the trust anchor certificate is also the](#)
 782 [distributed certificate.\) \(Other aspects of validity tests involve seeing whether certificates have](#)
 783 [expired or been revoked; these tasks are not discussed here.\)](#)
 784

785 [For collaborators to align TrustAnchors with Certificates, either one party MUST obtain a](#)
 786 [certificate that will be validated by the other party's TrustAnchors, or one party MUST update](#)
 787 [its TrustAnchors so that the other party's certificate is validated.](#)
 788

789
 790 There are 3 major levels for alignments in public-key infrastructure (PKI).

- 791 1. Transport-level security
- 792 2. Messaging-level security ([digital envelope and digital signature](#))
- 793 3. Application-level security

794
 795 [For transport-level security, \(transient\) encryption and authentication alignment are needed. Both](#)
 796 [server-side and client-side SSL or TLS need to have the trust anchors synchronized with](#)
 797 [corresponding certificates.](#)
 798

799 [For messaging-level \(persistent\) security, both digital envelopes for data confidentiality and](#)
 800 [digital signatures for non-repudiation \(of origin and/or receipt\) also require alignment.](#)
 801

Deleted: <#>Trust Anchors and Related Matters¶
 This section discusses the kinds of negotiation that might take place for aligning *SecurityDetails* and *TrustAnchors* with various *CertificateRefs*.¶

Inserted: <#>Trust Anchors and Related Matters¶
 This section discusses the kinds of negotiation that might take place for aligning *SecurityDetails* and *TrustAnchors* with various *CertificateRefs*.¶
 Security¶

Deleted: Security

For application-level (persistent) security, digital envelopes and non-repudiation (of origin and/or receipt) by means of digital signatures require alignment.

There are two main cases to be dealt with:

1. Add a certificate authority (CA) as a trust anchor.
2. Allow direct trust using a self-signed certificate. The self-signed certificate must be referenced as a trust anchor.

In general, if the proposer of the draft CPA seeks to request that the other party acquire a new certificate, the draft CPA may contain a placeholder for the certificate, and the NDD will indicate a need to supply a certificate. If the proposer seeks to request that the other party change its TrustAnchors, the proposer should make this addition to the other party's TrustAnchors, and then mark the item as negotiable (in case, a different issuer certificate somewhere in a chain of issuers is to be added as a trust anchor instead of the proposed issuer certificate.)

It is worth explicitly discussing the case where a certificate can be a self-signed certificate. The draft CPA proposer might add a self-signed certificate to the othe party's TrustAnchors/AnchorCertificateRef list. If the self-signed certificate was found in the CPP of the other party, the proposer might also update its own TrustAnchors so that the other party's self-signed certificate is referenced. These proposals can be viewed as adopting a direct trust model, rather than a hierarchical model involving certificate authorities.

Finally, when aligning TrustAnchors with the self-signed certificate fails, the security function resting on this PKI alignment might be changed. In this last case, the negotiation option might involve a change in the value of an attribute under BusinessTransactionCharacteristics, and so would need to be reviewed against any contracts in effect. This review would presumably not be an automated process in the near term.

As a result of the CPA Template formation process, various details could be up for negotiation.

First, a change to the PKI might be proposed. For the self-signed certificate addition option, the negotiatee might want to:

1. Reject adding a self-signed certificate and indicate rejection of the security function resting on this PKI alignment
2. Insist on the proposer getting a certificate from an existing CA.
3. Propose issuing another certificate signed by an acceptable authority.

For case 1, the negotiation "space" would involve a change in the value of an attribute under BusinessTransactionCharacteristics.

For case 2, the negotiatee would have to indicate rejection of the CPA Template and indicate that until the CPP certificate value changes, there will be no forward progress. The proposer would have to go out and get a new certificate.

Deleted: For transport-level security, (transient) encryption and authentication alignment are needed. Both server-side and client-side SSL or TLS need to have the trust anchors synchronized with corresponding certificates.¶

¶ For Messaging-level (persistent) security, digital envelopes and non-repudiation (of origin and/or receipt) by means of digital signatures require alignment. ¶

¶ For application-level (persistent) security, digital envelopes and non-repudiation (of origin and/or receipt) by means of digital signatures require alignment. ¶

Deleted: Failure to validate a certificate need not prevent formation of a CPA Template. First, the sender's signing certificate can be a self-signed certificate. If so, a reference to this self-signed certificate can be added to the receiver's TrustAnchors and AnchorCertificateRef lists. This proposal amounts to proposing to agree to a direct trust model, rather than a hierarchical model involving certificate authorities. Second, a proposal to add a trusted root might be made, again by appropriate revision of the TrustAnchors element.¶

848
849 For case 3, the negotiatee would propose a different certificate issued by its own CA. The
850 negotiatee would have to install it and use it for this transaction. This is not yet a common
851 practice, though it is logically possible. This would involve one side being a CA for the *Business*
852 *Process* and the ability of the other side to use more than one certificate for its existing key-pair.
853 The *CPA* proposed to do this would go outside of anything strictly derivable from the *CPP* (only
854 the old X.509 certificate would be used to put together a new X.509 certificate from a new
855 issuer).

856 Next, for the PKI trust anchor certificate addition option, the negotiatee might want to:

- 857
858
- 859 1. Reject adding a new CA to its trust anchors and indicate rejection of the security function
 - 860 resting on this PKI alignment.
 - 861 2. Insist on the proposer getting a certificate from some already trusted existing CA.
 - 862 3. Propose accepting another certificate signed by its own signing authority.
 - 863 4. Propose a different trust anchor either higher or lower in the validation chain than the one
 - 864 proposed by the other side.
- 865

866 Again, as for adding a self-signed certificate, for case 1, the negotiation "space" would involve a
867 change in the value of an attribute under the *BusinessTransactionCharacteristics* element. For
868 case 2, the response would have to be rejection with a call for a change in *CPP*. For case 3, the
869 negotiatee proceeds as described in case 3 above.

870
871 The new case 4 is logically possible but still exotic. In effect, the negotiation SHOULD not
872 matter to the other side, because it is just an adjustment to which trust anchor is added to one
873 side's PKI trust list and the certificate used would still validate to the alternative trust anchor. Yet
874 it would reflect a slight change in security details.

875 **9.8 Discussion of Various Elements and Attributes**

876 This section discusses some examples of negotiation of various elements and attributes. It is not
877 intended to be an exhaustive discussion of everything in the CPA.

878
879 *cpaid* attribute: The value of the *cpaid* attribute can be negotiated. In order to negotiate the value
880 of the *cpaid* attribute, it SHALL be a URI.

881
882 *PartyInfo* element: This element cannot be negotiated. There MUST be one in the CPA for each
883 Party. Various attributes and child elements can be negotiated.

884
885 *Start* and *End* elements: The value of the *Start* element MUST precede the value of the *End*
886 element and the times stated in the *Start* and *End* elements MUST NOT be outside the certificate
887 validity periods. If the values of the *Start* and *End* elements are negotiable, the *CPP* SHALL
888 specify the earliest acceptable start time and the latest acceptable end time.

889
890 *Status* element: The *Status* element is not negotiable; its value identifies the state of the
891 negotiation. The negotiation algorithm is responsible for changing the state at appropriate times.

892

893 [ConversationConstraints](#) element: First the *Parties* MUST agree on whether this element is to
 894 be used. Then, they MUST agree on the values of its *invocationLimit* and
 895 *concurrentConversations* attributes.

896
 897 **defaultMSHChannelId:** Since a delivery channel contains both *Parties* ' properties, the two
 898 *Parties* have to agree on both *Parties* ' default delivery channels.

899
 900 [PartyInfo](#) element: The presence of the *PartyInfo* element is not negotiable; there MUST be
 901 one for each *Party*. The contents of the *PartyInfo* element are negotiable.

902
 903 [NOTE:](#) The case where one or both *CPPs* contains more than one *PartyInfo* element
 904 will be discussed in a future version of this specification.

905
 906 **PartyId** type: The *type* attribute of the *PartyId* element identifies the naming system to which
 907 the *PartyId* belongs (e.g. DUNS). The *Negotiation Process* SHOULD select one possible
 908 *PartyId* type for each *Party* and eliminate any others that are in the *CPPs*. Each *Party*'s *PartyId*
 909 type MUST be understandable by the other *Party*. Eliminating the others ensures that each *Party*
 910 will always use the same *PartyId* for the other *Party*.

911
 912 **PartyRef:** One reason to negotiate is that a *Party* might not be able to understand the other
 913 *Party*'s *PartyRef Document*. For example, the geographical contexts might not match. While
 914 negotiating the contents of the *PartyRef Document* is out of scope for this specification,
 915 negotiating the contents might lead to negotiating the schema (type), which is in scope.

916
 917 **CollaborationRole:** the cardinality is one or more.

918
 919 **name** attribute of the *ProcessSpecification* element: This is not negotiable unless a future
 920 version of [ebBPSS] provides for more than one *ProcessSpecification* element in a BPSS
 921 instance *Document*.

922
 923 **Role:** The two *Parties* have to have opposite *Roles* in a *Business Collaboration*. This MUST be
 924 validated. There is no known use case for negotiating it.

925
 926 **ApplicationCertificateRef:** This is negotiable because one party's certificate authority might not
 927 be acceptable to the other *Party*. The value of the *certId* attribute could be an enumeration of
 928 possible certificates. There can be zero or more *ApplicationCertificateRef* elements.

929
 930 **ThisPartyActionBinding:** In general, each *Party* has to know the name that the other *Party* uses
 931 for each action but they don't need to negotiate since there is no reason for the names to match.

932
 933 **PackageId** might be negotiable.

934
 935 **ActionContext:** This is not negotiable. If BPSS is not being used, ignore the *ActionContext*
 936 element.

937
 938 **CollaborationActivity:** This allows a *Party* to specify a complete path inside the BPSS instance

Deleted: possible

Deleted: *version* attribute of the *ProcessSpecification* element: The two *Parties* ' *CPPs* might specify the same BPSS instance *Document* but different versions of it.¶

¶
 THE VERSION ATTRIBUTE OF THE BPSS PROCESSSPECIFICATION ELEMENT IS ACTUALLY THE VERSION OF THE SPECIFICATION. THE BPSS SPECIFICATION DOES NOT DEFINE A VERSION ATTRIBUTE OF A BPSS INSTANCE DOCUMENT. DECISIONS HAVE TO BE MADE ON HOW TO RESOLVE THE DISCREPANCY BETWEEN [EBCPP] AND [EBBPSS] AND THEN TO REWRITE REFERENCES TO THE VERSION ATTRIBUTE IN THIS SPECIFICATION. ONE SOLUTION IS TO CORRECT [EBCPP] TO REFER TO THE VERSION OF [EBBPSS] AND CHANGE THIS SPECIFICATION ACCORDINGLY. ANOTHER IS TO DEFINE A BPSS INSTANCE DOCUMENT VERSION AND CORRECT BOTH [EBCPP] AND THIS SPECIFICATION TO AGREE WITH THE BPSS SPECIFICATION. THE LATER SOLUTION IS FOR A FUTURE VERSION OF ALL THREE SPECIFICATIONS.¶

Deleted: *ds:Reference* child of *ProcessSpecification* element: IT IS TO BE DETERMINED WHETHER BOTH PARTIES MUST HAVE *DS:REFERENCE* IF EITHER HAS IT. IT HAS BEEN SUGGESTED THAT THIS IS NECESSARY SO THAT IF EITHER PARTY VALIDATES THE BPSS INSTANCE DOCUMENT USING *DS:REFERENCE*, BOTH PARTIES SHOULD VALIDATE.¶

939 *Document*. Its value is completely determined by the structure of the BPSS instance *Document*
940 and is therefore not negotiable.

941

942 ***channelId***: The *Parties* can negotiate which delivery channels to use or add new ones.

943

944 ***Certificate***: An enumeration of *keyinfo* types might be useful to help decide which certificates
945 are acceptable.

946

947 ***DeliveryChannel***: Cardinality is negotiable. It is suggested that a new delivery channel be
948 created rather than modifying an existing one.

949

950 ***Signing the CPA***: Negotiation of signing is accomplished by negotiating presence of the *CPA*
951 *Signature* element and its child *ds:Signature* elements. See Section 12.12 for details.

952

953 ***Comment***: [ebCPP] states that all comments in both *CPPs* SHALL be included in the *CPA*
954 unless the *Parties* agree otherwise. Therefore, each ***Comment*** element is separately negotiable.
955 Since comments are arbitrary text strings, negotiation about ***Comment*** elements MUST be by
956 human to human contact.

957 **10 Negotiation Descriptor Document**

958 NEELAKANTAN KARTHA POINTED OUT THAT THE FOLLOWING ANALYSIS IS
 959 NEEDED:

- 960 • CHECK WHETHER THERE IS SUFFICIENT SUPPORT IN THE NDD AND
 961 NEGOTIATION MESSAGES TO DO WHAT IS NEEDED TO SUPPORT THE PKI
 962 INFRASTRUCTURE, SUCH AS ADDING A NEW CERTIFICATE AUTHORITY.
- 963 • MAKE SURE THAT THE MESSAGES SUPPORT ALL THE INTERACTIONS
 964 SUGGESTED BY THE NDD. A WAY TO DO THIS IS TO TAKE THE SAMPLE NDD
 965 INSTANCE DOCUMENT AND VERIFY THAT ALL THE INTERACTIONS
 966 SUGGESTED IN IT CAN BE CARRIED OUT USING THE MESSAGES.

← --- Formatted: Bullets and Numbering

967
 968 The *Negotiation Descriptor Document (NDD)* describes what is negotiable in the accompanying
 969 *CPP* or *CPA Template*. It SHALL describe only the negotiable elements and attributes and
 970 SHALL omit those elements and attributes that are not negotiable.

971
 972 The *NDD* identifies the *CPP* or *CPA Template*. The *CPP* or *CPA Template* does not identify the
 973 *NDD* since a *Party* might have many different *NDDs* associated with the same *CPP* or *CPA*
 974 *Template*. These could be for different *Negotiation Processes*, different categories of partner,
 975 etc.
 976

977 **10.1 Use of NDD**

- 978 • An *NDD* can be placed in a *Registry* along with the *CPP*. The *NDD* and *CPP* would have to
 979 be connected by *Registry* metadata. Alternatively, a *Party* might choose not to include an
 980 *NDD* in the *Registry*. Instead, when a *Party* is discovered by a prospective trading partner,
 981 the *NDDs* can be exchanged prior to the opening step of the negotiation. This permits a *Party*
 982 to send an *NDD* that it considers appropriate for the particular prospective trading partner.
- 983 • An *NDD* is sent from the *Party* making the initial offer to the other *Party* during initialization
 984 of the *Negotiation Protocol*. After that, the *NDD* is not modified during negotiation and is
 985 not again sent from one *Party* to the other. All information about the state of negotiation of
 986 the negotiable items is exchanged in the *Negotiation Messages*.

987
 988 NOTE: This means that an item which is initially not negotiable cannot be made negotiable
 989 during the *Negotiation Protocol*.

990 **10.2 General Principles of Contents of NDD**

991 The *NDD* has been defined in an abstract manner to enable it to be applied to any kind of XML
 992 agreement. This avoids the need to define a new *NDD* schema for each kind of *Document* to be
 993 negotiated.

994
 995 NOTE: The abstract level of the *NDD* is an opportunity for tool vendors to produce *NDD*
 996 composition tools. Such a tool would have a GUI that would tailor the view of the *NDD* to
 997 the specific kind of *Document* to be negotiated. The tool would reference the schema of the
 998 *Document* being negotiated along with the *NDD* being constructed. This will supply the tool

999 with sufficient information to make the views understandable by someone who is composing
 1000 an *NDD*. This would enable that person to communicate with the tool in terms of the
 1001 specifics of the *Document* to be negotiated. The tool could then construct the *NDD* instance
 1002 *Document* in accord with the *NDD* schema.

1003
 1004 The *NDD* references both the *CPA Template* and the *CPPA XML* schema.

1005
 1006 The *NDD* consists of a variable length (cardinality 1 or more) set of [XPath] expressions, each
 1007 of which refers to a negotiable element or attribute.

1008
 1009 With each XPath expression, the negotiability of the element or attribute is defined by child
 1010 elements. These child elements represent the negotiability characteristics of the element or
 1011 attribute identified by the XPath statement. Examples are:

- 1012 • For a numeric value, minimum, maximum, and negotiation step size
- 1013 • For choices, XPath statements, ID attribute values, qnames, element values, etc. which
 1014 identify the specific choices within the *Document* being negotiated. Examples in the *CPA*
 1015 are certificates, delivery channels, transport protocols, and signature algorithms.

Deleted: <#>Cardinality (range of permitted cardinalities)¶

1016
 1017 The following rules define what is negotiable at the point referenced by an XPath expression:

- 1018 1. If the XPath expression references a non-leaf element, that element, and the whole sub tree
 1019 below that element, are negotiable.
- 1020 2. If the XPath expression references any attribute, it means that only that attribute is
 1021 negotiable and doesn't imply anything about the containing element or the rest of the sub tree
 1022 descended from the element containing that attribute.
- 1023 3. If the XPath expression references a leaf element, only that element and its contained
 1024 attributes are negotiable.

1025
 1026
 1027 **10.3 Composition of an NDD for a CPA Template**

1028 Formally, the negotiation defined in this specification begins when an offering Party, Party A,
 1029 presents an initial offer, consisting of a CPA Template and the corresponding NDD, to Party B.
 1030 If Party A cannot obtain Party B's NDD, Party A can offer its own NDD. If Party A can obtain
 1031 Party B's NDD, Party A SHALL make use of Party B's NDD in composing the NDD for the
 1032 initial offer to avoid situations in which the negotiation is certain to fail. The new *NDD* would be
 1033 a composite of the two sets of requirements that is acceptable to both *Parties* as a starting point
 1034 in negotiation. In the rest of this section, the normative statements are normative only if the
 1035 offering Party has obtained the other Party's NDD.

Deleted: one

Deleted: its

Deleted: another

Deleted: However, the following RECOMMENDATIONS on constructing an *NDD* for a *CPA Template* might facilitate the *Negotiation Process*.¶

1036
 1037 Party A's taking Party B's NDD into account will speed up convergence as well as recognition
 1038 of fatal incompatibilities and reduce the possibilities of unnecessary rejects during negotiation. In
 1039 other words, composing a CPA Template and combined NDD before starting negotiation
 1040 simplifies the Negotiation Process by:

Deleted: ¶
 If the initiating *Party* has access to the other *Party's NDD* that goes with its *CPP*, the initiating *Party* SHOULD use both its and the other *Party's NDD* to establish the *NDD* and *CPA Template* to be used as the initial offer.

- 1041 1. Removing subjects from negotiation that can be handled by simple matching.
- 1042 2. Quickly recognizing the existence of fatal incompatibilities. For fatal incompatibilities,
 1043 human to human contact to resolve the incompatibilities is RECOMMENDED.

1044
 1045 | In composing the *NDD* of the *CPA Template*, *Party A* SHALL exclude from the new *NDD*
 1046 anything that *Party A* understands (from *Party B*'s *NDD*) is not negotiable or is unacceptable to
 1047 *Party B*. For example, for an enumeration, the new *NDD* SHALL include only those choices
 1048 that are common to both of the original *NDD*s. For a range of values, *Party A* SHALL put in the
 1049 new *NDD* only the common range. If, for some element, *Party A* had specified values of 1-9
 1050 and *Party B* had specified values of 3-12, the new *NDD* SHALL specify values 3-9. The
 1051 intersection process might identify items with no common ground, making successful negotiation
 1052 unlikely.

Deleted: fParty A is
 Deleted: SHOULD
 Deleted: SHOULD
 Deleted: SHOULD
 Deleted: SH
 Deleted: OULD

1053
 1054 | *Party A* SHALL NOT include items in the new *NDD* that were not in *Party B*'s original *NDD*
 1055 because *Party B* did not intend to negotiate on the items that it did not put in its original *NDD*.
 1056 For those items that were not in *Party B*'s initial *NDD*, *Party A* MUST either accept what is in
 1057 *Party B*'s *CPP* or MUST recognize that there is an irreconcilable conflict.
 1058

Deleted: SH
 Deleted: OULD

1059 **10.4 Explanation of Contents of NDD**

1060 This section discusses the schema and example of an *NDD* instance *Document*. See Appendix A
 1061 for the schema and Appendix F for an example of an instance *Document*.
 1062

1063 The *NegotiationDescriptor* element is the top element of an *NDD*. It is a container element,
 1064 that contains one or more instances of the negotiable parts called *NegotiableInformationItem*
 1065 elements. The *documentLocation* attribute of the *NegotiationDescriptor* element is a URI that
 1066 points to the XML *Document* to which this *NDD Document* corresponds. For instance, if the
 1067 *NDD* pertains to a *CPA*, the *documentLocation* attribute points to that *CPA*.
 1068

1069 Each *NegotiableInformationItem* element contains an *xpath* attribute that identifies the
 1070 negotiable information item with respect to the *Document* pointed to by the *documentLocation*
 1071 attribute of the *NegotiationDescriptor* element.
 1072

1073 Each negotiable information item (which could be an XML element or an attribute) is one of the
 1074 following types, depending on what kind of negotiation that one needs to perform on this
 1075 negotiable information item.
 1076

- 1077 | 1. Value: For negotiating the value of the item, if the value of the item needs to be just filled in
 1078 by one party (instead of negotiated), the attribute *mustBeFilledIn* can be used to specify this
 1079 possibility.
- 1080 | 2. UnorderedValue: For negotiating the presence or absence of a member of a set of unordered
 1081 values.
- 1082 | 3. OrderedValue: For negotiating to choose among the members of a set of ordered values,
 1083 where the preference is of a simple kind (namely, a preference for earlier values or later
 1084 values in the set).
- 1085 | 4. ValuesWithPreferenceMeasure: For negotiating to choose among the members of a set of
 1086 values, where the preference measure is of a more complicated nature. For instance, it is
 1087 often possible to express the preference measure by a piecewise linear function. The
 1088 preference measure is approximated by a piecewise linear function, which then is specified
 1089 by defining each piece. Each piece is defined by giving the two (x,y) coordinates that define

Deleted: (for instance, expressed by a piecewise linear function or a function defined by an equation).

- 1090 [the start and end of the piece; it is assumed that the \(x,y\) coordinates that define the end of](#)
 1091 [one piece are identical to the \(x,y\) coordinates that define the start of the next piece. The y-](#)
 1092 [axis has an arbitrary scale \(say 1-100\) that gives the range of preferences.](#)
- 1093 5. PresentOrNot: For negotiating the presence or absence of a value. This type allows one to
 1094 express that a *Party* (a) insists that a value MUST be present; (b) insists that a value is
 1095 absent; (c) is ok with the value being present or absent, but has a preference for one or the
 1096 other or (d) is o.k with the value being present or absent, and has no preference.
 - 1097 6. IntegerValues: For expressing (a) whether an integer value is present or not (as in
 1098 PresentOrNot) and then (b) the choice between different integer values using simple
 1099 preference measures (such as smaller ones being preferred or larger ones being preferred) or
 1100 more complicated preference measures (such as those expressible via piecewise linear
 1101 functions). This type is provided mostly for convenience, since there are many entries in a
 1102 *CPP* or *CPA* that impose these kinds of negotiation requirements.
 - 1103 7. Preference: For expressing preference among values of a similar nature (such as multiple
 1104 elements at the same level, e.g., the *PartyInfo* element)
 - 1105 8. Cardinality: [This is for expressing preferences among different cardinalities using simple](#)
 1106 [preference measures \(such as smaller cardinalities being preferred or larger cardinalities](#)
 1107 [being preferred\) or more complicated preference measures \(such as those expressible via](#)
 1108 [piecewise linear functions\).](#)
 - 1109 9. BooleanValues: For expressing (a) whether a Boolean-valued item is present or not and then
 1110 (b) for expressing preference for either true or false as the value of the boolean-valued item.
 - 1111 10. DurationWithPreference: For expressing (a) whether a duration-valued item is present and
 1112 then (b) to give maximum and minimum possible values of the duration and to express a
 1113 preference for smaller values or larger values.

Deleted: Similar to IntegerValues.

1114 For more details, comments and examples of using each of these types, the reader is directed to
 1115 the *NDD* schema (Appendix A) and instance *Document* (Appendix F).
 1116
 1117

1118 **11 Negotiation Messages**

1119 [SEE THE NOTE IN THIS TYPE FACE AT THE BEGINNING OF THE NDD SECTION.](#)

1120
1121 [A Negotiation Message includes the details of a offer or a counter offer, identification of the](#)
1122 [NDD and CPA Template being negotiated, and information that controls the Negotiation](#)
1123 [Protocol. Some Messages include the NDD and the CPA Template or their URLs.](#)

1124
1125 [This section defines and discusses the details in the Negotiation Message in terms of the](#)
1126 [individual XML elements and attributes. The discussion is illustrated with XML fragments.](#)

1127
1128 [See Appendix B for the complete Negotiation-Message XML Schema. See Appendix G for](#)
1129 [examples of Negotiation-Message instance Documents.](#)

1130
1131 **[THIS SECTION HAS BEEN COMPLETELY REPLACED.](#)**

1132 **11.1 Negotiation Message Structure**

1133 This section discusses the overall structure of the *Negotiation Message*. Subsequent sections
1134 discuss each of the elements in more detail.

```
1135
1136 <NegotiationMessage
1137     xmlns:tp="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpa-negot-1_0.xsd"
1138     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:cppa="http://www.oasis-
1139     open.org/committees/ebxml-cppa/schema/cpp-cpa-2_0.xsd"
1140     xsi:schemaLocation="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpa-negot-
1141     1_0.xsd"
1142     businessMsgId="busMsg002"
1143     binding="false"
1144     inresponseTo="busMsg001"
1145     negotiationDialogId="negotDialog001"
1146     offerId="offer001"
1147     messageType="CounterOffer"
1148     error="ExpiredCPP">
1149     <NCPA uri="http://..." />
1150     <CPAIdentity>
1151     ...
1152     </CPAIdentity>
1153     <cppa:SecurityDetails cppa:securityId="ID">
1154     ...
1155     </cppa:SecurityDetails>
1156     <InitiatingParty>
1157     ...
1158     </InitiatingParty>
1159     <RespondingParty>
1160     ...
1161     </RespondingParty>
1162     <BPSSBusinessDocumentName name="CPA_Counter_Offer_Doc" />
1163     <ExpirationDate>...</ExpirationDate>
1164     <BusinessDocuments>
1165     ...
1166     </BusinessDocuments>
```

```

1167 <NegotiationContent>
1168     ...
1169 </NegotiationContent>
1170 <ResponseToURL>
1171     ...
1172 </ResponseToURL>
1173 <Offer-CounterOffer-Acceptance-Time>
1174 </Offer-CounterOffer-Acceptance-Time>
1175 <Comment/>
1176 </NegotiationMessage>

```

11.1.1 NegotiationMessage element

The *NegotiationMessage* element is the root element of the *Negotiation Message xml Document*. The *Negotiation Message Document* contains the following REQUIRED[XML]

Namespace[XMLNS] declarations:

- 1181 • The default namespace: xmlns==
- 1182 "http://www.oasis-open.org/committees/ebxml-cppa/schema/cpa-negot-1_0.xsd"
- 1183 • The schema instance namespace: xmlns:xsi=<http://www.w3.org/2001/XMLSchema-instance>
- 1184 • The ebXML CPPA namespace: xmlns:cppa=
- 1185 http://www.oasis-open.org/committees/ebxml-cppa/schema/cpp-cpa-2_0.xsd

1186

The *NegotiationMessage* element contains the following attributes:

- 1188 • a REQUIRED *businessMsgId* attribute that uniquely identifies the current *Business Message*
- 1189 within the scope of one *Negotiation Dialog*,
- 1190 • a REQUIRED *negotiationDialogId* attribute that uniquely identifies an ongoing *Negotiation*
- 1191 *Dialog* ([See Section 12.1](#)) that connects multiple offer/counter-offer transactions that pertain
- 1192 to the same *CPA Template*,
- 1193 • an IMPLIED *offerId* attribute that uniquely identifies each instance of an offer or counter
- 1194 offer,
- 1195 • an IMPLIED *inresponseTo* attribute that identifies the unique *Business Message* of the
- 1196 previous offer or counter-offer that this *Business Message* is responding to. Its value can be
- 1197 null for the initiating offer of the *Negotiation Dialog*,
- 1198 • a REQUIRED *binding* attribute that indicates whether the current *Message* is legally
- 1199 binding,
- 1200 • a REQUIRED *messageType* attribute that indicates the status of current negotiation. The
- 1201 legal values for the *messageType* attribute are:
- 1202 ♦ "Offer"
- 1203 ♦ "CounterOffer",
- 1204 ♦ "CounterOfferPending"
- 1205 ♦ "Rejected"
- 1206 ♦ "Accepted"
- 1207 ♦ "Expired"
- 1208 ♦ "SinglePartySigned"
- 1209 ♦ "Signed"
- 1210 ♦ "Unsigned"

1211

- 1212 • An IMPLIED **error** attribute that identifies the error code in case of rejection for a
 1213 CounterOffer.
 1214
- 1215 The *NegotiationMessage* element SHALL consist of the following child elements:
 1216 • One REQUIRED *NCPA* element to identify the current *NCPA*
 1217 • One REQUIRED *CPATemplateId* element to identify the *CPA* that is being negotiated.
 1218 • One REQUIRED *cpa:SecurityDetails* element to describe the security requirements of
 1219 current negotiation.
 1220 • One REQUIRED *InitiatingParty* element to describe the initiating *Party* of the current
 1221 *Negotiation Dialog*.
 1222 • One REQUIRED *RespondingParty* element to describe the responding *Party* of the current
 1223 *Negotiation Dialog*.
 1224 • One REQUIRED *BPSSBusinessDocumentName* element that indicates the name of the
 1225 *Business Document* this offer relates to.
 1226 • One REQUIRED *ExpirationDate* element that specifies the date when this offer or counter
 1227 offer expires. Its XML data type is “dateTime”.
 1228 • One REQUIRED *BusinessDocuments* element that describes the *CPA Template*.
 1229 • One REQUIRED *NegotiationContent* element that itemizes accepted, rejected, updated
 1230 elements within the current *CPA*.
 1231 • One IMPLIED *ResponseToURL* element that identifies the return (http) address where the
 1232 responding *Party* can send its response asynchronously. See Section 11.1.10 for more
 1233 information.
 1234 • One IMPLIED *Comment* element that can be used to record free text. For example, if the
 1235 counter offer is a rejection, the *Party* can list its contact information here for a more
 1236 traditional negotiation that will involve humans.

1237 11.1.2 NCPA element

1238 The *NCPA* element contains one REQUIRED *uri* attribute that SHALL have a value that is a
 1239 URI that conforms to [RFC2396] and identifies the location of the *Negotiation CPA* xml instance
 1240 *Document*.

1241 11.1.3 CPATemplateId element

1242 The *CPATemplateId* element contains a REQUIRED *id* attribute and a REQUIRED *version*
 1243 attribute. The definitions of the *id* and *version* attributes are the same as the definitions of the
 1244 *cpaid* and *version* attributes, respectively, of the [ebCPP] *CollaborationProtocolAgreement*
 1245 element.

1246 11.1.4 cpa:SecurityDetails element

1247 The *cpa:SecurityDetails* element is defined in detail in the “*SecurityDetails* element” section of
 1248 [ebCPP].

1249 11.1.5 InitiatingParty element

1250 The *InitiatingParty* element describes the *Party* that initiated the current *Negotiation Dialog*.
 1251 This element contains a REQUIRED *cpa:PartyId* element (per the definition in [ebCPP]), a
 1252 REQUIRED *CPPIId* element, and a *CPPNDD* element (cardinality 0 or 1) that identifies the
 1253 *NDD* that is associated with the initiating *Party*’s *CPP*.

1254
 1255 The **CPPI** element has two attributes:
 1256 • A REQUIRED **id** attribute, which SHALL contain the correct value as specified in [ebCPP]
 1257 for the **cppid** attribute of the **CollaborationProtocolProfile** element in a **CPP** document.
 1258 • A REQUIRED **version** attribute. See [ebCPP] for the definition of the **version** attribute of the
 1259 **CollaborationProtocolProfile** element.

1260
 1261 The **CPPNDD** element contains either a child **BinaryDoc** element or a child **Uri** element. The
 1262 **BinaryDoc** element has the xml type “base64Binary”. Its value is the **NDD Document**. The **Uri**
 1263 element SHALL have a value that is a URI that conforms to [RFC2396] and identifies the
 1264 location of the **NDD XML Document**.

1265 11.1.6 RespondingParty element

1266 The **RespondingParty** element describes the **Party** that the initiating **Party** wishes to establish a
 1267 **CPA** with. This element has the same structure as the **InitiatingParty** element.
 1268 The **RespondingParty** element contains a REQUIRED **cppa:PartyId** element, a REQUIRED
 1269 **CPPI** element (per [ebCPP]), and a **CPPNDD** element (cardinality 0 or 1) that identifies the
 1270 **NDD** that is associated with the **CPP**.

1271
 1272 The **CPPI** element has two attributes:
 1273 • A REQUIRED **id** attribute that SHALL contain the value specified for the **cppid** attribute of
 1274 the **CollaborationProtocolProfile** element in a **CPP** document. See [ebCPP].
 1275 • A REQUIRED **version** attribute. See [ebCPP] for the definition of the **version** attribute of the
 1276 **CollaborationProtocolProfile** element.

1277
 1278 The **CPPNDD** element contains either a child **BinaryDoc** element or a child **Uri** element. The
 1279 **BinaryDoc** element has the xml type “base64Binary”. Its value is the **NDD Document**. The **Uri**
 1280 element SHALL have a value that is a URI that conforms to [RFC2396] and identifies the
 1281 location of the **NDD XML Document**.

Deleted: xml

1282 11.1.7 BPSSBusinessDocumentName element

1283 The **BPSSBusinessDocumentName** element identifies the **Business Document** that is defined in
 1284 the negotiation **BPSS** instance **Document**. The **BPSSBusinessDocumentName** element has one
 1285 REQUIRED **name** attribute. The value of this attribute is the name of the **Business Document**
 1286 and SHALL be one of the following:

- 1287 • “CPA_Offer_Doc”
- 1288 • “CPA_Accept_Offer_Doc”
- 1289 • “CPA_Counter_Pending_Offer_Doc”
- 1290 • “CPA_Counter_Offer_Doc”
- 1291 • “CPA_Reject_Offer_Doc”
- 1292 • “CPA_Final_Doc”
- 1293 • “CPA_Final_Response_Doc”
- 1294 • “CPA_Final_Response_Doc_Signed”
- 1295 • “CPA_Final_Response_Reject_Doc”

1296 11.1.8 BusinessDocuments element

1297 The **BusinessDocuments** element has a **CPATemplateDoc** child element. The **CPATemplateDoc**
1298 element contains the following child elements:

- 1299 • A REQUIRED **NDD** element that identifies the **NDD** associated with the **CPA Template** in
1300 the initial offer. It has either a child **BinaryDoc** element or a child **Uri** element. The
1301 **BinaryDoc** element has the xml type "base64Binary". Its value is the **NDD Document**. The
1302 **Uri** element SHALL have a value that is a URI that conforms to [RFC2396] and identifies
1303 the location of the **NDD xml Document**.
- 1304 • A REQUIRED **CPATemplate** element that has either a **BinaryDoc** child element that has the
1305 type base64Binary and whose value is the proposed **CPA Template Document**, or a **Uri**
1306 element whose value is a URI that conforms to [RFC2396] and references the location of
1307 proposed **CPA Template xml Document**. The **CPATemplate** element is also used to send or
1308 reference the final **CPA** at the end of a **Negotiation Dialog**.

1309
1310 Throughout one negotiation process, the most up-to-date **CPATemplate** SHALL be available for
1311 the other **Party**. Therefore, each **Party** SHALL include either the updated **CPA Template** or a
1312 URI reference to it in each message. The receiving **Party** can reference it while examining the
1313 **NegotiationContent** element, which identifies the delta between the current copy of
1314 **CPATemplate** and the one that was sent out earlier. See Section 12.10 for a definition of
1315 "updated **CPA Template**".

1316 11.1.9 NegotiationContent element

1317 For an initial offer, the **NegotiationContent** element can be empty. If the offering **Party** had
1318 created the initial **CPA Template** and **NDD** by modifying information in the other **Party's CPP**
1319 and **NDD**, the **NegotiationContent** element in the initial offer SHALL describe the changes made
1320 by the offering **Party** to the information in the other **Party's CPP** or **CPA Template** when
1321 forming the **CPA Template** of the initial offer. For counter offers within this **Negotiation Dialog**,
1322 the **NegotiationContent** element SHALL describe the sending **Party's** proposed modifications to
1323 the **CPA Template**.

1324
1325 The **NegotiationContent** element SHALL list all items accepted by the sending **Party** since the
1326 start of the **Negotiation Dialog** (including the ones being accepted by this **Message**). The
1327 **NegotiationContent** element does not contain items accepted by the other **Party** since the start of
1328 the negotiation.

1329
1330 The **NegotiationContent** element SHALL list all items updated, deleted, or inserted by the
1331 sending **Party** since the sending **Party** received the previous offer or counter offer. Any item that
1332 has been deleted by one party can no longer be re-inserted in future counter offers.

1333
1334 It is up to the receiver of this **Message** to decide whether to continue negotiate, accept, or reject
1335 changes listed within the **NegotiationContent** element.

1336
1337 The **NegotiationContent** element contains the following child elements. In the following
1338 information, the Xpath of the item points to the item in the most recent update of the **CPA**
1339 **Template**, i.e. the one that is included in or referenced by this **Message**.

1340

1341 **THE ELEMENT AND ATTRIBUTE NAMES SHOULD BE ADDED TO THE SECOND-**
 1342 **LEVEL BULLETS. THE TERM "ELEMENT" OR "ATTRIBUTE" SHOULD BE ADDED**
 1343 **AFTER EACH ELEMENT OR ATTRIBUTE NAME BELOW.**
 1344

- 1345 • *AcceptedItem*¹(1...n): These are the items that have been accepted by the sending Party
 1346 during all exchanges prior to this Message and within the same Negotiation Dialog.
- 1347 ◆ Xpath of item
- 1348 • *DeletedItem* (0...n)
- 1349 ◆ Xpath of item
- 1350 • *UpdatedItem* (0...n)
- 1351 ◆ Xpath of item
- 1352 ◆ Original value of item
- 1353 ◆ Proposed value of item
- 1354 ◆ The status that indicates update of this item is "Required" or "Preferred".
- 1355 • *InsertedItem* (0...n)
- 1356 ◆ Xpath of item
- 1357 ◆ Proposed value of item
- 1358 ◆ The status that indicates addition of this item is "Required" or "Preferred".

1359 **WE NEED TO EXPLAIN "REQUIRED" AND "PREFERRED".**
 1360
 1361

1362 One use case when *InsertedItem* will be useful is when the responding party wants to add in its
 1363 own *Certificate* element under ebXML CPA/PartyInfo.

1364 <CollaborationProtocolAgreement...>

1365 <PartyInfo...>

1366 <!-- Certificates used by the "Seller" company -->

1367 <tp:Certificate tp:certId="CompanyA_AppCert">

1368 <ds:KeyInfo>

1369 <ds:KeyName>CompanyA_AppCert_Key</ds:KeyName>

1370 </ds:KeyInfo>

1371 </tp:Certificate>

1372 <tp:Certificate tp:certId="CompanyA_SigningCert">

1373 <ds:KeyInfo>

1374 <ds:KeyName>CompanyA_SigningCert_Key</ds:KeyName>

1375 </ds:KeyInfo>

1376 </tp:Certificate>

1377 <!-- Certificates used by the "Buyer" company -->

1378 <tp:Certificate tp:certId="CompanyB_EncryptionCert">

1379 <ds:KeyInfo>

1380 <ds:KeyName>CompanyB_EncryptionCert_Key</ds:KeyName>

1381 </ds:KeyInfo>

1382 </tp:Certificate>

1383 <tp:Certificate tp:certId="CompanyB_ServerCert">

1384 <ds:KeyInfo>

¹ An item can be either an element or an attribute.

```

1385 <ds:KeyName>CompanyB_ServerCert_Key</ds:KeyName>
1386 </ds:KeyInfo>
1387 </tp:Certificate>
1388 :
1389 </PartyInfo...>
1390 :
1391 </CollaborationProtocolAgreement>
    
```

ADDITIONAL DISCUSSION OF INSERTIONS WOULD BE HELPFUL. EXAMPLES ARE:

- **ONE CAN'T INSERT ANYTHING THAT ISN'T MENTIONED IN THE NDD.**
- **INSERTION AND DELETION OF INSTANCES OF A REPEATING ELEMENT.**

Any of the items under the *NegotiationContent* element can be either a leaf node or non-leaf node. A non-leaf node indicates that the entire subtree under that node is subject to the corresponding change action. In other words, the entire subtree has been accepted, deleted, updated, or inserted though for update, not all items within the subtree have necessarily changed. See Figure 3 and Figure 4 for an example of how these elements can be used in a negotiation.

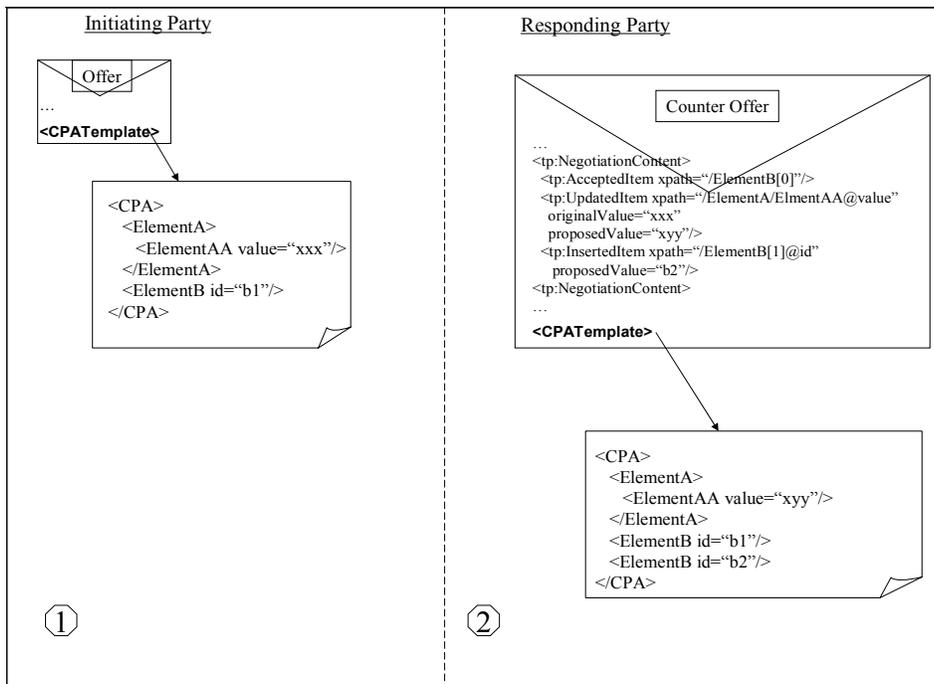
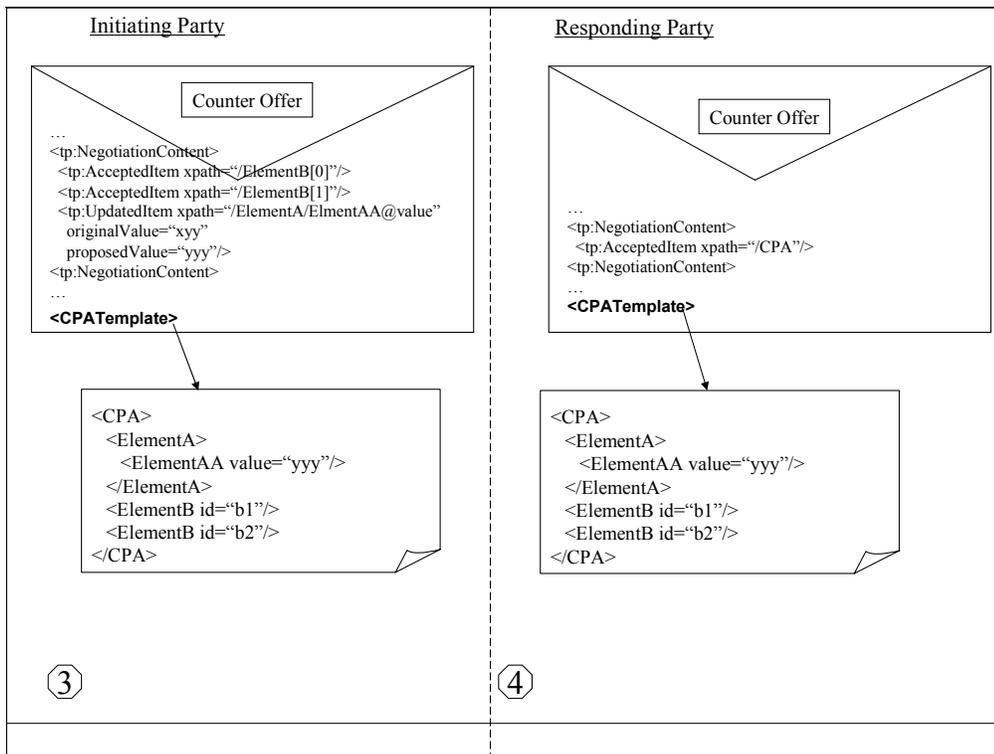


Figure 3. Example of how to use NegotiationContent element - Step 1

In Figure 3, the initiating Party created a CPA that contains one element A and one element B. The responding Party accepted element B, updated the given element A value from 'xxx' to 'xyy', and inserted a second element, B, which has the attribute, id, whose proposed value is

1409 'b2'. The attached/referenced *CPA Template* now reflects the above changes.

1410



1411

1412

Figure 4. Example of how to use the NegotiationContent element - Step 2

1413

1414 In Figure 4, the initiating Party received the changes proposed by the responding Party, made
 1415 the following changes, and sent back another proposal: Accepted newly added element B and
 1416 updated element A/element AA’s attribute value from ‘xyy’ to ‘yyy’. The responding Party
 1417 accepted this change and their corresponding CPA Templates reflected their changes.
 1418

1419 When the NegotiationMessage element has its messageType attribute set to
 1420 “CounterOfferPending”, the counter offer might consist of:

- 1421 • Deleted elements and attributes.
- 1422 • Inserted elements and attributes.
- 1423 • Re-ordered elements using a [XPath]-based list of changes with status of “Required” or
 1424 “Preferred”.
- 1425 • Changed values of elements and attributes.
- 1426 • Rejection: with reason(s) for rejection. See Section 11.4 for additional information. Rejection
 1427 is final. It ends the Negotiation Dialog and the two Parties SHOULD make human to human
 1428 contact to resolve their incompatibilities.

1429 11.1.10 ResponseToURL element

1430 The *ResponseToURL* element SHALL have a value that is a URI that conforms to [RFC2396]
 1431 and identifies the return address where the counter offer can be sent asynchronously. If the
 1432 sender of the *Message* omits the *ResponseToURL* element, the responding *Party* MUST send its
 1433 response *Message* synchronously.

1434 11.1.11 Offer-CounterOffer-Acceptance-Time

1435 The *Offer-CounterOffer-Acceptance-Time* element is an element that used in an offer. It is
 1436 used to specify the time by which a counter offer must be received by the initiating party. It is
 1437 permissible to specify a different acceptance time for each offer or counter offer.

Deleted: by

1438 11.2 CPA ID, Negotiation Dialog ID, Unique Business Message ID, and 1439 InResponseTo

1440 The values of the *id* and *version* attributes of the *CPATemplateId* element SHALL remain the
 1441 same throughout any *Negotiation Dialog*.

1442 The *Negotiation-Dialog Identifier* is used to identify a particular *Negotiation-Dialog* thread. See
 1443 the discussion of the *Negotiation-Dialog Identifier* in Section 12.3. The value of the *id* attribute
 1444 of the *CPATemplateId* element SHALL NOT be used as the *Negotiation-Dialog Identifier*. See
 1445 the discussion in Section 12.3.

1446 The value of the *businessMsgId* attribute of the *NegotiationMessage* element is a unique
 1447 identifier that identifies the current *Business Message* within the scope of one *Negotiation*
 1448 *Dialog*.

1449 The value of the *InResponseTo* element is the unique *Business Message* identifier of the last
 1450 incoming offer or counter-offer *Message* that this current *Message* is responding to.

1454 11.3 Offer and Counter Offer

1455 An offer differs from subsequent counter offers. An offer always contains (or references) the
 1456 *NDD Document* and the initial *CPA Template*. Each counter offer contains or references the
 1457 latest version of the *CPA Template*, containing all changes made up to and including the changes
 1458 accepted prior to this *Message*. The same XML schema defines both the offer and the counter
 1459 offer *Documents*.

1461 11.4 CPA Offer Rejected

1462 When a *CPA* offer or counter offer is rejected, the rejecting party SHALL set the
 1463 *NegotiationMessage* element's *messageType* attribute to "Rejected".

Deleted: should

1464 The process of composing the *CPA* from *CPPs* will detect many error conditions before the
 1465 *Negotiation Process* begins. Others might be discovered during the *Negotiation Process*.
 1466 Examples are mismatched *Process Specification Document* and mismatched delivery-channel
 1467 requirements. See the *CPA-Composition* appendix of [ebCPP] for information about error
 1468 conditions that can be detected during composition of a *CPA Template*.

1471 The rejection *Message* SHALL include reason, contact name, phone, and/or URL for further
 1472 information. The **Comment** element is used for this purpose.

1473

1474 Following are some reasons for rejection. The reason is indicated by the value of **error** attribute
 1475 of **NegotiationMessage** element:

1476

1477 • *CPA Template* contents. Examples:

1478

◆ "ExpiredCPP": Expired *CPP*

1479

◆ "UnableToFulfillSecurityRequirements": Unable to fulfill security requirements

1480

1481 ◆ "ProposedSecurityPolicyInadequate": Proposed security policy is inadequate

1482

1483 ◆ "OurOfSequenceCounterOffer": Out-of-sequence counter offer

1484

◆ "FailedSignatureValidation_CPATemplate": The signature on the *CPA Template* failed validation.

1485

1486 ◆ FailedSignatureValidation_CPA": The signature on the agreed *CPA* failed validation

1487

• The *CPA* is not signed until it is agreed to.

1488

◆ "UnsupportedPackaging": proposed *Packaging* not supported

1489

◆ "UnsupportedSignal": unable to support signals requested (BPSS instance *Document*)

1490

1491 • Business relationship

1492

◆ "UnsupportedBusinessRelationship": *CPA* unsupported without existing *Business* relationship.

1493

1494 • Negotiation Process

1495

◆ "FailedToConverge": In the judgment of the rejecting *Party*, too many counter offers were tried with no forward progress toward convergence.

1496

1497

1498 NOTE: A future version of this specification might formulate a definition of and protocol
 1499 for detecting "no forward progress".

1500

1501 ◆ "PreviouslyRejectedCPA": Proposed *CPA* previously received and not accepted.

1502

1503 • "ExpiredOffer": The current offer's validity interval has expired.

1504

1505 • "FormatError": CPA format problems

1506

◆ Examples: parsing error, data invalid

1507

• "UnknownSystemError": Internal System Error

1508

11.5 CPA Offer accepted

1509 When a *CPA* offer is accepted, the final *CPA Document* contained in or referenced by the
 1510 **CPATemplate** element SHALL be signed if both *Parties' NDDs* indicate they are capable of
 1511 signing the final *Document*.

1512

1513 When a negotiation ends in success, the final document will contain, in effect, a completed *CPA*
1514 document, even though the element name is still *CPATemplate*. The content is a fully qualified
1515 *CPA*. The user should be aware and not be confused by the element name in this case.

1516 **12 Negotiation Protocol**

1517 **12.1 General Principles of Negotiation Protocol**

1518 Figure 2 in Section 5.2 provides a high-level overview of the *Negotiation Process* including the
1519 discovery-related steps and the protocol to negotiate a *CPA*. This section describes the
1520 *Negotiation Protocol* in detail including a description of the negotiation BPSS instance
1521 *Document*.

1522 [NOTE: Although a BPSS instance document is used to document the details of the](#)
1523 [*Negotiation Protocol*, there is no requirement to implement BPSS as part of an](#)
1524 [implementation of this specification. In other words, there is no requirement to deploy](#)
1525 [the *Negotiation* BPSS instance document into the *negotiation* runtime system.](#)
1526

1527
1528 A *Negotiation Dialog* is a complete execution of the BPSS *Negotiation-Protocol* choreography
1529 from the initial offer until the *CPA* is completed successfully or the negotiation fails. A single
1530 *Negotiation Dialog* negotiates a single *CPA*.

1531 **12.2 CPA Identifier**

1532 When a *Party* creates a *CPA Template*, that *Party* SHALL assign a valid value to the *cpaid*
1533 attribute in the *CPA Template*. See Section 9.8 regarding negotiability of the *cpaid* attribute.

1534 **12.3 Negotiation-Dialog Identifier**

1535 A *Negotiation-Dialog Identifier* identifies the *Negotiation Dialog* from initial offer to
1536 completion. Each *Party* SHALL separately maintain the ongoing state information in association
1537 with the *Negotiation-Dialog* identifier. The value of the *Negotiation-Dialog* identifier MUST be
1538 common to the two *Parties* and MUST be unique among all on-going negotiations between a
1539 pair of *Parties*.

1540 The value of the *cpaid* attribute of the *CPA Template* SHALL NOT be used as the value of the
1541 *Negotiation-Dialog Identifier*. The value of the *Negotiation-Dialog Identifier* SHALL be
1542 determined independently of the value of the *cpaid* attribute. The reason is to ensure that if a
1543 negotiation fails and the same *CPA Template*, with the same value of the *cpaid* attribute is used
1544 in a second negotiation attempt, uniqueness of the *Negotiation-Dialog Identifier* is preserved.
1545

1546
1547 NOTE: Although it is not expected that *Negotiation Dialogs* involving the same *CPA*
1548 *Template* will overlap in time, the above rule ensures that saved state information from an
1549 earlier attempt at negotiation can be referenced by its *Negotiation-Dialog Identifier* during a
1550 later attempt with the same *CPA Template*.
1551

1552 **12.4 Offer Identifier**

1553 A counter offer MUST be associated with the offer or counter offer to which it is replying. Each
1554 offer or counter offer SHALL have a unique *Offer Identifier* defined by the negotiation
1555 application. A counter offer states the *Offer Identifier* of the offer or counter offer to which it is
1556 replying. The identifiers and the references to them are in the negotiation-*Message* payload.

1557
1558 The *Offer Identifier* MUST be unique among the initial offer and all counter offers issued by a
1559 given *Party* within a *Negotiation Dialog*. The *Offer Identifier* is qualified by the *Party Id* of the
1560 issuer and the *Negotiation-Dialog Identifier*.

1561
1562 NOTE: With ebXML *Messaging*, the *messageId* and *refToMessageId* attributes in the
1563 *Message* header could serve the purpose of the *Offer Identifier*. However, to enable
1564 alternative *Messaging* protocols, such as “vanilla SOAP”, which do not have these
1565 identifiers, the *Offer Identifier* is defined at the application level.

1566 12.5 Negotiation Status

1567 The *Status* element in the *CPA* records the state of the composition and *Negotiation Protocol*.
1568 The states of its *value* attribute progress as follows:

- 1569 • “Proposed” – This value is in the *CPA Template* sent with the initial offer and remains
1570 unchanged until an agreed *CPA* is completed.
- 1571 • “Agreed” – This value is in the completed *CPA* that is sent from one *Party* to the other for
1572 validation if the *Parties* had agreed not to sign the *CPA*. This is the final state.
- 1573 • “Signed” – If the *Parties* had agreed to sign the *CPA*, the *CPA* sent from one *Party* to the
1574 other *Party* is signed by the sending *Party* and the value of the *value* attribute is “Signed”.
1575 This is the final state.

1576
1577 NOTE: Because the *Status* element is included in the first *Party*’s signature, the value of the
1578 *value* attribute cannot be changed when the second *Party* signs.

1579 12.6 ebXML Conversation

1580 A single *Negotiation Dialog* corresponds to a single ebXML *Conversation*.

1581
1582 For use with *Message* services, such as “vanilla SOAP”, that have no *Conversation* construct, the
1583 *Negotiation-Dialog Identifier* serves the purpose of a *Conversation* identifier at the application
1584 level

1585 12.7 Negotiation CPA

1586 Prior to the initial offer, a *Negotiation CPA* MUST be activated between the two negotiating
1587 *Parties*. See Section 5.1 for a possible scenario.

1588
1589 NOTE: The negotiation subteam plans to simplify the [NCPA](#) to the greatest extent
1590 possible. The goal is to define a default [NCPA](#) that could be built into the *Negotiation* run-
1591 time systems and not have to be explicitly composed, negotiated, or deployed.

1592 12.8 Initial Offer

1593 A *Party* (B) can create and send an initial offer to another *Party* (A) in different ways, depending
1594 on whether *Party* B is starting with *Party* A’s *CPP* or *CPA Template*.

- 1595 • If *Party* B discovered the *CPP* of *Party* A (a potential trading partner), *Party* B composes a
1596 *CPA Template* from its *CPP* and *Party* A’s *CPP*. *Party* B then prepares an *NDD* that
1597 describes what is negotiable in the *CPA Template*. If *Party* A had also published an *NDD*,
1598 *Party* B SHOULD take that *NDD* into account in preparing the *NDD* for the initial offer.
- 1599 • If *Party* B discovered the *CPA Template* and *NDD* of *Party* A, *Party* B modifies the *CPA*

1600 *Template* to include information about itself, makes other modifications to negotiable items
 1601 in the *CPA Template* that are indicated in the *Party A*'s *NDD*, and prepares a new *NDD* to go
 1602 with the modified *CPA Template*.

1603
 1604 In either case, *Party B* is also responsible for inserting into the *CPA Template* the **Start**, **End**, and
 1605 other elements that are present in a *CPA* but not in a *CPP*.

1606
 1607 If *Party B* creates the initial offer by modifying *Party A*'s published *CPP* or *CPA Template*,
 1608 *Party A* SHOULD include a list of changes (Accepted, Deleted, Updated, Inserted) in the initial-
 1609 offer *Message* (Negotiation Content section) in addition to the initial-offer information

1610
 1611 *Party B* then submits the new *CPA Template* and *NDD* to *Party A* as an initial offer.

1612
 1613 It is RECOMMENDED that the *CPA Template* in an initial offer be signed by the offering *Party*.

1614 **12.9 Simultaneous Initial Offers**

1615 Two *Parties* might simultaneously discover each other and send each other initial offers. Since
 1616 the two initial offers will cause creation of two independent *Negotiation Dialogs*, this race
 1617 condition might only be discoverable and resolvable at the application level. Human contact will
 1618 be necessary to decide which *Negotiation Dialog* to proceed with.

1619 **12.10 Offer and Counter Offer**

1620 When a *Party* proposes an offer or counter offer, the details of the offer or counter offer are
 1621 expressed in a negotiation *Message*. The original *NDD* SHALL NOT be altered during the
 1622 course of the negotiation.

1623
 1624 If *Party A* initiates the *Negotiation Dialog* by sending *Party B* an offer, *Party B* sends back a
 1625 counter offer. In order to counter this counter offer, *Party A* sends another counter offer to *Party*
 1626 *B*. In other words, only the initiating *Message* is an offer; the rest of the negotiation will be
 1627 conducted by exchanging counter offers. Each counter offer message contains or references an
 1628 updated version of the *CPA Template* that contains all changes up to, and including, the latest
 1629 changes accepted prior to this *Message*.

1630
 1631 Throughout the *Negotiation Dialog*, each *Party* can terminate the negotiation by sending a
 1632 *Message rejecting an* offer or counter offer without proposing a counter offer. Human to human
 1633 contact is encouraged after a CPA is rejected, in order to resolve any impasse before initiating a
 1634 brand new *Negotiation Dialog*.

Deleted: "CPA offer Rejected" in
responding to an incoming

Deleted: "CPA offer Rejected" is sent

1635
 1636 A counter offer SHALL only refer to items that are listed in the *NDD*. Any offer or counter offer
 1637 that is outside the limits defined in the *NDD* MUST be rejected.

1638
 1639 A counter offer SHALL NOT propose a wholesale change of subject matter. For example a
 1640 counter offer SHALL NOT propose changes in the *Roles* of the participants.

1641
 1642 A *Party* that wishes to propose a different BPSS instance *Document* SHALL reject the received
 1643 offer or counter offer and can then issue its own initial offer including the desired BPSS instance
 1644 *Document*.

1645
1646 A counter offer SHALL NOT introduce a new *NDD*. To introduce a new *NDD*, a *Party* SHALL
1647 reject the received offer or counter offer and can then issue its own initial offer including the
1648 desired *NDD*.

1649
1650 When responding to an offer or counter offer, a *Party* SHALL indicate in its counter offer, which
1651 items in the prior offer or counter offer it accepted.

1652
1653 If a counter offer contains only indications of acceptance of items, the *Party* that sent it is
1654 indicating acceptance of the *CPA* as modified by the prior steps in the negotiation.

1655
1656 Once agreement has been reached on any part of the *CPA*, those elements and attributes SHALL
1657 NOT be reopened for negotiation.

1658

1659 **12.10.1 Responses to Offer and Counter Offer**

1660 A number of responses can be given to an offer or counter offer. The responses fall into the
1661 following categories:

- 1662 • Acceptance: Acceptance of an offer or counter offer means that the *Party* that received the
1663 offer or counter offer is accepting all remaining open items and hence the two *Parties* have
1664 reached agreement.
- 1665 • Counter offer pending: The *Party* that received the offer or counter offer wishes to negotiate
1666 further on some or all open items and is going to send its own counter offer.
- 1667 • Rejection: The *Party* that received the offer or counter offer believes that agreement cannot
1668 be reached. Human contact is REQUIRED in order to resolve the incompatibilities.
- 1669 • The responses are discussed in Section 11.4.

1670 **12.10.2 Offer-Counter Offer Acceptance Time**

1671 A maximum time (interval) for acceptance is associated with each offer or counter offer. The
1672 acceptance interval is a business-level timeout; processing it is independent of any *Document-*
1673 *exchange* or transport-level *Message-loss* recovery rules. When the acceptance interval expires
1674 without a response, the initiator SHALL record the current *Negotiation Dialog* as terminated. [The](#)
1675 [acceptance interval is in the Negotiation Message and can be varied for each Message.](#)

1676

1677 **12.11 Conclusion of Negotiation**

1678 The negotiation concludes when agreement has been reached. This might happen either by one
1679 *Party* accepting the initial offer or following an exchange of counter offers.

1680

1681 If agreement is reached on the initial offer, and the *Party* that received the initial offer does not
1682 have to add any information to the *CPA Template*, the negotiation concludes immediately. The
1683 *Party* that received the initial offer SHALL send a *Message* indicating acceptance and the *CPA*
1684 *Template* becomes the agreed *CPA*. If signing is included in the initial offer, the offering *Party*
1685 SHALL sign the *CPA Template* before sending it. The receiving *Party* SHALL then sign and
1686 return the *CPA*. At this point, the *Parties* are ready to deploy the *CPA* into their run-time systems
1687 and commence business. If the second *Party* does not agree to sign, and signing is negotiable, it

1688 | SHALL respond with a counter offer that excludes signing instead of accepting the initial offer.
 1689 | [See Section 12.12 for a discussion of negotiating signing of the CPA.](#)

1690
 1691 | When agreement has been reached following exchanges of counter offers, the *Party* that received
 1692 | and accepted the final counter offer SHALL send the completed *CPA* (or its URL) to the other
 1693 | *Party* for approval. The receiving *Party* SHALL respond, indicating either approval or rejection.
 1694 | If signing was agreed to, the sending *Party* SHALL sign the *CPA* before sending it. The
 1695 | receiving *Party* SHALL check that the new *CPA* conforms to its understanding of the contents of
 1696 | the *CPA*. The receiving *Party* can also validate the first *Party*'s signature. If the receiving *Party*
 1697 | approves the *CPA*, the receiving *Party* SHALL sign the *CPA* over the first *Party*'s signature and
 1698 | return it to the first *Party*. Otherwise the receiving *Party* SHALL respond indicating rejection.

1699
 1700 | The *Party* that received the completed *CPA* SHALL respond in one of the following ways:

- 1701 | • *Message* indicating that a completed *CPA* was received [and accepted](#) (*BusinessDocument*
 1702 | name = "CPA Final Response Doc")
- 1703 | • *Message* that sends a completed *CPA* signed by the sender (*BusinessDocument* name =
 1704 | "CPA Final Response Doc Signed").
 1705 | ◆ Used when signing was agreed to and the received *CPA* was signed by the sending *Party*.
- 1706 | • *Message* [indicating that a completed CPA was received and rejected](#) (*BusinessDocument*
 1707 | name = "CPA Ffinal Response Reject Doc").

Deleted: A separate indicator in the *Message* distinguishes between accept and reject.¶

1708
 1709 | Following are some reasons for rejecting the received *CPA*:

- 1710 | • The final *CPA* does not agree with the recipient's understanding of the contents of the *CPA*
 1711 | (some kind of state-tracking mismatch).
- 1712 | • The signature on the final *CPA* cannot be validated.
- 1713 | • The final *CPA* was not signed although signing was agreed to.

1714
 1715 | When signing by both *Parties* was agreed to, the *Party* that received the double-signed *CPA*
 1716 | SHALL test for the following conditions:

- 1717 | • The double-signed *CPA* is acceptable.
- 1718 | • The double-signed *CPA* is rejected. Reasons to reject this *CPA* include:
 1719 | ◆ The second signature on the double-signed *CPA* cannot be validated.
 1720 | ◆ An acknowledgment was received when a double-signed *CPA* was expected.

1721
 1722 | Acceptance and rejection of the double-signed *CPA* are indicated by business signals. See
 1723 | Section 12.13.2 for details.

1724
 1725 | Rejection at this stage is a fatal condition and the *Negotiation Dialog* SHALL be terminated. It is
 1726 | RECOMMENDED that the two *Parties* confer to resolve the discrepancy and then renegotiate
 1727 | the *CPA*. If the resolution of the discrepancy was successful, the renegotiation will generally
 1728 | consist of one *Party* sending a new offer that the other *Party* can accept without a counter offer.

1729 | 12.12 Signing the CPA

1730 | [Signing of the completed and agreed-to CPA is an item of negotiation. Refer to \[ebCPP\]](#)
 1731 | [regarding how to sign the CPA.](#) Negotiation of signing is accomplished by negotiating the
 1732 | presence of the *CPA Signature* element and its child *ds:Signature* elements. Following are the

Deleted: Signing the completed CPA proves who signed it ("legal" signing) and provides the usual integrity check on the contents of the CPA. Signing of the completed and agreed-to CPA is an item of negotiation. Refer to [ebCPP] regarding how to sign the CPA.¶

1733 outcomes:

- 1734 • Agree not to sign: The **Signature** element SHALL be omitted from the final *CPA*.
- 1735 • Agree on 2-Party signing: The final *CPA* SHALL contain the **Signature** element with two
- 1736 **ds:Signature** elements.
- 1737 • Agree on 3-Party signing: The final *CPA* SHALL contain the **Signature** element with three
- 1738 **ds:Signature** elements.

1739

1740 It is important to understand that the **ds:Signature** elements MUST be incorporated into the *CPA*

1741 one at a time, as the *Parties* sign. The **Signature** element MUST NOT be inserted into the *CPA*

1742 until the first *Party* signs. If it is incorporated earlier, the *CPA* will fail validation against the

1743 CPPA XML Schema because there will be no child **ds:Signature** elements.

1744

1745 If the *Parties* agreed to third-Party signing, they SHALL obtain the third *Party*'s signature

1746 before commencing to do business under the *CPA*. The means of obtaining the third *Party*'s

1747 signature are not defined in this specification.

1748 **12.13 BPSS Instance Document for Automated Negotiation**

1749 [THIS SECTION IS A COMPLETE REPLACEMENT. PLEASE REVIEW IN FULL.](#)

1750

1751 [COMMENT FROM MAR. 2003 FACE TO FACE MEETING: WE NEED TO DO SOME](#)

1752 [RATIONALIZATION OF THINGS LIKE HOW ELEMENTS ARE ORDERED IN THE](#)

1753 [BPSS INSTANCE DOCUMENT.](#)

1754

1755 The choreography of the *Negotiation Protocol* is defined by an instance *Document* of the

1756 ebXML *Business Process Specification Schema*[ebBPSS]. The BPSS instance *Document* for

1757 automated negotiation is in Appendix D.

1758

1759 [NOTE: Although in \[ebCPP\], the use of a BPSS instance Document to describe](#)

1760 [choreography is not required, this specification depends intimately on the selected](#)

1761 [choreography description. Therefore, the use of a BPSS instance Document is normative](#)

1762 [for this version of this specification.](#)

1763

1764 This BPSS instance *Document* defines the negotiation choreography beginning with an exchange

1765 of an offer and response.

- 1766 • If the response to the offer is [to accept the offer](#), the choreography transitions to the final
- 1767 *CPA* exchange (see below).
- 1768 • If the response to the offer is [to reject the offer](#), the choreography immediately concludes.
- 1769 • If the response to the offer is [that a counter offer is pending](#), the choreography then goes into
- 1770 an alternation of counter offer and response between the two *Parties* which continues until:
- 1771 ♦ [Acceptance of the offer](#) causes the choreography to transition to the final *CPA* exchange.
- 1772 ♦ [Rejection of the offer](#) concludes the choreography.

Deleted: “
Deleted: ”
Deleted: “
Deleted: ”
Deleted: “
Deleted: ”
Deleted: A response of “accept offer”
Deleted: A response of “r
Deleted: ”

1773

1774 Several *Business Document* names are defined directly under the **ProcessSpecification** element

1775 and referenced in various places as described below.

1776

1777 The BPSS instance *Document* defines initiator and responder *Role* names for each binary

1778 collaboration, collaboration activity, and binary transaction activity. For simplicity in the
 1779 explanation below, in most cases, the terms “initiator” and “responder” are used. For each stage
 1780 of the choreography, the *NCPA* associates *Role* names with actual *Parties* in the *Action* elements
 1781 under the *CollaborationRole* elements.

1782 **12.13.1 Offer-Counter-Offer Choreography**

1783 A counter offer is a requesting *Document* in a new *Business Transaction*, not a response to an
 1784 offer. To issue a counter offer, the recipient of an offer SHALL send a reply that indicates that a
 1785 counter offer is pending and then issue the counter offer as a new *Business Transaction*. This
 1786 avoids a race condition with respect to which *Party* sends the next *Message*. It also avoids any
 1787 need to for the two *Parties* to switch *Roles*.

Deleted: “
 Deleted: pending

1788 The choreography begins with the “CPA_Offer_BTA” *BusinessTransactionActivity* element
 1789 under the “CPA_Negotiation_BC” *BinaryCollaboration* element. A *CPA* offer *Message* is sent
 1790 from the “CPA_Negotiation_Initiator_Role” *Party* to the “CPA_Negotiation_Responder_Role”
 1791 *Party* by means of the “CPA_Offer_BT” *Business Transaction*. The “CPA Offer BT ReqBA”
 1792 *RequestingBusinessActivity* sends the “CPA_Offer_Doc” *Message* from the initiator *Party* to
 1793 the responder *Party*. The “CPA_Offer_BT_RespBA” *RespondingBusinessActivity* then sends
 1794 the response *Message* from the responder *Party* to the initiator *Party*. This *Message* is then
 1795 evaluated as defined by the *Failure* and *Transition* elements under the “CPA_Negotiation_BC”
 1796 binary collaboration. These are the elements whose *fromBusinessState* attribute has the value
 1797 “CPA Offer BTA”. The value of the *expression* attribute in each of these elements is the name of
 1798 the response *Message*, as follows:

Deleted: ¶
 THE FOLLOWING DISCUSSION HAS
 TO BE UPDATED TO CORRESPOND
 TO CHANGES IN THE BPSS
 INSTANCE DOCUMENT THAT
 WERE MADE SINCE THIS SECTION
 WAS WRITTEN.¶

Deleted: ”
 Deleted: -
 Deleted: CPA_Offer_ReqBA
 Deleted: Success,
 Deleted: ,

- 1800 • Transition element:
 - 1801 ♦ If the response Message is “CPA Accept Offer Doc”, the proposed CPA has been
 1802 accepted by the responder Party and the choreography transitions to the final CPA
 1803 exchange. The toBusinessState attribute of the Transition element identifies “CPA Final
 1804 BTA” (the name of the BusinessTransactionActivity element) as the next state in the
 1805 choreography.
 - 1806 ♦ If the response Message is “CPA Counter Pending Offer Doc”, the responder Party will
 1807 send a counter offer as the next Business Transaction. The toBusinessState attribute of
 1808 the Transition element identifies “CPA Counter Offer CA” (the name of the
 1809 CollaborationActivity element) as the next state in the choreography.
- 1810 • *Failure* element: If the response *Message* is “CPA Reject Offer Doc”, the proposed *CPA* has
 1811 been unconditionally rejected by the responder *Party* and the choreography concludes.

Deleted: Success

1813 If the response *Message* to the “CPA_Offer_Doc” *Message* was “CPA Counter Pending Offer
 1814 Doc”, the transition described above takes place and takes the choreography to state “CPA
 1815 Counter Offer CA”, i.e. to the *CollaborationActivity* element named “CPA Counter Offer CA”.
 1816 This *CollaborationActivity* element references the “CPA_Negotiation_CounterOfferBC”
 1817 *BinaryCollaboration* element.

Deleted: Transition element: If the
 response Message is “CPA Counter
 Pending Offer Doc”, the responder Party
 will send a counter offer as the next
 Business Transaction. The
 toBusinessState attribute of the
 Transition element identifies “CPA
 Counter Offer CA” (the name of the
 CollaborationActivity element) as the
 next state in the choreography.¶

1819 The initial request *Message* is under the “CPA_Counter_Offer_1_BTA”
 1820 *BusinessTransactionActivity* element, which is the “from” state for the following. The *Party*
 1821 which received the original *CPA* offer is now the initiator in this *BusinessTransactionActivity*.
 1822 The *Message* is sent from the “CPA_Negotiation_CounterOfferInitiator_Role” *Role* to the

1823 “CPA_Negotiation_CounterOfferResponder_Role” *Role* by means of the
 1824 “CPA_Counter_Offer_BT” **BusinessTransaction**. The *Message*, “CPA_Counter_Offer_Doc”, is
 1825 identified in the “CPA_Counter_Offer_BT_ReqBA” **RequestingBusinessActivity**. The response
 1826 *Message* is sent by means of the “CPA_Counter_Offer_BT_RespBA”
 1827 **RespondingBusinessActivity**. One of three response *Messages* can be sent, as discussed below.
 1828 The response *Message* is then evaluated as defined by the **Failure** and **Transition** elements
 1829 under the “CPA_Negotiation_CounterOfferBC” **BinaryCollaboration**.

Deleted: Success,
 Deleted: ,
 Deleted: Success

- 1830 • **Transition** element:
 - 1831 ♦ If the response *Message* is “CPA Accept Offer Doc”, the proposed *CPA* has been
 - 1832 accepted by the responder *Party* and the choreography transitions to the final *CPA*
 - 1833 exchange. The **toBusinessState** attribute of the **Transition** element identifies the “CPA
 - 1834 Final BTA Init Responder” *Business Transaction* Activity as the next state in the
 - 1835 choreography.
 - 1836 ♦ If the response *Message* is “CPA Counter Pending Offer Doc”, the responder *Party* will
 - 1837 send a counter offer as the next *Business Transaction*. The **toBusinessState** attribute of
 - 1838 the **Transition** element identifies the “CPA Counter Offer 2 BTA” *Business Transaction*
 - 1839 Activity as the next state in the choreography.
- 1840 • **Failure** element: If the response *Message* is “CPA Reject Offer Doc”, the proposed *CPA* has
- 1841 been unconditionally rejected by the responder *Party* and the choreography concludes.

1842 If the above transition takes place, it means that the *Party* that was the responder now becomes
 1843 the initiator to supply a counter offer to the counter offer. The “CPA_Counter_Offer_2_BTA”
 1844 **BusinessTransactionActivity** is now performed in the same manner as the
 1845 “CPA_Counter_Offer_1_BTA” **BusinessTransactionActivity**, described above. Both the
 1846 **BusinessTransactionActivity** “CPA Counter Offer 1 BT” and “CPA Counter Offer 2 BTA”
 1847 use the same **BusinessTransaction** “CPA Counter Offer BT”.

Deleted: <#>Transition element: If the response Message is “CPA Counter Pending Offer Doc”, the responder Party will send a counter offer as the next Business Transaction. The toBusinessState attribute of the Transition element identifies the “CPA Counter Offer 2 BTA” Business Transaction Activity as the next state in the choreography.¶
 <#>¶

1850 The choreography then iterates between the “CPA_Counter_Offer_1_BTA”
 1851 **BusinessTransactionActivity** and the “CPA_Counter_Offer_2_BTA”
 1852 **BusinessTransactionActivity** until success or failure is achieved. If the response Message is
 1853 “CPA Accept Offer Doc”, the choreography then transitions to the final CPA exchange. These
 1854 transitions are indicated by a transition to “CPA Final BTA Init Responder” if the “CPA Accept
 1855 Offer Doc” message is the response message from BusinessTransactionActivity “CPA Counter
 1856 Offer 1 BTA” and “CPA Final BTA Init Initiator” if the “CPA Accept Offer Doc” is the
 1857 response message from BusinessTransactionActivity “CPA Counter Offer 2 BTA”. Failure
 1858 ends the choreography.

Deleted: CPA
 Deleted: Counter
 Deleted: Offer
 Deleted: 2
 Deleted: CPA
 Deleted: Counter
 Deleted: Offer
 Deleted: 2
 Deleted: a
 Deleted: Success causes the
 Deleted: to
 Deleted: se
 Inserted: se
 Deleted: are
 Inserted: are indicated by
 Deleted: s
 Deleted: ”
 Deleted: Init Responder

1859 **12.13.2 Final CPA exchange**

1860 When either the initial offer or a counter offer is accepted in full, the choreography transitions to
 1861 the “CPA Final BT” **Business Transaction**. The purpose of this *Business Transaction* is for the
 1862 *Party* that accepted the offer or counter offer to send the completed *CPA* to the other *Party* and
 1863 for the Responding *Party* of this **Business Transaction** to either accept or reject the final *CPA*. If
 1864 required, the final CPA sent by the Party that accepted the offer can be signed and the response
 1865 message can include a double signed CPA.

1866 If the initial offer was accepted, the next *Business* state is the “CPA Final BTA”

1868 *BusinessTransactionActivity*, which references the “CPA Final BT” *Business Transaction*. The
1869 initiator *Role* “CPA Negotiation Responder” for the “Final CPA BT ReqBA” requesting
1870 *Business* activity sends the “CPA Final Doc” *Message*, containing the CPA or its URL, to the
1871 other (responder) *Party* “CPA Negotiation Initiator” for this *Business Transaction*. “CPA Final
1872 *Doc*” can include a signed CPA if this was negotiated. The responder *Party* checks the CPA and
1873 performs the responding *Business* activity conveying one of:

Deleted: Party
Deleted: Req BA

- 1874 • The “CPA Final Response Doc” *Message*, to acknowledge receipt of an acceptable CPA.
- 1875 • The “CPA Final Response Reject Doc” *Message*, to acknowledge receipt of an unacceptable
1876 CPA and reject the final proposed CPA. One of the reasons for rejecting the final proposed
1877 CPA could be that the signature on the proposed CPA may not be valid.
- 1878 • The “CPA Final Response Doc Signed” *Message*, to acknowledge receipt of an acceptable
1879 signed CPA and return that CPA with the responder *Party*’s signature over the initiator
1880 *Party*’s signature.

Deleted: or rejection. Acceptance and rejection are indicated by values of the status indicator in the *Negotiation Message*
Deleted: THE ABOVE SENTENCE NEEDS TO BE REVISED TO USE THE CORRECT NAME OF THE STATUS ELEMENT OR ATTRIBUTE WHEN THE MESSAGE SCHEMA IS COMPLETED.¶

1881
1882 The response *Message* is then evaluated as defined by the *Success* and *Failure* elements under
1883 the “CPA Negotiation BC” *BinaryCollaboration*.

- 1884 • *Success* element:
1885 ♦ If the response *Message* is “CPA Final Response Doc”, the final CPA has been accepted
1886 by the responder *Party* and the choreography concludes.
1887 ♦ If the response *Message* is “CPA Final Response Doc Signed”, the final CPA has been
1888 accepted by the responder *Party* and the choreography concludes. Please refer to the
1889 discussion below on how the choreography could conclude with a failure because of
1890 failure to verify the double-signed CPA by using a process level *Exception*.
- 1891 • *Failure* element: If the response *Message* is “CPA Final Response Reject Doc”, the final
1892 CPA has been rejected by the responder *Party* and the choreography concludes.

Deleted: Success element:
Deleted: note
Deleted: F

1894 If the counter offer was accepted, the next *Business* state is the “CPA Final BTA INIT Initiator”
1895 or “CPA Final BTA Init Responder” *BusinessTransactionActivity* depending on the state of
1896 “CPA Negotiation CounterOfferBC”. These *BusinessTransactionActivity* elements reference
1897 the “CPA Final BT” *BusinessTransaction*. The sequence of request and response messages is
1898 similar to *BusinessTransactionActivity* “CPA Final BTA” as described above.

Deleted: a
Inserted: activity depending on the state of “CPA_Negotiation_CounterOfferBC”. These *BusinessTransactionActivity*

1899
1900 The response *Message* is then evaluated as defined by the *Success* and *Failure* elements under
1901 the “CPA Negotiation Counter OfferBC” *BinaryCollaboration*.

Deleted: S
Inserted: Sequence of request and response messages is similar to *BusinessTransactionActivity* “CPA Final BTA” as described above.¶
¶ The response *Message* is then evaluated as defined by the *Success* and *Failure* elements under the “CPA_Negotiation_Counter OfferBC” *BinaryCollaboration*. ¶
Success element:

- 1902 • *Success* element:
1903 ♦ If the response *Message* is “CPA Final Response Doc”, the final CPA has been accepted
1904 by the responder *Party* and the choreography concludes.
1905 ♦ If the response *Message* is “CPA Final Response Doc Signed”, the final CPA has been
1906 accepted by the responder *Party* and the choreography concludes. Please refer to the
1907 discussion below on how choreography could conclude with a Failure because of failure
1908 to verify the double signed CPA by using a process level *Exception*.
- 1909 • *Failure* element: If the response *Message* is “CPA Final Response Reject Doc”, the final
1910 CPA has been rejected by the responder *Party* and the choreography concludes.

Deleted: Success element:
Inserted: Success element:
Deleted: note
Inserted: note below on how choreography could conclude with a Failure because of failure to verify the double signed CPA by using a process level *Exception*.¶

1911
1912 The *Party* that receives the final (double signed) CPA SHOULD test it for possible error
1913 conditions as described in Section 12.11. The *Party* that received the double-signed CPA

1914 SHALL reply with the AcceptanceAcknowledgment business signal if the CPA is acceptable or
1915 with the Exception business signal if the CPA is not acceptable. These signals are instance
1916 Documents of the business signals defined in [ebBPSS].

Deleted: Examples of these instance Documents are in Appendix E.

1918 If a counter offer was accepted in full, the choreography transitions to the "CPA Final BTA Init
1919 Initiator" BusinessTransactionActivity. That BusinessTransactionActivity references the "CPA
1920 Final BT" BusinessTransaction and proceeds as for acceptance of an initial offer.

Deleted:
Deleted: a
Deleted:
Deleted: a
Deleted:

1922 **12.13.3 Negotiation Business Signals**

1924 HIMA MUKKAMALA SUGGESTED CREATING EXAMPLES OF SIGNALS THAT SHOW
1925 HOW SOMEONE COULD SEND A NEGATIVE ACCEPTANCE ACKNOWLEDGMENT
1926 TO DECLINE A DOUBLE SIGNED CPA.

1928 **12.13.4 State Diagrams**

1929 The choreography is illustrated by the state diagram shown in Figure 5 and Figure 6.

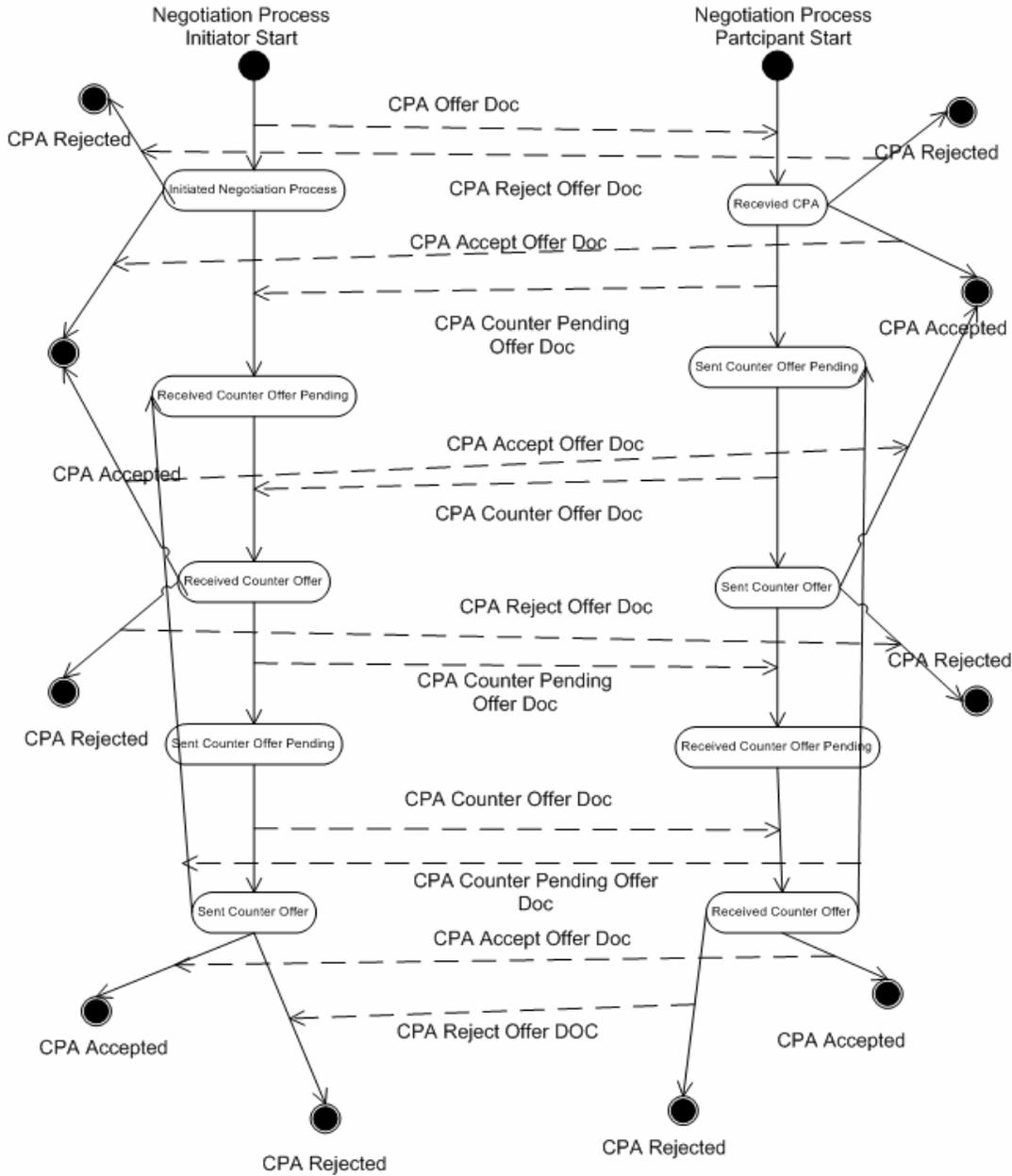
Deleted: <#>Negotiation Business Signals¶
¶
ADD DISCUSSION OF THE CONTENTS OF THE BUSINESS SIGNALS.¶
¶
IS IT FEASIBLE TO CONSTRUCT EXAMPLES OF INSTANCE DOCUMENTS OF THE BUSINESS SIGNALS THAT ARE SPECIFIED IN THE NEGOTIATION BPSS INSTANCE DOCUMENT?¶

1931 THE NEW VERSION OF THE STATE DIAGRAM HAS A PROBLEM. IF THE
1932 DIAGRAMS ARE PASTED INTO THIS WORD DOCUMENT DIRECTLY FROM THE
1933 SOURCE HTM FILE, THEY CANNOT BE RENDERED PROPERLY BY AROBAT
1934 ALTHOUGH THEY ARE RENDERED PROPERLY BY MICROSOFT WORD. TO
1935 OVERCOME THIS PROBLEM FOR THE TIME BEING, THE FOLLOWING
1936 PROCEDURE WAS USED.

- 1937 • OPEN THE SOURCE HTM FILE IN IE.
- 1938 • ENLARGE THE WINDOW A BIT TO MAKE THE SMALL LABELS LEGIBLE.
- 1939 • PRINT SCREEN.
- 1940 • PASTE INTO PAINT.
- 1941 • SELECT THE DIAGRAM WITHIN THE PAINT WINDOW.
- 1942 • COPY, PASTE SPECIAL INTO MICROSOFT WORD AS A BITMAP IMAGE OBJECT.

Deleted: THE STATE DIAGRAMS NEED SOME CORRECTIONS:¶
- IN Error! Reference source not found., THE DOCUMENT NAMES IN THE CENTER DO NOT ALWAYS AGREE WITH THE NAMES IN THE INSTANCE DOCUMENT. ¶
- IN Error! Reference source not found.:¶
+ UNLIKE Error! Reference source not found., THE DOCUMENT NAMES ARE NOT USED.¶
THE RETURN OF THE DOUBLE-SIGNED CPA IS NOT SHOWN. PRESUMABLY + IT IS ANOTHER OUTPUT FROM THE "RECEIVED FINAL CPA" STATE.¶

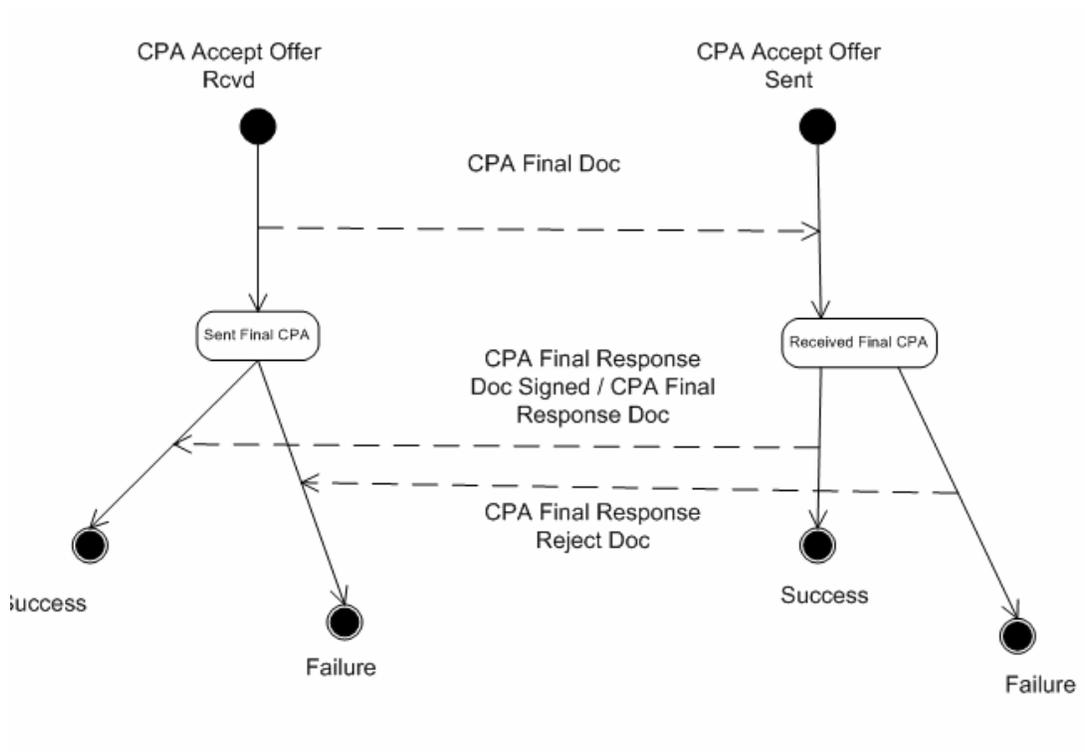
1943



1944

Figure 5. State Diagram for Initial Offer and Counter Offers

1945



1946
1947

Figure 6. State Diagram for Final Transaction

1948 **13 Negotiation Algorithm**

1949 The negotiation algorithm is an application (*Business Process*). It is embodied in the private
1950 process at each *Party*. Note that the BPSS instance *Document* describes only the choreography of
1951 the *Message* exchanges and not the private processes. This section discusses the normative
1952 aspects of negotiation algorithms, i.e. the rules that ensure interoperability between two *Parties*'
1953 implementations of the negotiation algorithm.

1954
1955 NOTE: The negotiation algorithm is out of scope for version 1 of this specification. This
1956 section provides a brief introduction and serves as a place holder for material that might
1957 be introduced in future versions.

1958
1959 Historically, research on negotiation has categorized negotiations as follows:

- 1960 1. Simple matchmaking: The subject (set of negotiable parameters) is static and the
1961 ontology is clear. The two *Parties* have a common understanding of the meanings,
1962 values, and interdependencies of the negotiable parameters. The utility functions are
1963 binary (acceptable *vs.* not acceptable). Negotiation in these situations can be easily
1964 automated.
- 1965 2. Negotiations on static subjects: This is similar to (1) except that the utility functions are
1966 more complex (more than 2 choices, numerical values, etc.). These situations can be
1967 automated but might require human intervention.
- 1968 3. Negotiations on dynamic subjects: Here, the negotiable parameter set can be expanded
1969 during the process of negotiation and the parameters are more likely to interact than in (1)
1970 and (2). Dynamic subject negotiations are much more difficult to automate.

1971
1972 Negotiation of a *CPA* is mostly category 1 with some amount of category 2. On the other hand,
1973 business-level negotiations involve much more complex subject matter with parameters that are
1974 very likely to interact (consider price and delivery time). Therefore, these negotiations can be
1975 expected to be a mixture of categories 2 and 3.

1976 **13.1 CPPs and NDDs**

1977 It is RECOMMENDED that the negotiation algorithms refer to the *Parties*' original *NDDs* (if
1978 available) that go with the *CPPs* as well as the *CPA Template* and its *NDD* to assist in evaluating
1979 offers and counter offers. The original *NDDs* might contain information, such as a *Party*'s
1980 original preference ordering and other constraints that might have been lost when the composite
1981 *NDD* was composed for the *CPA Template*.

1982 **14 References**1983 ***VERSION NUMBERS AND URLS TBD.***1984 ***CHECK THE URLS THAT ARE HERE FOR CORRECTNESS.***

1985

1986 [bpPATT] ebXML E-Commerce Patterns, version 1.0,

1987 <http://www.ebxml.org/specs/bpPATT.pdf>

1988

1989 [ebBPSS] ebXML Business Process Specification Schema

1990

1991 [ebCPP] ebXML Collaboration-Protocol Profile and Agreement Specification, version 2.0.

1992

1993 [ebMS] ebXML Message Service Specification, version 2.0.

1994

1995 [ebRS] ebXML Registry Services Specification

1996

1997 [RFC2119] Key Words for use in RFCs to indicate Requirement Levels, Internet Engineering

1998 Task Force RFC 2119, <http://www.ietf.org/rfc/rfc2119.txt>

1999

2000 [RFC2396] Uniform Resource Identifiers (URI): General Syntax, Internet Engineering Task

2001 Force RFC 2396, <http://www.ietf.org/rfc/rfc2396.txt>

2002

2003 [SOAPATTACH] SOAP Messages with Attachments, John J. Barton, Hewlett Packard Labs;

2004 Satish Thatte and Henrik Frystyk Nielsen, Microsoft, Published Oct 09 2000.

2005 <http://www.w3.org/TR/2000/NOTE-SOAP-attachments-20001211>

2006

2007 [XML] Extensible Markup Language (XML), World Wide Web Consortium,

2008 <http://www.w3.org/XML>.

2009

2010 [XMLDSIG] XML Signature Syntax and Processing, Worldwide Web Consortium,

2011 <http://www.w3.org/TR/xmlsig-core/>

2012

2013 [XMLENC] XML Encryption Syntax and Processing, Worldwide Web Consortium,

2014 <http://www.w3.org/TR/2002/CR-xmlenc-core-20020304/>

2015

2016 [XMLNS] Namespaces in XML, Worldwide Web Consortium,

2017 <http://www.w3.org/TR/REC-xml-names/>

2018

2019 [XPATH] XML Path Language (XPath) Version 1.0,

2020 <http://www.w3.org/TR/xpath>

2021

2022 **15 Conformance**

2023 In order to conform to this specification, an implementation:

- 2024 a) SHALL support all the functional and interface requirements defined in this specification,
2025 b) SHALL NOT specify any requirements that would contradict or cause non-conformance
2026 to this specification.

2027
2028 A conforming implementation SHALL satisfy the conformance requirements of the applicable
2029 parts of this specification.

2030
2031 The objective of conformance testing is to determine whether an implementation being tested
2032 conforms to the requirements stated in this specification. Conformance testing enables vendors to
2033 implement compatible and interoperable systems. Implementations and applications SHALL be
2034 tested using available test suites to verify their conformance to this specification.

2035
2036 Publicly available test suites from vendor neutral organizations such as OASIS and the U.S.A.
2037 National Institute of Science and Technology (NIST) SHOULD be used to verify the
2038 conformance of implementations, applications, and components claiming conformance to this
2039 specification. Open-source reference implementations might be available to allow vendors to test
2040 their products for interface compatibility, conformance, and interoperability.

2041 **15.1 NDD and Negotiation Messages**

2042 An implementation of a tool or service that creates or maintains ebXML instance *Documents* of
2043 the *NDD* and *Negotiation Messages* SHALL be determined to be conformant by validation of the
2044 instance *Documents*, created or modified by said tool or service, against the XML
2045 Schema[XMLSCHEMA-1] definition of these *Documents* in Appendix A and Appendix B,
2046 respectively, and available from

2047
2048 ***URLS TO BE SUPPLIED***

2049
2050 by using two or more validating XML Schema parsers that conform to the W3C XML Schema
2051 specifications[XMLSCHEMA-1, XMLSCHEMA-2].

2052 **15.2 NCPA Instance Document**

2053 An implementation of a tool or service that creates or maintains *NCPA* instance *Documents*
2054 SHALL be determined to be conformant by validation of the *NCPA* instance *Documents*, created
2055 or modified by said tool or service, against the XML Schema[XMLSCHEMA-1] definition of
2056 the *CPA* in [ebCPP]and available from

2057
2058 http://www.oasis-open.org/committees/ebxml-cppa/schema/cpp-cpa-2_0.xsd

2059
2060 by using two or more validating XML Schema parsers that conform to the W3C XML Schema
2061 specifications[XMLSCHEMA-1, XMLSCHEMA-2].

2062 **15.3 Negotiation BPSS Instance Document**

2063 An implementation of a tool or service that creates or maintains negotiation BPSS instance

Automated Negotiation Specification

Page 58 of 97

Copyright © OASIS, 2003. All Rights Reserved

Negotiation.spec.01Nov03.doc

11/2/2003 4:05 PM

2064 *Documents* SHALL be determined to be conformant by validation of the BPSS instance
2065 *Documents*, created or modified by said tool or service, against the XML
2066 Schema[XMLSCHEMA-1] definition of the BPSS in available from

2067

2068 ***URL TO BE SUPPLIED.***

2069

2070 by using two or more validating XML Schema parsers that conform to the W3C XML Schema
2071 specifications[XMLSCHEMA-1, XMLSCHEMA-2].

2072 **15.4 Negotiation Business Signals**

2073 An implementation of a tool or service that creates or maintains negotiation *Business*-signal
2074 instance *Documents* SHALL be determined to be conformant by validation of the *Business*-
2075 signal instance *Documents*, created or modified by said tool or service, against the XML
2076 Schema[XMLSCHEMA-1] definition of the *Business* signals and available from

2077

2078 ***URL TO BE SUPPLIED.***

2079

2080 by using two or more validating XML Schema parsers that conform to the W3C XML Schema
2081 specifications[XMLSCHEMA-1, XMLSCHEMA-2].

2082 **16 Disclaimer**

2083 The views and specification expressed in this document are those of the authors and are not
2084 necessarily those of their employers. The authors and their employers specifically disclaim
2085 responsibility for any problems arising from correct or incorrect implementation or use of this
2086 design.

2087 **17 Contact Information**

2088 | Martin W. Sachs (sub team lead)
2089 | Cyclone Commerce
2090 | 8388 E. Hartford Drive
2091 | Scottsdale, AZ 85255
2092 | USA
2093 | Phone: 203-226-0524
2094 | email: msachs@cyclonecommerce.com

2095 | Dale W. Moberg (author)
2096 | Cyclone Commerce
2097 | 8388 E. Hartford Drive
2098 | Scottsdale, AZ 85255
2099 | USA
2100 | Phone: 480-627-2648
2101 | email: <mailto:dmoberg@cyclonecommerce.com>
2102

Deleted: OTHERS TBD¶

2103 **Notices**

2104 | Portions of this document are copyright © 2002 OASIS.

2105

Deleted: NEED TO DETERMINE OF UN/CEFACT HAS TO BE MENTIONED.

Copyright (C) The Organization for the Advancement of Structured Information Standards [OASIS] 2002. 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 OASIS, except as needed for the purpose of developing OASIS specifications, in which case the procedures for copyrights defined in the OASIS 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 OASIS or its successors or assigns.

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

2106
2107

2108 OASIS takes no position regarding the validity or scope of any intellectual property or other
2109 rights that might be claimed to pertain to the implementation or use of the technology described
2110 in this document or the extent to which any license under such rights might or might not be
2111 available; neither does it represent that it has made any effort to identify any such rights.

2112 Information on OASIS's procedures with respect to rights in OASIS specifications can be found
2113 at the OASIS website. Copies of claims of rights made available for publication and any
2114 assurances of licenses to be made available, or the result of an attempt made to obtain a general
2115 license or permission for the use of such proprietary rights by implementers or users of this
2116 specification, can be obtained from the OASIS Executive Director.

2117

2118 OASIS invites any interested party to bring to its attention any copyrights, patents or patent
2119 applications, or other proprietary rights which may cover technology that may be required to
2120 implement this specification. Please address the information to the OASIS Executive Director.

2121

2122 OASIS has been notified of intellectual property rights claimed in regard to some or all of the
2123 contents of this specification. For more information consult the online list of claimed rights.

2124 **Appendix A XML Schema for Negotiation Descriptor**
 2125 **Document (Normative)**

2126 The XML Schema document for the *NDD* is available as a text file at:

2127

2128 ***FILL IN THE URLS OF THE XML DOCUMENTS IN ALL THE APPENDICES.***

2129

2130 **[THIS APPENDIX HAS BEEN COMPLETELY REPLACED.](#)**

2131

2132

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

```
<!-- edited with XML Spy v4.4 U (http://www.xmlspy.com) by neelakantan kartha (Sterling Commerce) -->
```

```
<schema targetNamespace="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpp-cpa-negot-2_0.xsd"
```

```
xmlns:tns="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpp-cpa-negot-2_0.xsd"
```

```
xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns="http://www.w3.org/2001/XMLSchema"
```

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

```
  <element name="NegotiationDescriptor">
```

```
    <annotation>
```

```
      <documentation>This is the schema representing the NDD</documentation>
```

```
    </annotation>
```

```
    <complexType>
```

```
      <sequence>
```

```
        <element name="NegotiableInformationItem" maxOccurs="unbounded">
```

```
          <complexType>
```

```
            <sequence>
```

```
              <choice>
```

```
                <element name="Value" type="tns:ValueType"/>
```

```
                <element name="UnOrderedValue" type="tns:EnumeratedValues"/>
```

```
                <element name="OrderedValue" type="tns:OrderedEnumeratedValues"/>
```

```
                <element name="ValueWithPreferenceMeasure"
```

```
type="tns:ValueWithPreferenceMeasureType"/>
```

```
                <element name="PresentOrNot" type="tns:PresentOrNotType"/>
```

```
                <element name="IntegerValues" type="tns:IntegerValuesType"/>
```

```
                <element name="Preference">
```

```
                  <complexType>
```

```
                    <attribute name="value" type="xs:integer"/>
```

```
                  </complexType>
```

```
                </element>
```

```
                <element name="Cardinality" type="tns:IntegerValuesType"/>
```

```
                <element name="BooleanValue" type="tns:BooleanValuesType"/>
```

```
                <element name="DurationWithPreference" type="tns:DurationWithPreferenceType"/>
```

```
              </choice>
```

```
            </sequence>
```

```
          <attribute name="xpath" type="xs:string" use="required"/>
```

```
        </complexType>
```

```
      </element>
```

```
    </sequence>
```

```
  <attribute name="documentLocation" type="xs:anyURI" use="required"/>
```

```
</complexType>
```

```
</element>
```

```
<!-- TYPE DEFINITIONS THAT ARE USED IN THE DEFINITIONS ABOVE -->
```

```
<complexType name="ValueType">
```

```
  <attribute name="mustBeFilledIn" use="optional">
```

```
    <simpleType name="fillType">
```

```
      <restriction base="xs:string">
```

```
        <enumeration value="Yes"/>
```

```
        <enumeration value="No"/>
```

```
        <enumeration value="Preferred"/>
```

```
      </restriction>
```

```
    </simpleType>
```

```
  </attribute>
```

2182

Automated Negotiation Specification

Page 63 of 97

Copyright © OASIS, 2003. All Rights Reserved

Negotiation.spec.01Nov03.doc

11/2/2003 4:05 PM

```

2183 </complexType>
2184 <complexType name="EnumeratedValues">
2185 <sequence>
2186 <element name="PresentOrNot" type="tns:PresentOrNotType" minOccurs="0"/>
2187 <element name="Value" type="xs:string" maxOccurs="unbounded"/>
2188 </sequence>
2189 </complexType>
2190 <!--For also stating that the enumerated values have some order associated with them-->
2191 <complexType name="OrderedEnumeratedValues">
2192 <complexContent>
2193 <extension base="tns:EnumeratedValues">
2194 <attribute name="preference" use="optional">
2195 <simpleType name="orderName">
2196 <restriction base="xs:string">
2197 <enumeration value="EarlierPreferred"/>
2198 <enumeration value="LaterPreferred"/>
2199 </restriction>
2200 </simpleType>
2201 </attribute>
2202 </extension>
2203 </complexContent>
2204 </complexType>
2205 <!--For giving the endpoints for elements like Start/End. The type is currently set to string because XML spy does not
2206 seem to validate dateTime entries correctly, but should be changed to dateTime later -->
2207 <complexType name="DateEndpointsType">
2208 <sequence>
2209 <element name="EarliestStart" type="xs:dateTime"/>
2210 <element name="LatestEnd" type="xs:dateTime"/>
2211 </sequence>
2212 </complexType>
2213 <!--For giving a type of preference function. Currently, the preference function can be one of two: (a) a piecewise linear
2214 function, (b) an arbitrary function expressed as a string (such as x*x+ y*y) -->
2215 <complexType name="PreferenceFunctionType">
2216 <sequence>
2217 <choice>
2218 <element name="PiecewiseLinearPiece" maxOccurs="unbounded">
2219 <complexType>
2220 <sequence>
2221 <element name="x-coordinate1" type="xs:string"/>
2222 <element name="y-coordinate1" type="xs:NMTOKEN"/>
2223 <element name="x-coordinate2" type="xs:string"/>
2224 <element name="y-coordinate2" type="xs:NMTOKEN"/>
2225 <!--TODO: Make dateTime/NMTOKEN for generality?-->
2226 </sequence>
2227 </complexType>
2228 </element>
2229 <element name="FunctionDefinedByEquation" type="xs:string"/>
2230 <!--<xs:element name="NoPreference"/> -->
2231 </choice>
2232 </sequence>
2233 </complexType>
2234 <!--For giving a value, and associating with it some preference function -->
2235 <complexType name="ValueWithPreferenceMeasureType">
2236 <sequence>
2237 <element name="Endpoints" type="tns:DateEndpointsType"/>
2238 <element name="PreferenceFunction" type="tns:PreferenceFunctionType" minOccurs="0"/>
2239 </sequence>
2240 </complexType>
2241 <!--For specifying a preference for whether an entry must be present or not -->
2242 <complexType name="PresentOrNotType">
2243 <attribute name="value" use="required">
2244 <simpleType>
2245 <restriction base="xs:string">
2246 <enumeration value="MustBePresent"/>
2247 <enumeration value="MustBeAbsent"/>
2248 <enumeration value="PreferredPresent"/>

```

```

2249         <enumeration value="PreferredAbsent"/>
2250         <enumeration value="Agnostic"/>
2251     </restriction>
2252 </simpleType>
2253 </attribute>
2254 </complexType>
2255 <!--For giving the integer endpoints for elements like retries. -->
2256 <complexType name="IntegerEndPointsType">
2257     <sequence>
2258         <element name="SmallestValue" type="xs:integer"/>
2259         <element name="LatestValue" type="xs:integer"/>
2260     </sequence>
2261 </complexType>
2262 <!--This associates a preference order to the integer end points or a preference such as Smaller is Preferred-->
2263 <complexType name="IntegerValuesWithPreferenceMeasureType">
2264     <sequence>
2265         <element name="EndPoints" type="tns:IntegerEndPointsType"/>
2266         <element name="PreferenceFunction" type="tns:PreferenceFunctionType" minOccurs="0"/>
2267     </sequence>
2268     <attribute name="preferenceOrder">
2269         <simpleType name="orderName">
2270             <restriction base="xs:string">
2271                 <enumeration value="SmallerPreferred"/>
2272                 <enumeration value="LargerPreferred"/>
2273             </restriction>
2274         </simpleType>
2275     </attribute>
2276 </complexType>
2277 <!--This type is for integer values whose (a) presence can be potentially negotiated (b) the values themselves can be
2278 negotiated -->
2279 <complexType name="IntegerValuesType">
2280     <sequence>
2281         <element name="PresentOrNot" type="tns:PresentOrNotType" minOccurs="0"/>
2282         <element name="RangeInfo" type="tns:IntegerValuesWithPreferenceMeasureType"/>
2283     </sequence>
2284 </complexType>
2285 <!--For specifying a preference for whether an entry must be present or not -->
2286 <complexType name="BooleanValuesType">
2287     <sequence>
2288         <element name="PresentOrNot" type="tns:PresentOrNotType" minOccurs="0"/>
2289     </sequence>
2290     <attribute name="preference" use="required">
2291         <simpleType>
2292             <restriction base="xs:string">
2293                 <enumeration value="TruePreferred"/>
2294                 <enumeration value="FalsePreferred"/>
2295                 <enumeration value="Agnostic"/>
2296             </restriction>
2297         </simpleType>
2298     </attribute>
2299 </complexType>
2300 <!--For specifying the minimum and maximum allowable durations-->
2301 <complexType name="DurationType">
2302     <sequence>
2303         <element name="PresentOrNot" type="tns:PresentOrNotType" minOccurs="0"/>
2304         <element name="MinimumDuration" type="xs:duration"/>
2305         <element name="MaximumDuration" type="xs:duration"/>
2306     </sequence>
2307 </complexType>
2308 <!--For also stating that the durations have some preference associated with them-->
2309 <complexType name="DurationWithPreferenceType">
2310     <complexContent>
2311         <extension base="tns:DurationType">
2312             <attribute name="preferenceOrder" use="optional">
2313                 <simpleType name="preferenceName">
2314                     <restriction base="xs:string">

```

```
2315         <enumeration value="SmallerPreferred"/>
2316         <enumeration value="LargerPreferred"/>
2317     </restriction>
2318 </simpleType>
2319 </attribute>
2320 </extension>
2321 </complexContent>
2322 </complexType>
2323 </schema>
2324
2325
2326
```

2327 **Appendix B XML Schema for Negotiation Messages** 2328 **(Normative)**

2329 The XML Schema for the negotiation *Messages* is available in text form at:

2330

2331 **THIS SECTION HAS BEEN REPLACED IN FULL.**

```

2332 <?xml version="1.0" encoding="UTF-8"?>
2333 <!-- This is the schema that corresponds to the version 1.0 CPP/A Negotiation spec
2334 -->
2335 <schema targetNamespace="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpa-negot-1_0.xsd"
2336 xmlns:tns="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpa-negot-1_0.xsd" xmlns:cppa="http://www.oasis-
2337 open.org/committees/ebxml-cppa/schema/cpp-cpa-2_0.xsd" xmlns="http://www.w3.org/2001/XMLSchema">
2338 <import namespace="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpp-cpa-2_0.xsd"
2339 schemaLocation="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpp-cpa-2_0.xsd"/>
2340 <attributeGroup name="id.grp">
2341 <attribute name="id" type="cppa:non-empty-string" use="required"/>
2342 <attribute ref="cppa:version" use="required"/>
2343 </attributeGroup>
2344 <element name="NegotiationMessage">
2345 <complexType>
2346 <sequence>
2347 <element ref="tns:NCPA"/>
2348 <element ref="tns:CPATemplateId"/>
2349 <element ref="cppa:SecurityDetails"/>
2350 <element ref="tns:InitiatingParty"/>
2351 <element ref="tns:RespondingParty"/>
2352 <element ref="tns:BPSSBusinessDocumentName"/>
2353 <element name="ExpirationDate" type="dateTime"/>
2354 <element ref="tns:BusinessDocuments"/>
2355 <element ref="tns:NegotiationContent"/>
2356 <element name="ResponseToURL" type="anyURI" minOccurs="0"/>
2357 <element name="Offer-CounterOffer-Acceptance-Time" type="dateTime" minOccurs="0"/>
2358 <element name="Comment" type="string" minOccurs="0"/>
2359 </sequence>
2360 <attribute name="businessMsgId" type="ID" use="required"/>
2361 <attribute name="negotiationDialogId" type="cppa:non-empty-string" use="required"/>
2362 <attribute name="offerId" type="cppa:non-empty-string"/>
2363 <attribute name="inresponseTo" type="cppa:non-empty-string"/>
2364 <attribute name="binding" type="boolean" use="required"/>
2365 <attribute name="messageType" type="tns:messageTypeValue.type" use="required"/>
2366 <attribute name="error" type="tns:errorValue.type"/>
2367 </complexType>
2368 </element>
2369 <element name="CPATemplateId">
2370 <complexType>
2371 <attributeGroup ref="tns:id.grp"/>
2372 </complexType>
2373 </element>
2374 <element name="NCPA">
2375 <complexType>
2376 <attribute name="uri" type="anyURI" use="required"/>
2377 </complexType>
2378 </element>
2379 <element name="BPSSBusinessDocumentName">
2380 <complexType>
2381 <attribute name="name" type="tns:bpssBusinessDocumentName.type" use="required"/>
2382 </complexType>
2383 </element>
2384 <element name="BusinessDocuments">
2385 <complexType>
2386 <choice>

```

```

2387     <element name="CPATemplateDoc" type="tns:CPATemplateDoc.type"/>
2388   </choice>
2389 </complexType>
2390 </element>
2391 <element name="NegotiationContent">
2392   <complexType>
2393     <sequence>
2394       <element ref="tns:AcceptedItem" maxOccurs="unbounded"/>
2395       <element ref="tns:DeletedItem" minOccurs="0" maxOccurs="unbounded"/>
2396       <element ref="tns:UpdatedItem" minOccurs="0" maxOccurs="unbounded"/>
2397       <element ref="tns:InsertedItem" minOccurs="0" maxOccurs="unbounded"/>
2398     </sequence>
2399   </complexType>
2400 </element>
2401 <element name="AcceptedItem" type="tns:simpleItem.type"/>
2402 <element name="DeletedItem" type="tns:simpleItem.type"/>
2403 <element name="UpdatedItem" type="tns:updatedItem.type"/>
2404 <element name="InsertedItem" type="tns:insertedItem.type"/>
2405 <element name="InitiatingParty" type="tns:partySummary.type"/>
2406 <element name="RespondingParty" type="tns:partySummary.type"/>
2407 <complexType name="simpleItem.type">
2408   <attribute name="xpath" type="tns:xpath.type" use="required"/>
2409 </complexType>
2410 <complexType name="updatedItem.type">
2411   <attribute name="xpath" type="tns:xpath.type" use="required"/>
2412   <attribute name="originalValue" type="cppa:non-empty-string" use="required"/>
2413   <attribute name="proposedValue" type="cppa:non-empty-string" use="required"/>
2414   <attribute name="itemstatus" type="tns:itemstatus.type" use="required"/>
2415 </complexType>
2416 <complexType name="insertedItem.type">
2417   <attribute name="xpath" type="tns:xpath.type" use="required"/>
2418   <attribute name="proposedValue" type="cppa:non-empty-string" use="required"/>
2419   <attribute name="itemstatus" type="tns:itemstatus.type" use="required"/>
2420 </complexType>
2421 <complexType name="doc.type">
2422   <choice>
2423     <element name="BinaryDoc" type="base64Binary"/>
2424     <element name="Uri" type="anyURI"/>
2425   </choice>
2426 </complexType>
2427 <complexType name="CPATemplateDoc.type">
2428   <sequence>
2429     <element name="NDD" type="tns:doc.type"/>
2430     <element name="CPATemplate" type="tns:doc.type"/>
2431   </sequence>
2432 </complexType>
2433 <complexType name="partySummary.type">
2434   <sequence>
2435     <element ref="cppa:PartyId"/>
2436     <element name="CPPId">
2437       <complexType>
2438         <attributeGroup ref="tns:id.grp"/>
2439       </complexType>
2440     </element>
2441     <element name="CPPNDD" type="tns:doc.type" minOccurs="0"/>
2442   </sequence>
2443 </complexType>
2444 <simpleType name="xpath.type">
2445   <restriction base="string"/>
2446 </simpleType>
2447 <simpleType name="itemstatus.type">
2448   <restriction base="NMTOKEN">
2449     <enumeration value="Required"/>
2450     <enumeration value="Preferred"/>
2451   </restriction>
2452 </simpleType>

```

```
2453 <simpleType name="messageTypeValue.type">
2454 <restriction base="NMTOKEN">
2455 <enumeration value="Offer"/>
2456 <enumeration value="CounterOffer"/>
2457 <enumeration value="CounterOfferPending"/>
2458 <enumeration value="Rejected"/>
2459 <enumeration value="Accepted"/>
2460 <enumeration value="Expired"/>
2461 <enumeration value="SinglePartySigned"/>
2462 <enumeration value="Signed"/>
2463 <enumeration value="Unsigned"/>
2464 </restriction>
2465 </simpleType>
2466 <simpleType name="errorValue.type">
2467 <restriction base="NMTOKEN">
2468 <enumeration value="ExpiredCPP"/>
2469 <enumeration value="UnableToFulfillSecurityRequirements"/>
2470 <enumeration value="ProposedSecurityPolicyInadequate"/>
2471 <enumeration value="OutOfSequenceCounterOffer"/>
2472 <enumeration value="FailedSignatureValidation_CPATemplate"/>
2473 <enumeration value="FailedSignatureValidation_CPA"/>
2474 <enumeration value="UnsupportedBusinessRelationship"/>
2475 <enumeration value="UnsupportedPackaging"/>
2476 <enumeration value="UnsupportedSignal"/>
2477 <enumeration value="FailedToConverge"/>
2478 <enumeration value="PreviouslyRejectedCPA"/>
2479 <enumeration value="ExpiredOffer"/>
2480 <enumeration value="FormatError"/>
2481 <enumeration value="UnknownSystemError"/>
2482 </restriction>
2483 </simpleType>
2484 <simpleType name="bpsBusinessDocumentName.type">
2485 <restriction base="NMTOKEN">
2486 <enumeration value="CPA_Offer_Doc"/>
2487 <enumeration value="CPA_Accept_Offer_Doc"/>
2488 <enumeration value="CPA_Counter_Pending_Offer_Doc"/>
2489 <enumeration value="CPA_Counter_Offer_Doc"/>
2490 <enumeration value="CPA_Reject_Offer_Doc"/>
2491 <enumeration value="CPA_Final_Doc"/>
2492 <enumeration value="CPA_Final_Response_Doc"/>
2493 <enumeration value="CPA_Final_Response_Doc_Signed"/>
2494 <enumeration value="CPA_Final_Response_Reject_Doc"/>
2495 </restriction>
2496 </simpleType>
2497 </schema>
2498
2499
```

2500 Appendix C Negotiation CPA Example (Non-Normative)

2501 The text file for this *NCPA* example is available at:

2502
 2503 **THE NCPA'S PACKAGING DEFINITIONS HAVE TO BE COMPLETED NOW THAT**
 2504 **THE NDD AND MESSAGE SCHEMA ARE COMPLETED. OF PARTICULAR CONCERN**
 2505 **ARE THE PACKAGING DEFINITION AND CHANGES THAT MAY BE NEEDED TO**
 2506 **MATCH THE CHANGES IN THE NEGOTIATION BPSS WITH REGARD TO THE**
 2507 **CONCLUSION OF NEGOTIATION.**

Deleted: AFTER

```

2509 <?xml version="1.0"?>
2510 <!-- edited with XML Spy v4.4 U (http://www.xmlspy.com) by Hima Mukkamala (Web Services Architecture WG) -->
2511 <tp:CollaborationProtocolAgreement xmlns:tp="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpp-cpa-
2512 2_0.xsd" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xlink="http://www.w3.org/1999/xlink"
2513 xmlns:ds="http://www.w3.org/2000/09/xmldsig#" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
2514 xsi:schemaLocation="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpp-cpa-2_0.xsd
2515 cpp-cpa-2_0.xsd" tp:cpaid="uri:NegotInit-and-NegoResp-cpa" tp:version="2_0a">
2516   <tp:Status tp:value="proposed"/>
2517   <tp:Start>2001-05-20T07:21:00Z</tp:Start>
2518   <tp:End>2003-05-20T07:21:00Z</tp:End>
2519   <tp:ConversationConstraints tp:invocationLimit="100" tp:concurrentConversations="10"/>
2520   <!-- Party info for Negotiation Initiator -->
2521   <tp:PartyInfo tp:partyName="NegotiationInitiator" tp:defaultMshChannelId="asyncChannelA1"
2522 tp:defaultMshPackageId="Negolnit_MshSignalPackage">
2523     <tp:PartyId tp:type="urn:oasis:names:tc:ebxml-cppa:partyid-type:duns">123456789</tp:PartyId>
2524     <tp:PartyRef xlink:href="http://Negolnit.com/about.html"/>
2525     <!-- This role is for Negotiation Initiator performing the role of Negotiation Initiator -->
2526     <tp:CollaborationRole>
2527       <tp:ProcessSpecification tp:version="2.0" tp:name="CPA-Negotiation" xlink:type="simple"
2528 xlink:href="http://www.oasis-open.org/committees/ebxml-cppa-negot/CPA_Negotiation_BPSS.xml"
2529 tp:uuid="bpid:ebXML:CPA-Negotiation"/>
2530       <tp:Role tp:name="CPA Negotiation Initiator" xlink:type="simple" xlink:href="http://www.oasis-
2531 open.org/committees/ebxml-cppa-negot/CPA_Negotiation_BPSS.xml#CPA Negotiation Initiator"/>
2532       <tp:ServiceBinding>
2533         <tp:Service>bpid:ebXML:CPA-Negotiation</tp:Service>
2534         <!-- This send is for sending the Negotiation Offer -->
2535         <tp:CanSend>
2536           <tp:ThisPartyActionBinding tp:id="Negolnit_ABID1" tp:action="CPA_Offer_BT_ReqBA"
2537 tp:packageId="Negolnit_OfferRequestPackage">
2538             <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
2539 tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2540 tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2541             <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2542 tp:businessTransactionActivity="CPA Offer BTA" tp:requestOrResponseAction="CPA_Offer_BT_ReqBA"/>
2543             <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2544           </tp:ThisPartyActionBinding>
2545           <tp:OtherPartyActionBinding>NegoResp_ABID1</tp:OtherPartyActionBinding>
2546         </tp:CanSend>
2547         <!-- This send is for sending the Receipt Acknowledgment -->
2548         <tp:CanSend>
2549           <tp:ThisPartyActionBinding tp:id="Negolnit_ABID2" tp:action="ReceiptAcknowledgement"
2550 tp:packageId="Negolnit_ReceiptAcknowledgmentPackage">
2551             <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="true"
2552 tp:isNonRepudiationReceiptRequired="true" tp:isConfidential="transient" tp:isAuthenticated="persistent"
2553 tp:isTamperProof="persistent" tp:isAuthorizationRequired="true"/>
2554             <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2555           </tp:ThisPartyActionBinding>
2556           <tp:OtherPartyActionBinding>NegoResp_ABID2</tp:OtherPartyActionBinding>
2557         </tp:CanSend>
2558         <!-- This send is for send the Final message in the collaboration. This would be the double signed CPA

```

```

2559 document or acceptance or reject of the CPA in the final Response document-->
2560 <tp:CanSend>
2561 <tp:ThisPartyActionBinding tp:id="Negolnit_FinalResponseMessageA"
2562 tp:action="Final_CPA_BT_RespBA" tp:packageId="Negolnit_FinalMessage">
2563 <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
2564 tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2565 tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2566 <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2567 tp:businessTransactionActivity="CPA Final BTA" tp:requestOrResponseAction="Final_CPA_BT_RespBA"/>
2568 <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2569 </tp:ThisPartyActionBinding>
2570 <tp:OtherPartyActionBinding>NegoResp_FinalResponseMessageB</tp:OtherPartyActionBinding>
2571 </tp:CanSend>
2572 <!-- This receive is for receiving the response for Negotiation Offer, could be acceptance, reject or counter
2573 offer-->
2574 <tp:CanReceive>
2575 <tp:ThisPartyActionBinding tp:id="Negolnit_ABID9" tp:action="CPA_Offer_BT_RespBA"
2576 tp:packageId="Negolnit_OfferResponsePackage">
2577 <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
2578 tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2579 tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2580 <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2581 tp:businessTransactionActivity="CPA Offer BTA" tp:requestOrResponseAction="CPA_Offer_BT_RespBA"/>
2582 <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2583 </tp:ThisPartyActionBinding>
2584 <tp:OtherPartyActionBinding>NegoResp_ABID9</tp:OtherPartyActionBinding>
2585 </tp:CanReceive>
2586 <!-- This receive is for receiving the Final Response document in the final BTA -->
2587 <tp:CanReceive>
2588 <tp:ThisPartyActionBinding tp:id="Negolnit_FinalResponseA" tp:action="Final_CPA_BT_ReqBA"
2589 tp:packageId="Negolnit_FinalMessage">
2590 <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
2591 tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2592 tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2593 <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2594 tp:businessTransactionActivity="CPA Final BTA" tp:requestOrResponseAction="Final_CPA_BT_ReqBA"/>
2595 <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2596 </tp:ThisPartyActionBinding>
2597 <tp:OtherPartyActionBinding>NegoResp_FinalResponseB</tp:OtherPartyActionBinding>
2598 </tp:CanReceive>
2599 <!-- This Receive is for receiving the Receipt Acknowledgment -->
2600 <tp:CanReceive>
2601 <tp:ThisPartyActionBinding tp:id="Negolnit_ABID3" tp:action="ReceiptAcknowledgment"
2602 tp:packageId="Negolnit_ReceiptAcknowledgmentPackage">
2603 <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="true"
2604 tp:isNonRepudiationReceiptRequired="true" tp:isConfidential="transient" tp:isAuthenticated="persistent"
2605 tp:isTamperProof="persistent" tp:isAuthorizationRequired="true"/>
2606 <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2607 </tp:ThisPartyActionBinding>
2608 <tp:OtherPartyActionBinding>NegoResp_ABID3</tp:OtherPartyActionBinding>
2609 </tp:CanReceive>
2610 <!-- This Receive is for receiving the Exception -->
2611 <tp:CanReceive>
2612 <tp:ThisPartyActionBinding tp:id="Negolnit_ABID4" tp:action="Exception"
2613 tp:packageId="Negolnit_ExceptionPackage">
2614 <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="true"
2615 tp:isNonRepudiationReceiptRequired="true" tp:isConfidential="transient" tp:isAuthenticated="persistent"
2616 tp:isTamperProof="persistent" tp:isAuthorizationRequired="true"/>
2617 <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2618 </tp:ThisPartyActionBinding>
2619 <tp:OtherPartyActionBinding>NegoResp_ABID4</tp:OtherPartyActionBinding>
2620 </tp:CanReceive>
2621 </tp:ServiceBinding>
2622 </tp:CollaborationRole>
2623 <!-- This role is for Negotiation Initiator company performing the role of Negotiation Counter offer responder -->
2624 <tp:CollaborationRole>

```

```

2625         <tp:ProcessSpecification tp:version="2.0" tp:name="CPA-Negotiation" xlink:type="simple"
2626         xlink:href="http://www.oasis-open.org/committees/ebxml-cppa-negot/CPA_Negotiation_BPSS.xml"
2627         tp:uid="bpid:ebXML:CPPA-Negotiation"/>
2628         <tp:Role tp:name="CPA Negotiation Counter Offer Responder" xlink:type="simple" xlink:href="http://www.oasis-
2629         open.org/committees/ebxml-cppa-negot/CPA_Negotiation_BPSS.xml#CPA Negotiation Counter Offer Responder"/>
2630         <tp:ServiceBinding>
2631             <tp:Service>bpid:ebXML:CPPA-Negotiation</tp:Service>
2632             <!-- This send is for sending the Negotiation Counter Offer in "CPA Counter Offer 2 BTA"-->
2633             <tp:CanSend>
2634                 <tp:ThisPartyActionBinding tp:id="Negolnit_ABID5" tp:action="CPA_Counter_Offer_BT_ReqBA"
2635                 tp:packageId="Negolnit_CounterOfferRequestPackage">
2636                     <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
2637                     tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2638                     tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2639                     <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2640                     tp:businessTransactionActivity="CPA Counter Offer 2 BTA"
2641                     tp:requestOrResponseAction="CPA_Counter_Offer_BT_ReqBA">
2642                         <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
2643                         </tp:ActionContext>
2644                         <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2645                         </tp:ThisPartyActionBinding>
2646                         <tp:OtherPartyActionBinding>NegoResp_ABID5</tp:OtherPartyActionBinding>
2647                     </tp:CanSend>
2648                     <!-- This send is for sending the Negotiation Counter Offer Response in "CPA Counter Offer 1 BTA"-->
2649                     <tp:CanSend>
2650                         <tp:ThisPartyActionBinding tp:id="Negolnit_ABID6" tp:action="CPA_Counter_Offer_BT_ReqBA"
2651                         tp:packageId="Negolnit_CounterOfferResponsePackage">
2652                             <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
2653                             tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2654                             tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2655                             <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2656                             tp:businessTransactionActivity="CPA Counter Offer 1 BTA"
2657                             tp:requestOrResponseAction="CPA_Counter_Offer_BT_RespBA">
2658                                 <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
2659                                 </tp:ActionContext>
2660                                 <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2661                                 </tp:ThisPartyActionBinding>
2662                                 <tp:OtherPartyActionBinding>NegoResp_ABID6</tp:OtherPartyActionBinding>
2663                             </tp:CanSend>
2664                             <!-- This send is for sending the Final CPA Response message in CPA_Final_BTA_init_Responder"-->
2665                             <tp:CanSend>
2666                                 <tp:ThisPartyActionBinding tp:id="NegoCOR_FinalMessageA" tp:action="Final_CPA_BT_ReqBA"
2667                                 tp:packageId="Negolnit_FinalMessage">
2668                                     <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
2669                                     tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2670                                     tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2671                                     <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2672                                     tp:businessTransactionActivity="CPA_Final_BTA_init_Responder" tp:requestOrResponseAction="Final_CPA_BT_ReqBA">
2673                                         <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
2674                                         </tp:ActionContext>
2675                                         <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2676                                         </tp:ThisPartyActionBinding>
2677                                         <tp:OtherPartyActionBinding>NegoCOResp_FinalMessageB</tp:OtherPartyActionBinding>
2678                                     </tp:CanSend>
2679                                     <!-- This send is for sending the response to the Final CPA Response message in
2680                                     CPA_Final_BTA_init_Initiator"-->
2681                                     <tp:CanSend>
2682                                         <tp:ThisPartyActionBinding tp:id="NegoCOR_FinalMessageResponseA"
2683                                         tp:action="Final_CPA_BT_RespBA" tp:packageId="Negolnit_FinalMessage">
2684                                             <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
2685                                             tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2686                                             tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2687                                             <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2688                                             tp:businessTransactionActivity="CPA_Final_BTA_init_Initiator" tp:requestOrResponseAction="Final_CPA_BT_RespBA">
2689                                                 <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
2690                                                 </tp:ActionContext>

```

```

2691     <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2692     </tp:ThisPartyActionBinding>
2693     <tp:OtherPartyActionBinding>NegoCOResp_FinalMessageResponseB</tp:OtherPartyActionBinding>
2694 </tp:CanSend>
2695 <!-- This send is for sending the Receipt Acknowledgment -->
2696 <tp:CanSend>
2697     <tp:ThisPartyActionBinding tp:id="NegoInit_ABID13" tp:action="ReceiptAcknowledgement"
2698 tp:packageId="NegoInit_ReceiptAcknowledgmentPackage">
2699         <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="true"
2700 tp:isNonRepudiationReceiptRequired="true" tp:isConfidential="transient" tp:isAuthenticated="persistent"
2701 tp:isTamperProof="persistent" tp:isAuthorizationRequired="true"/>
2702         <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2703     </tp:ThisPartyActionBinding>
2704     <tp:OtherPartyActionBinding>NegoResp_ABID13</tp:OtherPartyActionBinding>
2705 </tp:CanSend>
2706 <!-- This receive is for receiving the Final CPA message in CPA_Final_BTA_init_Initiator-->
2707 <tp:CanReceive>
2708     <tp:ThisPartyActionBinding tp:id="NegoCOR_FinalMessageA1" tp:action="Final_CPA_BT_ReqBA"
2709 tp:packageId="NegoInit_FinalMessage">
2710         <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
2711 tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2712 tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2713         <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2714 tp:businessTransactionActivity="CPA_Final_BTA_init_Initiator" tp:requestOrResponseAction="Final_CPA_BT_ReqBA">
2715             <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
2716         </tp:ActionContext>
2717         <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2718     </tp:ThisPartyActionBinding>
2719     <tp:OtherPartyActionBinding>NegoCOResp_FinalMessageB1</tp:OtherPartyActionBinding>
2720 </tp:CanReceive>
2721 <!-- This receive is for receiving the response to the Final CPA message in
2722 CPA_Final_BTA_init_Responder-->
2723 <tp:CanReceive>
2724     <tp:ThisPartyActionBinding tp:id="NegoCOR_FinalMessageResponseA2"
2725 tp:action="Final_CPA_BT_RespBA" tp:packageId="NegoInit_FinalMessage">
2726         <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
2727 tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2728 tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2729         <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2730 tp:businessTransactionActivity="CPA_Final_BTA_init_Responder"
2731 tp:requestOrResponseAction="Final_CPA_BT_RespBA">
2732             <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
2733         </tp:ActionContext>
2734         <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2735     </tp:ThisPartyActionBinding>
2736     <tp:OtherPartyActionBinding>NegoCOResp_FinalMessageResponseB2</tp:OtherPartyActionBinding>
2737 </tp:CanReceive>
2738 <!-- This receive is for receiving the response forNegotiation Counter Offer, could be accept, reject or again
2739 send a counter offer This happens in "CPA Counter Offer 2 BTA"-->
2740 <tp:CanReceive>
2741     <tp:ThisPartyActionBinding tp:id="NegoInit_ABID10" tp:action="CPA_Counter_Offer_BT_RespBA"
2742 tp:packageId="NegoInit_CounterOfferResponsePackage">
2743         <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
2744 tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2745 tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2746         <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2747 tp:businessTransactionActivity="CPA Counter Offer 2 BTA"
2748 tp:requestOrResponseAction="CPA_Counter_Offer_BT_RespBA">
2749             <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
2750         </tp:ActionContext>
2751         <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2752     </tp:ThisPartyActionBinding>
2753     <tp:OtherPartyActionBinding>NegoResp_ABID10</tp:OtherPartyActionBinding>
2754 </tp:CanReceive>
2755 <!-- This receive is for receiving the Negotiation Counter Offer. This happens in "CPA Counter Offer 1 BTA"--
2756 >

```

```

2757         <tp:CanReceive>
2758             <tp:ThisPartyActionBinding tp:id="Negolnit_ABID12" tp:action="CPA_Counter_Offer_BT_RespBA"
2759 tp:packageId="Negolnit_CounterOfferRequestPackage">
2760                 <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
2761 tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2762 tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2763                 <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2764 tp:businessTransactionActivity="CPA Counter Offer 1 BTA"
2765 tp:requestOrResponseAction="CPA_Counter_Offer_BT_ReqBA">
2766                     <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
2767                     </tp:ActionContext>
2768                     <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2769                 </tp:ThisPartyActionBinding>
2770                 <tp:OtherPartyActionBinding>NegoResp_ABID12</tp:OtherPartyActionBinding>
2771             </tp:CanReceive>
2772             <!-- This Receive is for receiving the Receipt Acknowledgment -->
2773             <tp:CanReceive>
2774                 <tp:ThisPartyActionBinding tp:id="Negolnit_ABID7" tp:action="ReceiptAcknowledgment"
2775 tp:packageId="Negolnit_ReceiptAcknowledgmentPackage">
2776                     <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="true"
2777 tp:isNonRepudiationReceiptRequired="true" tp:isConfidential="transient" tp:isAuthenticated="persistent"
2778 tp:isTamperProof="persistent" tp:isAuthorizationRequired="true"/>
2779                     <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2780                 </tp:ThisPartyActionBinding>
2781                 <tp:OtherPartyActionBinding>NegoResp_ABID7</tp:OtherPartyActionBinding>
2782             </tp:CanReceive>
2783             <!-- This Receive is for receiving the Exception -->
2784             <tp:CanReceive>
2785                 <tp:ThisPartyActionBinding tp:id="Negolnit_ABID8" tp:action="Exception"
2786 tp:packageId="Negolnit_ExceptionPackage">
2787                     <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="true"
2788 tp:isNonRepudiationReceiptRequired="true" tp:isConfidential="transient" tp:isAuthenticated="persistent"
2789 tp:isTamperProof="persistent" tp:isAuthorizationRequired="true"/>
2790                     <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2791                 </tp:ThisPartyActionBinding>
2792                 <tp:OtherPartyActionBinding>NegoResp_ABID8</tp:OtherPartyActionBinding>
2793             </tp:CanReceive>
2794             </tp:ServiceBinding>
2795         </tp:CollaborationRole>
2796         <!-- Certificates used by the "Negotiation Initiator" company -->
2797         <tp:Certificate tp:certId="Negolnit_AppCert">
2798             <ds:KeyInfo>
2799                 <ds:KeyName>Negolnit_AppCert_Key</ds:KeyName>
2800             </ds:KeyInfo>
2801         </tp:Certificate>
2802         <tp:SecurityDetails tp:securityId="Negolnit_MessageSecurity">
2803             <tp:TrustAnchors>
2804                 <tp:AnchorCertificateRef tp:certId="Negolnit_AppCert"/>
2805             </tp:TrustAnchors>
2806         </tp:SecurityDetails>
2807         <tp:DeliveryChannel tp:channelId="asyncChannelA1" tp:transportId="transportA1"
2808 tp:docExchangeId="docExchangeA1">
2809             <tp:MessagingCharacteristics tp:syncReplyMode="none" tp:ackRequested="always"
2810 tp:ackSignatureRequested="always" tp:duplicateElimination="always"/>
2811         </tp:DeliveryChannel>
2812         <tp:Transport tp:transportId="transportA1">
2813             <tp:TransportSender>
2814                 <tp:TransportProtocol tp:version="1.1">HTTP</tp:TransportProtocol>
2815                 <tp:AccessAuthentication>basic</tp:AccessAuthentication>
2816             </tp:TransportSender>
2817             <tp:TransportReceiver>
2818                 <tp:TransportProtocol tp:version="1.1">HTTP</tp:TransportProtocol>
2819                 <tp:AccessAuthentication>basic</tp:AccessAuthentication>
2820                 <tp:Endpoint tp:uri="https://www.Negolnit.com/servlets/ebxmlhandler/async" tp:type="allPurpose"/>
2821             </tp:TransportReceiver>
2822         </tp:Transport>

```

```

2823     <tp:DocExchange tp:docExchangeId="docExchangeA1">
2824       <tp:ebXMLSenderBinding tp:version="2.0"/>
2825       <tp:ebXMLReceiverBinding tp:version="2.0"/>
2826     </tp:DocExchange>
2827   </tp:PartyInfo>
2828   <!-- Party info for Negotiation Responder -->
2829   <tp:PartyInfo tp:partyName="NegotiationResponder" tp:defaultMshChannelId="asyncChannelB1"
2830 tp:defaultMshPackageId="Negolnit_MshSignalPackage">
2831     <tp:PartyId tp:type="urn:oasis:names:tc:ebxml-cppa:partyid-type:duns">123456789</tp:PartyId>
2832     <tp:PartyRef xlink:href="http://NegoResp.com/about.html"/>
2833     <!-- This role is for Negotiation Responder performing the role of Negotiation Responder -->
2834     <tp:CollaborationRole>
2835       <tp:ProcessSpecification tp:version="2.0" tp:name="CPA-Negotiation" xlink:type="simple"
2836 xlink:href="http://www.oasis-open.org/committees/ebxml-cppa-negot/CPA_Negotiation_BPSS.xml"
2837 tp:uuid="bpid:ebXML:CPA-Negotiation"/>
2838       <tp:Role tp:name="CPA Negotiation Responder" xlink:type="simple" xlink:href="http://www.oasis-
2839 open.org/committees/ebxml-cppa-negot/CPA_Negotiation_BPSS.xml#CPA Negotiation Responder"/>
2840       <tp:ServiceBinding>
2841         <tp:Service>bpid:ebXML:CPA-Negotiation</tp:Service>
2842         <!-- This send is for sending the Negotiation Offer Response, this could be accept, pending, response-->
2843         <tp:CanSend>
2844           <tp:ThisPartyActionBinding tp:id="NegoResp_ABID9" tp:action="CPA_Offer_BT_RespBA"
2845 tp:packageId="Negolnit_OfferResponsePackage">
2846             <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
2847 tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2848 tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2849             <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2850 tp:businessTransactionActivity="CPA Offer BTA" tp:requestOrResponseAction="CPA_Offer_BT_RespBA"/>
2851             <tp:ChannelId>asyncChannelB1</tp:ChannelId>
2852           </tp:ThisPartyActionBinding>
2853           <tp:OtherPartyActionBinding>Negolnit_ABID9</tp:OtherPartyActionBinding>
2854         </tp:CanSend>
2855         <!-- This send is for sending the Final Response document in the final BTA -->
2856         <tp:CanSend>
2857           <tp:ThisPartyActionBinding tp:id="NegoResp_FinalResponseB" tp:action="Final_CPA_BT_ReqBA"
2858 tp:packageId="Negolnit_FinalMessage">
2859             <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
2860 tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2861 tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2862             <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2863 tp:businessTransactionActivity="CPA Final BTA" tp:requestOrResponseAction="Final_CPA_BT_ReqBA"/>
2864             <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2865           </tp:ThisPartyActionBinding>
2866           <tp:OtherPartyActionBinding>Negolnit_FinalResponseA</tp:OtherPartyActionBinding>
2867         </tp:CanSend>
2868         <!-- This send is for sending the Receipt Acknowledgment -->
2869         <tp:CanSend>
2870           <tp:ThisPartyActionBinding tp:id="NegoResp_ABID3" tp:action="ReceiptAcknowledgement"
2871 tp:packageId="Negolnit_ReceiptAcknowledgmentPackage">
2872             <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="true"
2873 tp:isNonRepudiationReceiptRequired="true" tp:isConfidential="transient" tp:isAuthenticated="persistent"
2874 tp:isTamperProof="persistent" tp:isAuthorizationRequired="true"/>
2875             <tp:ChannelId>asyncChannelB1</tp:ChannelId>
2876           </tp:ThisPartyActionBinding>
2877           <tp:OtherPartyActionBinding>Negolnit_ABID3</tp:OtherPartyActionBinding>
2878         </tp:CanSend>
2879         <!-- This receive is for receiving the Final message in the collaboration. This would be the double signed
2880 CPA document or acceptance or reject of the CPA in the final Response document-->
2881         <tp:CanReceive>
2882           <tp:ThisPartyActionBinding tp:id="NegoResp_FinalResponseMessageB"
2883 tp:action="Final_CPA_BT_RespBA" tp:packageId="Negolnit_FinalMessage">
2884             <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
2885 tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2886 tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2887             <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2888 tp:businessTransactionActivity="CPA Final BTA" tp:requestOrResponseAction="Final_CPA_BT_RespBA"/>

```

```

2889     <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2890     </tp:ThisPartyActionBinding>
2891     <tp:OtherPartyActionBinding>Negolnit_FinalResponseMessageA</tp:OtherPartyActionBinding>
2892   </tp:CanReceive>
2893   <!-- This receive is for receiving the offer in the first place -->
2894   <tp:CanReceive>
2895     <tp:ThisPartyActionBinding tp:id="NegoResp_ABID1" tp:action="CPA_Offer_BT_ReqBA"
2896     tp:packageId="Negolnit_OfferRequestPackage">
2897       <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
2898       tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2899       tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2900       <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2901       tp:businessTransactionActivity="CPA Offer BTA" tp:requestOrResponseAction="CPA_Offer_BT_ReqBA"/>
2902       <tp:ChannelId>asyncChannelB1</tp:ChannelId>
2903     </tp:ThisPartyActionBinding>
2904     <tp:OtherPartyActionBinding>Negolnit_ABID1</tp:OtherPartyActionBinding>
2905   </tp:CanReceive>
2906   <!-- This Receive is for receiving the Receipt Acknowledgment -->
2907   <tp:CanReceive>
2908     <tp:ThisPartyActionBinding tp:id="NegoResp_ABID2" tp:action="ReceiptAcknowledgment"
2909     tp:packageId="Negolnit_ReceiptAcknowledgmentPackage">
2910       <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="true"
2911       tp:isNonRepudiationReceiptRequired="true" tp:isConfidential="transient" tp:isAuthenticated="persistent"
2912       tp:isTamperProof="persistent" tp:isAuthorizationRequired="true"/>
2913       <tp:ChannelId>asyncChannelB1</tp:ChannelId>
2914     </tp:ThisPartyActionBinding>
2915     <tp:OtherPartyActionBinding>Negolnit_ABID2</tp:OtherPartyActionBinding>
2916   </tp:CanReceive>
2917   <!-- This Receive is for receiving the Exception -->
2918   <tp:CanReceive>
2919     <tp:ThisPartyActionBinding tp:id="NegoResp_ABID4" tp:action="Exception"
2920     tp:packageId="Negolnit_ExceptionPackage">
2921       <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="true"
2922       tp:isNonRepudiationReceiptRequired="true" tp:isConfidential="transient" tp:isAuthenticated="persistent"
2923       tp:isTamperProof="persistent" tp:isAuthorizationRequired="true"/>
2924       <tp:ChannelId>asyncChannelB1</tp:ChannelId>
2925     </tp:ThisPartyActionBinding>
2926     <tp:OtherPartyActionBinding>Negolnit_ABID4</tp:OtherPartyActionBinding>
2927   </tp:CanReceive>
2928   </tp:ServiceBinding>
2929 </tp:CollaborationRole>
2930 <!-- This role is for Negotiation Responder company performing the role of Negotiation Counter offer initiator -->
2931 <tp:CollaborationRole>
2932   <tp:ProcessSpecification tp:version="2.0" tp:name="CPPA-Negotiation" xlink:type="simple"
2933   xlink:href="http://www.oasis-open.org/committees/ebxml-cppa-negot/CPA_Negotiation_BPSS.xml"
2934   tp:uuid="bpid:ebXML:CPPA-Negotiation"/>
2935   <tp:Role tp:name="CPA Negotiation Counter Offer Initiator" xlink:type="simple" xlink:href="http://www.oasis-
2936   open.org/committees/ebxml-cppa-negot/CPA_Negotiation_BPSS.xml#CPA Negotiation Counter Offer Initiator"/>
2937   <tp:ServiceBinding>
2938     <tp:Service>bpid:ebXML:CPPA-Negotiation</tp:Service>
2939     <!-- This send is for sending the Negotiation Counter Offer.This happens in "CPA Counter Offer 1 BTA" -->
2940     <tp:CanSend>
2941       <tp:ThisPartyActionBinding tp:id="NegoResp_ABID12" tp:action="CPA_Counter_Offer_BT_ReqBA"
2942       tp:packageId="Negolnit_CounterOfferRequestPackage">
2943         <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
2944         tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2945         tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2946         <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2947         tp:businessTransactionActivity="CPA Counter Offer 1 BTA"
2948         tp:requestOrResponseAction="CPA_Counter_Offer_BT_ReqBA">
2949           <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
2950         </tp:ActionContext>
2951         <tp:ChannelId>asyncChannelB1</tp:ChannelId>
2952       </tp:ThisPartyActionBinding>
2953       <tp:OtherPartyActionBinding>Negolnit_ABID12</tp:OtherPartyActionBinding>
2954     </tp:CanSend>

```

```

2955         <!-- This send is for sending the Negotiation Counter Offer response.This happens in "CPA Counter Offer 2
2956 BTA" -->
2957         <tp:CanSend>
2958             <tp:ThisPartyActionBinding tp:id="NegoResp_ABID10" tp:action="CPA_Counter_Offer_BT_ReqBA"
2959 tp:packageId="Negolnit_CounterOfferResponsePackage">
2960                 <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
2961 tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2962 tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2963                 <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2964 tp:businessTransactionActivity="CPA Counter Offer 2 BTA"
2965 tp:requestOrResponseAction="CPA_Counter_Offer_BT_RespBA">
2966                     <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
2967                 </tp:ActionContext>
2968                 <tp:ChannelId>asyncChannelB1</tp:ChannelId>
2969             </tp:ThisPartyActionBinding>
2970             <tp:OtherPartyActionBinding>Negolnit_ABID10</tp:OtherPartyActionBinding>
2971         </tp:CanSend>
2972         <!-- This send is for sending the Receipt Acknowledgment -->
2973         <tp:CanSend>
2974             <tp:ThisPartyActionBinding tp:id="NegoResp_ABID7" tp:action="ReceiptAcknowledgement"
2975 tp:packageId="Negolnit_ReceiptAcknowledgmentPackage">
2976                 <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="true"
2977 tp:isNonRepudiationReceiptRequired="true" tp:isConfidential="transient" tp:isAuthenticated="persistent"
2978 tp:isTamperProof="persistent" tp:isAuthorizationRequired="true"/>
2979                 <tp:ChannelId>asyncChannelB1</tp:ChannelId>
2980             </tp:ThisPartyActionBinding>
2981             <tp:OtherPartyActionBinding>Negolnit_ABID7</tp:OtherPartyActionBinding>
2982         </tp:CanSend>
2983         <!-- This send is for sending the Final CPA message in CPA_Final_BTA_init_Initiator"-->
2984         <tp:CanSend>
2985             <tp:ThisPartyActionBinding tp:id="NegoCOResp_FinalMessageB1" tp:action="Final_CPA_BT_ReqBA"
2986 tp:packageId="Negolnit_FinalMessage">
2987                 <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
2988 tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2989 tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2990                 <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2991 tp:businessTransactionActivity="CPA_Final_BTA_init_Initiator" tp:requestOrResponseAction="Final_CPA_BT_ReqBA">
2992                     <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
2993                 </tp:ActionContext>
2994                 <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2995             </tp:ThisPartyActionBinding>
2996             <tp:OtherPartyActionBinding>NegoCOR_FinalMessageA1</tp:OtherPartyActionBinding>
2997         </tp:CanSend>
2998         <!-- This send is for sending the response to the Final CPA message in CPA_Final_BTA_init_Responder"-->
2999         <tp:CanSend>
3000             <tp:ThisPartyActionBinding tp:id="NegoCOResp_FinalMessageResponseB2"
3001 tp:action="Final_CPA_BT_RespBA" tp:packageId="Negolnit_FinalMessage">
3002                 <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
3003 tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
3004 tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
3005                 <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
3006 tp:businessTransactionActivity="CPA_Final_BTA_init_Responder"
3007 tp:requestOrResponseAction="Final_CPA_BT_RespBA">
3008                     <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
3009                 </tp:ActionContext>
3010                 <tp:ChannelId>asyncChannelA1</tp:ChannelId>
3011             </tp:ThisPartyActionBinding>
3012             <tp:OtherPartyActionBinding>NegoCOR_FinalMessageResponseA2</tp:OtherPartyActionBinding>
3013         </tp:CanSend>
3014         <!-- This receive is for receiving the response forNegotiation Counter Offer, could be accept, reject or again
3015 send a counter offer This happens in "CPA Counter Offer 1 BTA"-->
3016         <tp:CanReceive>
3017             <tp:ThisPartyActionBinding tp:id="NegoResp_ABID6" tp:action="CPA_Counter_Offer_BT_RespBA"
3018 tp:packageId="Negolnit_CounterOfferResponsePackage">
3019                 <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
3020 tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"

```

```

3021 tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
3022 <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
3023 tp:businessTransactionActivity="CPA Counter Offer 1 BTA"
3024 tp:requestOrResponseAction="CPA_Counter_Offer_BT_RespBA">
3025 <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
3026 </tp:ActionContext>
3027 <tp:ChannelId>asyncChannelB1</tp:ChannelId>
3028 </tp:ThisPartyActionBinding>
3029 <tp:OtherPartyActionBinding>Negolnit_ABID6</tp:OtherPartyActionBinding>
3030 </tp:CanReceive>
3031 <!-- This receive is for receiving Negotiation Counter Offer.This happens in "CPA Counter Offer 2 BTA"-->
3032 <tp:CanReceive>
3033 <tp:ThisPartyActionBinding tp:id="NegoResp_ABID5" tp:action="CPA_Counter_Offer_BT_ReqBA"
3034 tp:packageId="Negolnit_CounterOfferRequestPackage">
3035 <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
3036 tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
3037 tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
3038 <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
3039 tp:businessTransactionActivity="CPA Counter Offer 2 BTA"
3040 tp:requestOrResponseAction="CPA_Counter_Offer_BT_ReqBA">
3041 <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
3042 </tp:ActionContext>
3043 <tp:ChannelId>asyncChannelB1</tp:ChannelId>
3044 </tp:ThisPartyActionBinding>
3045 <tp:OtherPartyActionBinding>Negolnit_ABID5</tp:OtherPartyActionBinding>
3046 </tp:CanReceive>
3047 <!-- This receive is for receiving the Final CPA message in CPA_Final_BTA_init_Responder"-->
3048 <tp:CanReceive>
3049 <tp:ThisPartyActionBinding tp:id="NegoCOResp_FinalMessageB" tp:action="Final_CPA_BT_ReqBA"
3050 tp:packageId="Negolnit_FinalMessage">
3051 <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
3052 tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
3053 tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
3054 <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
3055 tp:businessTransactionActivity="CPA_Final_BTA_init_Responder" tp:requestOrResponseAction="Final_CPA_BT_ReqBA">
3056 <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
3057 </tp:ActionContext>
3058 <tp:ChannelId>asyncChannelA1</tp:ChannelId>
3059 </tp:ThisPartyActionBinding>
3060 <tp:OtherPartyActionBinding>NegoCOR_FinalMessageA</tp:OtherPartyActionBinding>
3061 </tp:CanReceive>
3062 <!-- This receive is for receiving the response to the Final CPA message in CPA_Final_BTA_init_Initiator"-->
3063 <tp:CanReceive>
3064 <tp:ThisPartyActionBinding tp:id="NegoCOResp_FinalMessageResponseB"
3065 tp:action="Final_CPA_BT_RespBA" tp:packageId="Negolnit_FinalMessage">
3066 <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
3067 tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
3068 tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
3069 <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
3070 tp:businessTransactionActivity="CPA_Final_BTA_init_Initiator" tp:requestOrResponseAction="Final_CPA_BT_RespBA">
3071 <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
3072 </tp:ActionContext>
3073 <tp:ChannelId>asyncChannelA1</tp:ChannelId>
3074 </tp:ThisPartyActionBinding>
3075 <tp:OtherPartyActionBinding>NegoCOR_FinalMessageResponseA</tp:OtherPartyActionBinding>
3076 </tp:CanReceive>
3077 <!-- This Receive is for receiving the Receipt Acknowledgment -->
3078 <tp:CanReceive>
3079 <tp:ThisPartyActionBinding tp:id="NegoResp_ABID13" tp:action="ReceiptAcknowledgment"
3080 tp:packageId="Negolnit_ReceiptAcknowledgmentPackage">
3081 <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="true"
3082 tp:isNonRepudiationReceiptRequired="true" tp:isConfidential="transient" tp:isAuthenticated="persistent"
3083 tp:isTamperProof="persistent" tp:isAuthorizationRequired="true"/>
3084 <tp:ChannelId>asyncChannelB1</tp:ChannelId>
3085 </tp:ThisPartyActionBinding>
3086 <tp:OtherPartyActionBinding>Negolnit_ABID13</tp:OtherPartyActionBinding>

```

```

3087         </tp:CanReceive>
3088         <!-- This Receive is for receiving the Exception -->
3089         <tp:CanReceive>
3090             <tp:ThisPartyActionBinding tp:id="NegoResp_ABID8" tp:action="Exception"
3091 tp:packageId="NegoInit_ExceptionPackage">
3092                 <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="true"
3093 tp:isNonRepudiationReceiptRequired="true" tp:isConfidential="transient" tp:isAuthenticated="persistent"
3094 tp:isTamperProof="persistent" tp:isAuthorizationRequired="true"/>
3095                 <tp:ChannelId>asyncChannelB1</tp:ChannelId>
3096             </tp:ThisPartyActionBinding>
3097             <tp:OtherPartyActionBinding>NegoInit_ABID8</tp:OtherPartyActionBinding>
3098         </tp:CanReceive>
3099     </tp:ServiceBinding>
3100 </tp:CollaborationRole>
3101 <!-- Certificates used by the "Negotiation Initiator" company -->
3102 <tp:Certificate tp:certId="NegoResp_AppCert">
3103     <ds:KeyInfo>
3104         <ds:KeyName>NegoResp_AppCert_Key</ds:KeyName>
3105     </ds:KeyInfo>
3106 </tp:Certificate>
3107 <tp:SecurityDetails tp:securityId="NegoResp_MessageSecurity">
3108     <tp:TrustAnchors>
3109         <tp:AnchorCertificateRef tp:certId="NegoResp_AppCert"/>
3110     </tp:TrustAnchors>
3111 </tp:SecurityDetails>
3112 <tp:DeliveryChannel tp:channelId="asyncChannelB1" tp:transportId="transportB1"
3113 tp:docExchangeId="docExchangeB1">
3114     <tp:MessagingCharacteristics tp:syncReplyMode="none" tp:ackRequested="always"
3115 tp:ackSignatureRequested="always" tp:duplicateElimination="always"/>
3116 </tp:DeliveryChannel>
3117 <tp:Transport tp:transportId="transportB1">
3118     <tp:TransportSender>
3119         <tp:TransportProtocol tp:version="1.1">HTTP</tp:TransportProtocol>
3120         <tp:AccessAuthentication>basic</tp:AccessAuthentication>
3121     </tp:TransportSender>
3122     <tp:TransportReceiver>
3123         <tp:TransportProtocol tp:version="1.1">HTTP</tp:TransportProtocol>
3124         <tp:AccessAuthentication>basic</tp:AccessAuthentication>
3125         <tp:Endpoint tp:uri="https://www.NegoResp.com/servlets/ebxmlhandler/async" tp:type="allPurpose"/>
3126     </tp:TransportReceiver>
3127 </tp:Transport>
3128 <tp:DocExchange tp:docExchangeId="docExchangeB1">
3129     <tp:ebXMLSenderBinding tp:version="2.0"/>
3130     <tp:ebXMLReceiverBinding tp:version="2.0"/>
3131 </tp:DocExchange>
3132 </tp:PartyInfo>
3133 <!-- SimplePart corresponding to the SOAP Envelope -->
3134 <tp:SimplePart tp:id="NegoInit_MsgHdr" tp:mimetype="text/xml">
3135     <tp:NamespaceSupported tp:location="http://www.oasis-open.org/committees/ebxml-msg/schema/msg-header-
3136 2_0.xsd" tp:version="2.0">
3137         http://www.oasis-open.org/committees/ebxml-msg/schema/msg-header-2_0.xsd
3138     </tp:NamespaceSupported>
3139 </tp:SimplePart>
3140 <tp:SimplePart tp:id="NegoResp_MsgHdr" tp:mimetype="text/xml">
3141     <tp:NamespaceSupported tp:location="http://www.oasis-open.org/committees/ebxml-msg/schema/msg-header-
3142 2_0.xsd" tp:version="2.0">
3143         http://www.oasis-open.org/committees/ebxml-msg/schema/msg-header-2_0.xsd
3144     </tp:NamespaceSupported>
3145 </tp:SimplePart>
3146 <!-- SimplePart corresponding to a Receipt Acknowledgment business signal -->
3147 <tp:SimplePart tp:id="NegoInit_ReceiptAcknowledgment" tp:mimetype="application/xml">
3148     <tp:NamespaceSupported tp:location="http://www.ebxml.org/bpss/ReceiptAcknowledgment.xsd"
3149 tp:version="2.0">http://www.ebxml.org/bpss/ReceiptAcknowledgment.xsd
3150 </tp:NamespaceSupported>
3151 </tp:SimplePart>
3152 <tp:SimplePart tp:id="NegoResp_ReceiptAcknowledgment" tp:mimetype="application/xml">

```

```

3153     <tp:NamespaceSupported tp:location="http://www.ebxml.org/bpss/ReceiptAcknowledgment.xsd" tp:version="2.0">
3154     http://www.ebxml.org/bpss/ReceiptAcknowledgment.xsd
3155 </tp:NamespaceSupported>
3156 </tp:SimplePart>
3157 <!-- SimplePart corresponding to an Exception business signal -->
3158 <tp:SimplePart tp:id="Negolnit_Exception" tp:mimetype="application/xml">
3159     <tp:NamespaceSupported tp:location="http://www.oasis-open.org/committees/ebxml-msg/schema/msg-header-
3160 2_0.xsd" tp:version="2.0">
3161     http://www.ebxml.org/bpss/Exception.xsd
3162 </tp:NamespaceSupported>
3163 </tp:SimplePart>
3164 <tp:SimplePart tp:id="NegoResp_Exception" tp:mimetype="application/xml">
3165     <tp:NamespaceSupported tp:location="http://www.oasis-open.org/committees/ebxml-msg/schema/msg-header-
3166 2_0.xsd" tp:version="2.0">
3167     http://www.ebxml.org/bpss/Exception.xsd
3168 </tp:NamespaceSupported>
3169 </tp:SimplePart>
3170 <!-- SimplePart corresponding to a negotiation offer request action -->
3171 <tp:SimplePart tp:id="Negolnit_OfferRequest" tp:mimetype="application/xml">
3172     <tp:NamespaceSupported tp:location="http://www.ebxml.org/schemas/NegotiationOffer.xsd" tp:version="1.0">
3173     http://www.ebxml.org/schemas/NegotiationOffer.xsd
3174 </tp:NamespaceSupported>
3175 </tp:SimplePart>
3176 <!-- SimplePart corresponding to a Negotiation offer response action (accept) -->
3177 <tp:SimplePart tp:id="Negolnit_OfferAccept" tp:mimetype="application/xml">
3178     <tp:NamespaceSupported tp:location="http://www.ebxml.org/schemas/OfferAccept.xsd" tp:version="1.0">
3179     http://www.ebxml.org/schemas/OfferAccept.xsd
3180 </tp:NamespaceSupported>
3181 </tp:SimplePart>
3182 <!-- SimplePart corresponding to a Negotiation offer response action (reject) -->
3183 <tp:SimplePart tp:id="Negolnit_OfferReject" tp:mimetype="application/xml">
3184     <tp:NamespaceSupported tp:location="http://www.ebxml.org/schemas/OfferReject.xsd" tp:version="1.0">
3185     http://www.ebxml.org/schemas/OfferReject.xsd
3186 </tp:NamespaceSupported>
3187 </tp:SimplePart>
3188 <!-- SimplePart corresponding to a Negotiation offer response action (counter pending) -->
3189 <tp:SimplePart tp:id="Negolnit_OfferCounterPending" tp:mimetype="application/xml">
3190     <tp:NamespaceSupported tp:location="http://www.ebxml.org/schemas/OfferCounterPending.xsd" tp:version="1.0">
3191     http://www.ebxml.org/schemas/OfferCounterPending.xsd
3192 </tp:NamespaceSupported>
3193 </tp:SimplePart>
3194 <!-- SimplePart corresponding to a Negotiation Counter offer request action -->
3195 <tp:SimplePart tp:id="Negolnit_CounterOfferRequest" tp:mimetype="application/xml">
3196     <tp:NamespaceSupported tp:location="http://www.ebxml.org/schemas/CounterOffer.xsd" tp:version="1.0">
3197     http://www.ebxml.org/schemas/CounterOfferRequest.xsd
3198 </tp:NamespaceSupported>
3199 </tp:SimplePart>
3200 <!-- SimplePart corresponding to a Negotiation Final document being sent in the negotiation process -->
3201 <tp:SimplePart tp:id="Negolnit_FinalMessage" tp:mimetype="application/xml">
3202     <tp:NamespaceSupported tp:location="http://www.ebxml.org/schemas/FinalMessage.xsd" tp:version="1.0">
3203     http://www.ebxml.org/schemas/FinalMessage.xsd
3204 </tp:NamespaceSupported>
3205 </tp:SimplePart>
3206 <!-- SimplePart corresponding to a Negotiation Counter offer request action -->
3207 <tp:SimplePart tp:id="Negolnit_FinalMessageResponse" tp:mimetype="application/xml">
3208     <tp:NamespaceSupported tp:location="http://www.ebxml.org/schemas/FinalMessageResponse.xsd"
3209 tp:version="1.0">
3210     http://www.ebxml.org/schemas/FinalMessageResponse.xsd
3211 </tp:NamespaceSupported>
3212 </tp:SimplePart>
3213 <!-- An ebXML message with a SOAP Envelope only -->
3214 <tp:Packaging tp:id="Negolnit_MshSignalPackage">
3215     <tp:ProcessingCapabilities tp:parse="true" tp:generate="true"/>
3216     <tp:CompositeList>
3217         <tp:Composite tp:id="Negolnit_MshSignal" tp:mimetype="multipart/related" tp:mimeparameters="type=text/xml">
3218             <tp:Constituent tp:idref="Negolnit_MsgHdr"/>

```

```

3219         </tp:Composite>
3220     </tp:CompositeList>
3221 </tp:Packaging>
3222 <!-- An ebXML message with a SOAP Envelope plus a Offer action payload -->
3223 <tp:Packaging tp:id="Negolnit_OfferRequestPackage">
3224     <tp:ProcessingCapabilities tp:parse="true" tp:generate="true"/>
3225     <tp:CompositeList>
3226         <tp:Composite tp:id="Negolnit_OfferRequestMsgId" tp:mimetype="multipart/related"
3227 tp:mimeparameters="type=text/xml">
3228             <tp:Constituent tp:idref="Negolnit_MsgHdr"/>
3229             <tp:Constituent tp:idref="Negolnit_OfferRequest"/>
3230         </tp:Composite>
3231     </tp:CompositeList>
3232 </tp:Packaging>
3233 <!-- An ebXML message with a SOAP Envelope plus a offer response action payload -->
3234 <tp:Packaging tp:id="Negolnit_OfferResponsePackage">
3235     <tp:ProcessingCapabilities tp:parse="true" tp:generate="true"/>
3236     <tp:CompositeList>
3237         <tp:Composite tp:id="Negolnit_OfferResponseAcceptMsgId" tp:mimetype="multipart/related"
3238 tp:mimeparameters="type=text/xml">
3239             <tp:Constituent tp:idref="Negolnit_MsgHdr"/>
3240             <tp:Constituent tp:idref="Negolnit_OfferAccept"/>
3241         </tp:Composite>
3242     </tp:CompositeList>
3243     <tp:CompositeList>
3244         <tp:Composite tp:id="Negolnit_OfferResponseRejectMsgId" tp:mimetype="multipart/related"
3245 tp:mimeparameters="type=text/xml">
3246             <tp:Constituent tp:idref="Negolnit_MsgHdr"/>
3247             <tp:Constituent tp:idref="Negolnit_OfferReject"/>
3248         </tp:Composite>
3249     </tp:CompositeList>
3250     <tp:CompositeList>
3251         <tp:Composite tp:id="Negolnit_OfferResponsePendingMsgId" tp:mimetype="multipart/related"
3252 tp:mimeparameters="type=text/xml">
3253             <tp:Constituent tp:idref="Negolnit_MsgHdr"/>
3254             <tp:Constituent tp:idref="Negolnit_OfferCounterPending"/>
3255         </tp:Composite>
3256     </tp:CompositeList>
3257 </tp:Packaging>
3258 <!-- An ebXML message with a SOAP Envelope plus a counter offer request action payload -->
3259 <tp:Packaging tp:id="Negolnit_CounterOfferRequestPackage">
3260     <tp:ProcessingCapabilities tp:parse="true" tp:generate="true"/>
3261     <tp:CompositeList>
3262         <tp:Composite tp:id="Negolnit_CounterOfferRequestMsgId" tp:mimetype="multipart/related"
3263 tp:mimeparameters="type=text/xml">
3264             <tp:Constituent tp:idref="Negolnit_MsgHdr"/>
3265             <tp:Constituent tp:idref="Negolnit_CounterOfferRequest"/>
3266         </tp:Composite>
3267     </tp:CompositeList>
3268 </tp:Packaging>
3269 <!-- An ebXML message with a SOAP Envelope plus a counter offer response action payload -->
3270 <tp:Packaging tp:id="Negolnit_CounterOfferResponsePackage">
3271     <tp:ProcessingCapabilities tp:parse="true" tp:generate="true"/>
3272     <tp:CompositeList>
3273         <tp:Composite tp:id="Negolnit_CounterOfferResponseAcceptMsgId" tp:mimetype="multipart/related"
3274 tp:mimeparameters="type=text/xml">
3275             <tp:Constituent tp:idref="Negolnit_MsgHdr"/>
3276             <tp:Constituent tp:idref="Negolnit_OfferAccept"/>
3277         </tp:Composite>
3278     </tp:CompositeList>
3279     <tp:CompositeList>
3280         <tp:Composite tp:id="Negolnit_CounterOfferResponseRejectMsgId" tp:mimetype="multipart/related"
3281 tp:mimeparameters="type=text/xml">
3282             <tp:Constituent tp:idref="Negolnit_MsgHdr"/>
3283             <tp:Constituent tp:idref="Negolnit_OfferReject"/>
3284         </tp:Composite>

```

```

3285     </tp:CompositeList>
3286     <tp:CompositeList>
3287         <tp:Composite tp:id="Negolnit_CounterOfferResponsePendingMsgId" tp:mimetype="multipart/related"
3288 tp:mimeparameters="type=text/xml">
3289             <tp:Constituent tp:idref="Negolnit_MsgHdr"/>
3290             <tp:Constituent tp:idref="Negolnit_OfferCounterPending"/>
3291         </tp:Composite>
3292     </tp:CompositeList>
3293 </tp:Packaging>
3294 <!-- An ebXML message with a SOAP Envelope plus a Receipt Acknowledgment payload -->
3295 <tp:Packaging tp:id="Negolnit_ReceiptAcknowledgmentPackage">
3296     <tp:ProcessingCapabilities tp:parse="true" tp:generate="true"/>
3297     <tp:CompositeList>
3298         <tp:Composite tp:id="Negolnit_ReceiptAcknowledgmentMsg" tp:mimetype="multipart/related"
3299 tp:mimeparameters="type=text/xml">
3300             <tp:Constituent tp:idref="Negolnit_MsgHdr"/>
3301             <tp:Constituent tp:idref="Negolnit_ReceiptAcknowledgment"/>
3302         </tp:Composite>
3303     </tp:CompositeList>
3304 </tp:Packaging>
3305 <tp:Packaging tp:id="NegoResp_ReceiptAcknowledgmentPackage">
3306     <tp:ProcessingCapabilities tp:parse="true" tp:generate="true"/>
3307     <tp:CompositeList>
3308         <tp:Composite tp:id="NegoResp_ReceiptAcknowledgmentMsg" tp:mimetype="multipart/related"
3309 tp:mimeparameters="type=text/xml">
3310             <tp:Constituent tp:idref="NegoResp_MsgHdr"/>
3311             <tp:Constituent tp:idref="NegoResp_ReceiptAcknowledgment"/>
3312         </tp:Composite>
3313     </tp:CompositeList>
3314 </tp:Packaging>
3315 <!-- An ebXML message with a SOAP Envelope plus an Exception payload -->
3316 <tp:Packaging tp:id="Negolnit_ExceptionPackage">
3317     <tp:ProcessingCapabilities tp:parse="true" tp:generate="true"/>
3318     <tp:CompositeList>
3319         <tp:Composite tp:id="Negolnit_ExceptionMsg" tp:mimetype="multipart/related"
3320 tp:mimeparameters="type=text/xml">
3321             <tp:Constituent tp:idref="Negolnit_MsgHdr"/>
3322             <tp:Constituent tp:idref="Negolnit_Exception"/>
3323         </tp:Composite>
3324     </tp:CompositeList>
3325 </tp:Packaging>
3326 <tp:Packaging tp:id="NegoResp_ExceptionPackage">
3327     <tp:ProcessingCapabilities tp:parse="true" tp:generate="true"/>
3328     <tp:CompositeList>
3329         <tp:Composite tp:id="NegoResp_ExceptionMsg" tp:mimetype="multipart/related"
3330 tp:mimeparameters="type=text/xml">
3331             <tp:Constituent tp:idref="NegoResp_MsgHdr"/>
3332             <tp:Constituent tp:idref="NegoResp_Exception"/>
3333         </tp:Composite>
3334     </tp:CompositeList>
3335 </tp:Packaging>
3336     <tp:Comment xml:lang="en-US">CPPA negotiation between Negolnit.com and NegoResp.com</tp:Comment>
3337 </tp:CollaborationProtocolAgreement>

```

3338 **Appendix D Negotiation BPSS Instance Document**
 3339 **(Normative)**

3340 The text file for this example of the BPSS instance document for automated negotiation is
 3341 available at:

3342 ***THIS APPENDIX HAS BEEN REPLACED IN FULL.***

3343 ***THIS SHOULD BE REPLACED BY A "PRETTY PRINTING" VERSION.***

```

3346
3347 <?xml version="1.0" encoding="UTF-8"?>
3348 <ProcessSpecification xmlns="http://www.ebxml.org/BusinessProcess"
3349 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
3350 xsi:schemaLocation="http://www.ebxml.org/BusinessProcess ebBPSS1.04.xsd"
3351 name="CPPA-Negotiation" uuid="bpid:ebXML:CPPA-Negotiation" version="R02.00">
3352   <Documentation>This business process describes CPPA negotiation
3353 process</Documentation>
3354   <!--CPA Offer Document-->
3355   <BusinessDocument name="CPA Offer Doc" nameID="CPA_Offer_Doc"/>
3356   <!--CPA Acceptt Offer Document-->
3357   <BusinessDocument name="CPA Accept Offer Doc"
3358 nameID="CPA_Accept_Offer_Doc"/>
3359   <!--CPA Counter Pending Offer Document-->
3360   <BusinessDocument name="CPA Counter Pending Offer Doc"
3361 nameID="CPA_Counter_Pending_Offer_Doc"/>
3362   <!--CPA Counter Offer Document-->
3363   <BusinessDocument name="CPA Counter Offer Doc"
3364 nameID="CPA_Counter_Offer_Doc"/>
3365   <!--CPA Reject Offer Document-->
3366   <BusinessDocument name="CPA Reject Offer Doc"
3367 nameID="CPA_Reject_Offer_Doc"/>
3368   <!--Changed 09/16 CPA Document. This will probably come from the CPA
3369 specification-->
3370   <BusinessDocument name="CPA Final Doc" nameID="CPA_Final_Doc"/>
3371   <!--Changed 09/16 . Response to final CPA Document. This will probably
3372 come from the CPA specification
3373   This is used when the CPA is not signed just to show acceptance
3374 or denial of final CPA-->
3375   <BusinessDocument name="CPA Final Response Doc"
3376 nameID="CPA_Final_Response_Doc"/>
3377   <!--Changed 09/16 . Response to final CPA Document which is signed and
3378 agreed to create a double singed CPA. Receiving party will create a Signature
3379 over the signed CPA and send that. This will probably come from the CPA
3380 specification-->
3381   <BusinessDocument name="CPA Final Response Doc Signed"
3382 nameID="CPA_Final_Response_Doc_Signed"/>
3383   <!--Changed 02/28. Response to final CPA Document Could be rejecting
3384 the final CPA cause it's different
3385 from the agreed upon CPA or signature does not verify-->
3386   <BusinessDocument name="CPA Final Response Reject Doc"
3387 nameID="CPA_Final_Response_Reject_Doc"/>
3388
  
```

```

3389      <!-- Changed 09/16. Business Transaction for sending the CPA. This CPA
3390 is sent by the party finally accepting the offer-->
3391      <BusinessTransaction name="CPA Final BT" nameID="CPA_Final_BT">
3392          <RequestingBusinessActivity name="Final_CPA_BT_ReqBA"
3393 nameID="Final_CPA_BT_ReqBA" isAuthorizationRequired="false"
3394 isIntelligibleCheckRequired="false" isNonRepudiationReceiptRequired="false"
3395 isNonRepudiationRequired="false">
3396              <DocumentEnvelope businessDocument="CPA Final Doc"
3397 businessDocumentIDRef="CPA_Final_Doc" isAuthenticated="none"
3398 isConfidential="none" isTamperProof="none"/>
3399          </RequestingBusinessActivity>
3400          <RespondingBusinessActivity name="Final_CPA_BT_RespBA"
3401 nameID="Final_CPA_BT_RespBA" isAuthorizationRequired="false"
3402 isIntelligibleCheckRequired="false" isNonRepudiationRequired="false">
3403              <DocumentEnvelope businessDocument="CPA Final Response Doc"
3404 businessDocumentIDRef="CPA_Final_Response_Doc" isAuthenticated="none"
3405 isConfidential="none" isPositiveResponse="true" isTamperProof="none"/>
3406              <DocumentEnvelope businessDocument="CPA Final Response Doc
3407 Signed" businessDocumentIDRef="CPA_Final_Response_Doc_Signed"
3408 isAuthenticated="none" isConfidential="none" isPositiveResponse="true"
3409 isTamperProof="none"/>
3410              <DocumentEnvelope businessDocument="CPA Final Response
3411 Reject Doc" businessDocumentIDRef="CPA_Final_Response_Reject_Doc"
3412 isAuthenticated="none" isConfidential="none" isPositiveResponse="false"
3413 isTamperProof="none"/>
3414          </RespondingBusinessActivity>
3415      </BusinessTransaction>
3416      <!-- Business Transaction for the original negotiation cpa -->
3417      <BusinessTransaction name="CPA Offer BT" nameID="CPA_Offer_BT">
3418          <RequestingBusinessActivity name="CPA_Offer_BT_ReqBA"
3419 nameID="CPA_Offer_BT_ReqBA" isAuthorizationRequired="false"
3420 isIntelligibleCheckRequired="false" isNonRepudiationReceiptRequired="false"
3421 isNonRepudiationRequired="false">
3422              <DocumentEnvelope businessDocument="CPA Offer Doc"
3423 businessDocumentIDRef="CPA_Offer_Doc" isAuthenticated="none"
3424 isConfidential="none" isTamperProof="none"/>
3425          </RequestingBusinessActivity>
3426          <RespondingBusinessActivity name="CPA_Offer_BT_RespBA"
3427 nameID="CPA_Offer_BT_RespBA" isAuthorizationRequired="false"
3428 isIntelligibleCheckRequired="false" isNonRepudiationRequired="false">
3429              <DocumentEnvelope businessDocument="CPA Accept Offer Doc"
3430 businessDocumentIDRef="CPA_Accept_Offer_Doc" isAuthenticated="none"
3431 isConfidential="none" isPositiveResponse="true" isTamperProof="none"/>
3432              <DocumentEnvelope businessDocument="CPA Reject Offer Doc"
3433 businessDocumentIDRef="CPA_Reject_Offer_Doc" isAuthenticated="none"
3434 isConfidential="none" isPositiveResponse="false" isTamperProof="none"/>
3435              <DocumentEnvelope businessDocument="CPA Counter Pending
3436 Offer Doc" businessDocumentIDRef="CPA_Counter_Pending_Offer_Doc"
3437 isAuthenticated="none" isConfidential="none" isPositiveResponse="true"
3438 isTamperProof="none"/>
3439          </RespondingBusinessActivity>
3440      </BusinessTransaction>
3441      <!-- Business Transaction for sending the counter offer -->
3442      <BusinessTransaction name="CPA Counter Offer BT"
3443 nameID="CPA_Counter_Offer_BT">

```

```

3445     <RequestingBusinessActivity name="CPA_Counter_Offer_BT_ReqBA"
3446 nameID="CPA_Counter_Offer_BT_ReqBA" isAuthorizationRequired="false"
3447 isIntelligibleCheckRequired="false" isNonRepudiationReceiptRequired="false"
3448 isNonRepudiationRequired="false">
3449         <DocumentEnvelope businessDocument="CPA Counter Offer Doc"
3450 businessDocumentIDRef="CPA_Counter_Offer_Doc" isAuthenticated="none"
3451 isConfidential="none" isTamperProof="none"/>
3452     </RequestingBusinessActivity>
3453     <RespondingBusinessActivity name="CPA_Counter_Offer_BT_RespBA"
3454 nameID="CPA_Counter_Offer_BT_RespBA" isAuthorizationRequired="false"
3455 isIntelligibleCheckRequired="false" isNonRepudiationRequired="false">
3456         <DocumentEnvelope businessDocument="CPA Accept Offer Doc"
3457 businessDocumentIDRef="CPA_Accept_Offer_Doc" isAuthenticated="none"
3458 isConfidential="none" isPositiveResponse="true" isTamperProof="none"/>
3459         <DocumentEnvelope businessDocument="CPA Reject Offer Doc"
3460 businessDocumentIDRef="CPA_Reject_Offer_Doc" isAuthenticated="none"
3461 isConfidential="none" isPositiveResponse="false" isTamperProof="none"/>
3462         <DocumentEnvelope businessDocument="CPA Counter Pending
3463 Offer Doc" businessDocumentIDRef="CPA_Counter_Pending_Offer_Doc"
3464 isAuthenticated="none" isConfidential="none" isPositiveResponse="true"
3465 isTamperProof="none"/>
3466     </RespondingBusinessActivity>
3467 </BusinessTransaction>
3468 <!-- Main collaboration for negotiation business process -->
3469 <BinaryCollaboration name="CPA Negotiation BC"
3470 nameID="CPA_Negotiation_BC" initiatingRole="CPA_Negotiation_Initiator_Role">
3471     <!-- Role for initiator for negotiation process -->
3472     <Role name="CPA Negotiation Initiator"
3473 nameID="CPA_Negotiation_Initiator_Role"/>
3474     <!-- Role for initial responder of business collaboration -->
3475     <Role name="CPA Negotiation Responder"
3476 nameID="CPA_Negotiation_Responder_Role"/>
3477     <Start toBusinessState="CPA_Offer_BTA"/>
3478     <BusinessTransactionActivity name="CPA Offer BTA"
3479 nameID="CPA_Offer_BTA" businessTransaction="CPA Offer BT"
3480 businessTransactionIDRef="CPA_Offer_BT" fromRole="CPA Negotiation Initiator"
3481 fromRoleIDRef="CPA_Negotiation_Initiator_Role" toRole="CPA Negotiation
3482 Responder" toRoleIDRef="CPA_Negotiation_Responder_Role"
3483 isLegallyBinding="false" isConcurrent="false"/>
3484     <CollaborationActivity name="CPA Counter Offer CA"
3485 binaryCollaboration="CPA Negotiation Counter Offer BC"
3486 binaryCollaborationIDRef="CPA_Negotiation_CounterOfferBC" fromRole="CPA
3487 Negotiation Counter Offer Initiator"
3488 fromRoleIDRef="CPA_Negotiation_CounterOfferInitiator_Role" toRole="CPA
3489 Negotiation Counter Offer Responder"
3490 toRoleIDRef="CPA_Negotiation_CounterOfferResponder_Role"
3491 precondition="Initiating Role for this activity corresponds to Responding
3492 Role in CPA Offer BTA"/>
3493     <BusinessTransactionActivity name="CPA Final BTA"
3494 nameID="CPA_Final_BTA" businessTransaction="CPA Final BT"
3495 businessTransactionIDRef="CPA_Final_BT" fromRole="CPA Negotiation Responder"
3496 fromRoleIDRef="CPA_Negotiation_Responder_Role" toRole="CPA Negotiation
3497 Initiator" toRoleIDRef="CPA_Negotiation_Initiator_Role"
3498 isLegallyBinding="false" isConcurrent="false"/>
3499     <!-- If final CPA BTA goes through fine, then overall
3500 collaboration is marked success -->

```

```

3501         <Success fromBusinessState="CPA Final BTA"
3502 conditionGuard="Success"/>
3503         <!-- If inner collaboration goes through fine, then overall
3504 collaboration is marked success. Inner collaboration
3505             Would have gone through the transaction that ends up with
3506 either the final CPA (Signed if needed) -->
3507         <Success fromBusinessState="CPA Counter Offer CA"
3508 conditionGuard="Success"/>
3509         <!-- If Reject offer document is sent for offer bta collaboration
3510 is marked as failure-->
3511         <Failure fromBusinessState="CPA Offer BTA"
3512 conditionGuard="BusinessFailure">
3513             <ConditionExpression
3514 expressionLanguage="DocumentEnvelopeLanguage" expression="CPA Reject Offer
3515 Doc"/>
3516         </Failure>
3517         <!-- Changed 02/28. If final CPA is rejected then its a failure
3518 too. -->
3519         <Failure fromBusinessState="CPA Final BTA"
3520 conditionGuard="Failure">
3521             <ConditionExpression
3522 expressionLanguage="DocumentEnvelopeLanguage" expression="CPA Final Response
3523 Reject Doc"/>
3524         </Failure>
3525         <!-- If Final CPA BTA fails for some reason, then collaboration
3526 is marked as failure -->
3527         <Failure fromBusinessState="CPA Final BTA"
3528 conditionGuard="Failure"/>
3529         <Failure fromBusinessState="CPA Counter Offer CA"
3530 conditionGuard="Failure"/>
3531         <!-- Transition to Final CPA offer binary Transaction if the
3532 responder for main transaction accepts the initial offer -->
3533         <Transition fromBusinessState="CPA Offer BTA"
3534 toBusinessState="CPA Final BTA">
3535             <ConditionExpression
3536 expressionLanguage="DocumentEnvelopeLanguage" expression="CPA Accept Offer
3537 Doc"/>
3538         </Transition>
3539         <!-- Transition to counter offer binary collaboration if the
3540 responder for main transaction returns a counter offer pending message -->
3541         <Transition fromBusinessState="CPA Offer BTA"
3542 toBusinessState="CPA Counter Offer CA">
3543             <ConditionExpression
3544 expressionLanguage="DocumentEnvelopeLanguage" expression="CPA Counter Pending
3545 Offer Doc"/>
3546         </Transition>
3547     </BinaryCollaboration>
3548     <BinaryCollaboration name="CPA Negotiation Counter Offer BC"
3549 nameID="CPA_Negotiation_CounterOfferBC"
3550 initiatingRole="CPA_Negotiation_CounterOfferInitiator_Role">
3551         <Role name="CPA Negotiation Counter Offer Initiator"
3552 nameID="CPA_Negotiation_CounterOfferInitiator_Role"/>
3553         <Role name="CPA Negotiation Counter Offer Responder"
3554 nameID="CPA_Negotiation_CounterOfferResponder_Role"/>
3555         <!-- This collaboration starts with the negotiation process
3556 responder sending the counter offer -->
3557         <Start toBusinessState="CPA Counter Offer 1 BTA"/>

```

```
3558         <!-- This transaction activity is for negotiation process
3559 responder sending the counter offer -->
3560         <BusinessTransactionActivity name="CPA Counter Offer 1 BTA"
3561 nameID="CPA_Counter_Offer_1_BTA" businessTransaction="CPA Counter Offer BT"
3562 businessTransactionIDRef="CPA_Counter_Offer_BT" fromRole="CPA Negotiation
3563 Counter Offer Initiator"
3564 fromRoleIDRef="CPA_Negotiation_CounterOfferInitiator_Role" toRole="CPA
3565 Negotiation Counter Offer Non Initiator"
3566 toRoleIDRef="CPA_Negotiation_CounterOfferResponder_Role"
3567 isLegallyBinding="false" isConcurrent="false" postCondition="Parties reverse
3568 roles they play"/>
3569         <!-- This transaction acitivity is for negotiation process
3570 initiator sending the counter offer -->
3571         <BusinessTransactionActivity name="CPA Counter Offer 2 BTA"
3572 nameID="CPA_Counter_Offer_2_BTA" businessTransaction="CPA Counter Offer BT"
3573 businessTransactionIDRef="CPA_Counter_Offer_BT" fromRole="CPA Negotiation
3574 Counter Offer Responder"
3575 fromRoleIDRef="CPA_Negotiation_CounterOfferResponder_Role" toRole="CPA
3576 Negotiation Counter Offer Initiator"
3577 toRoleIDRef="CPA_Negotiation_CounterOfferInitiator_Role"
3578 isLegallyBinding="false" isConcurrent="false" postCondition="Parties reverse
3579 roles they play"/>
3580         <BusinessTransactionActivity name="CPA Final BTA Init Initiator"
3581 nameID="CPA_Final_BTA_init_Initiator" businessTransaction="CPA Final BT"
3582 businessTransactionIDRef="CPA_Final_BT" fromRole="CPA Negotiation Counter
3583 Offer Initiator" fromRoleIDRef="CPA_Negotiation_CounterOfferInitiator_Role"
3584 toRole="CPA Negotiation Counter Offer Responder"
3585 toRoleIDRef="CPA_Negotiation_CounterOfferResponder_Role"
3586 isLegallyBinding="false" isConcurrent="false"/>
3587         <BusinessTransactionActivity name="CPA Final BTA Init Responder"
3588 nameID="CPA_Final_BTA_init_Responder" businessTransaction="CPA Final BT"
3589 businessTransactionIDRef="CPA_Final_BT" fromRole="CPA Negotiation Counter
3590 Offer Responder" fromRoleIDRef="CPA_Negotiation_CounterOfferResponder_Role"
3591 toRole="CPA Negotiation Counter Offer Initiator"
3592 toRoleIDRef="CPA_Negotiation_CounterOfferInitiator_Role"
3593 isLegallyBinding="false" isConcurrent="false"/>
3594         <!-- Inner collaboration succeeds if the final BTA which involves
3595 sending final CPA succeeds -->
3596         <Success fromBusinessState="CPA Final BTA Init Initiator"
3597 conditionGuard="Success"/>
3598         <!-- Inner collaboration succeeds if the final BTA which involves
3599 sending final CPA succeeds. This is
3600 the same as above but the difference is this initiated by a
3601 different party -->
3602         <Success fromBusinessState="CPA Final BTA Init Responder"
3603 conditionGuard="Success"/>
3604         <!-- Changed 02/28. If final CPA is rejected then its a failure
3605 too. -->
3606         <Failure fromBusinessState="CPA Final BTA Init Initiator"
3607 conditionGuard="BusinessFailure">
3608             <ConditionExpression
3609 expressionLanguage="DocumentEnvelopeLanguage" expression="CPA Final Response
3610 Reject Doc"/>
3611         </Failure>
3612         <Failure fromBusinessState="CPA Final BTA Init Responder"
3613 conditionGuard="BusinessFailure">
```

```

3614         <ConditionExpression
3615 expressionLanguage="DocumentEnvelopeLanguage" expression="CPA Final Response
3616 Reject Doc"/>
3617         </Failure>
3618         <!-- Inner collaboration fails if the final BTA which involves
3619 sending final CPA fails -->
3620         <Failure fromBusinessState="CPA Final BTA Init Initiator"
3621 conditionGuard="Failure"/>
3622         <!-- Inner collaboration fails if the final BTA which involves
3623 sending final CPA fails. This is
3624 the same as above but the difference is this initiated by a
3625 different party -->
3626         <Failure fromBusinessState="CPA Final BTA Init Responder"
3627 conditionGuard="Failure"/>
3628         <Failure fromBusinessState="CPA Counter Offer 1 BTA"
3629 conditionGuard="BusinessFailure">
3630         <ConditionExpression
3631 expressionLanguage="DocumentEnvelopeLanguage" expression="CPA Reject Offer
3632 Doc"/>
3633         </Failure>
3634         <Failure fromBusinessState="CPA Counter Offer 2 BTA"
3635 conditionGuard="BusinessFailure">
3636         <ConditionExpression
3637 expressionLanguage="DocumentEnvelopeLanguage" expression="CPA Reject Offer
3638 Doc"/>
3639         </Failure>
3640         <!-- If the negotiation process responder (initiator in this
3641 innercollaboration) sends an acceptace offer, negotiation process responder
3642 sends the final CPA -->
3643         <Transition fromBusinessState="CPA Counter Offer 2 BTA"
3644 toBusinessState="CPA Final BTA Init Initiator">
3645         <ConditionExpression
3646 expressionLanguage="DocumentEnvelopeLanguage" expression="CPA Accept Offer
3647 Doc"/>
3648         </Transition>
3649         <!-- If the negotiation process initiator (responder in this
3650 inner collaboration) sends an acceptace offer, negotiation process initiator
3651 sends the final CPA -->
3652         <Transition fromBusinessState="CPA Counter Offer 1 BTA"
3653 toBusinessState="CPA Final BTA Init Responder">
3654         <ConditionExpression
3655 expressionLanguage="DocumentEnvelopeLanguage" expression="CPA Accept Offer
3656 Doc"/>
3657         </Transition>
3658         <!-- If the negotiation process responder sends counter offer and
3659 negotiation process initiator sends a counter offer, negotiation process
3660 initiator sends the counter offer next time -->
3661         <Transition fromBusinessState="CPA Counter Offer 1 BTA"
3662 toBusinessState="CPA Counter Offer 2 BTA">
3663         <ConditionExpression
3664 expressionLanguage="DocumentEnvelopeLanguage" expression="CPA Counter Pending
3665 Offer Doc"/>
3666         </Transition>
3667         <!-- If the negotiation process initiator sends a counter offer
3668 and negotiation process responds sends a counter offer, negotiation process
3669 responder sends the counter offer next time, hence the transition back to
3670 original BTA-->

```

```
3671         <Transition fromBusinessState="CPA Counter Offer 2 BTA"  
3672 toBusinessState="CPA Counter Offer 1 BTA">  
3673             <ConditionExpression  
3674 expressionLanguage="DocumentEnvelopeLanguage" expression="CPA Counter Pending  
3675 Offer Doc"/>  
3676         </Transition>  
3677     </BinaryCollaboration>  
3678 </ProcessSpecification>  
3679
```

3680 **Appendix E Instance Documents for Business Signals**

3681 The XML Schemas of the business signals are defined in [ebBPSS].

3682 **Acceptance Acknowledgment**

3683 The instance document for the AcceptanceAcknowledgment business signal is available as a text
3684 file at:

3685 **Exception**

3686 The instance document for the Exception business signal is available as a text file at:

3687 **Appendix F Example of NDD Instance Document (Non-**
 3688 **Normative)**

3689 The text file for this example of an *NDD* instance document for automated negotiation is
 3690 available at:

3691

3692 [THIS APPENDIX HAS BEEN COMPLETELY REPLACED.](#)

3693

3694 `<?xml version="1.0" encoding="UTF-8"?>`

3695 `<!-- edited with XML Spy v4.4 U (http://www.xmlspy.com) by neelakantan kartha (Sterling Commerce) -->`

3696 `<!-- edited by neelakantan kartha (Sterling Commerce) -->`

3697 `<NegotiationDescriptor xmlns="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpp-cpa-negot-2_0.xsd"`

3698 `xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://www.oasis-`

3699 `open.org/committees/ebxml-cppa/schema/cpp-cpa-2_0.xsd`

3700 `NDD11.xsd" xmlns:xsd="http://www.w3.org/2001/XMLSchema" documentLocation="C:\Documents and`

3701 `Settings\Inkartha\My Documents\ebxml\negotiation\cpa-example-2_0a.xml">`

3702 `<!--The value of cpaid might be negotiable, since a party might require that the cpaid conform to a particular format.`

3703 `However, automatic negotiation on the values is difficult. For instance, how does a party convey to the other party the kinds`

3704 `of cpaid that it deems o.k? Without this information being conveyed somehow, it will be difficult to come to agreement`

3705 `automatically. Fortunately, this issue pertains to the negotiation algorithm and not the NDD. For version 1, we can stipulate`

3706 `that the cpaid must be URI. The mustBeFilledIn attribute signals to the other party that this value must be filled in. This`

3707 `attribute, which is optional, is often useful in situations (maybe other than cpaid), where the dominant partner fills out most of`

3708 `the CPA and the only thing that is required of the other partner is to fill in some information. -->`

3709 `<NegotiableInformationItem xpath="/CollaborationProtocolAgreement/@cpaid">`

3710 `<Value mustBeFilledIn="Yes"/>`

3711 `</NegotiableInformationItem>`

3712 `<!-- Versions might be negotiable, since one party might have a product that conforms to an earlier version of the spec.`

3713 `By an OrderedValue, it is implied that there is a preference to the earlier values or later values, as given by the attribute`

3714 `preferenceOrder-->`

3715 `<NegotiableInformationItem xpath="/CollaborationProtocolAgreement/@version">`

3716 `<OrderedValue preference="EarlierPreferred">`

3717 `<Value> 1.0 </Value>`

3718 `<Value> 2_0.a</Value>`

3719 `</OrderedValue>`

3720 `</NegotiableInformationItem>`

3721 `<!--The value can be one of the following three: proposed, agreed and signed. However, I do not think that this attribute`

3722 `is negotiated--rather one party might set the value of this attribute to agreed, once it is satisfied that a satisfactory CPA has`

3723 `been reached. Again, how and when this attribute is set seems part of the negotiation algorithm.`

3724 `Also, the possible values of this attribute are part of the CPPA schema--hence there is no need to repeat them here -->`

3725 `<NegotiableInformationItem xpath="/CollaborationProtocolAgreement/Status/@value">`

3726 `<Value/>`

3727 `</NegotiableInformationItem>`

3728 `<!--Here, I am taking the (simplistic) assumption that each party specifies the earliest time for starting and the latest time`

3729 `for ending the Start element (that specifies the Starting Date and Time for the CPA). No preference function is given. See the`

3730 `next entry for an example of how one would encode a piecewise linear preference function.-->`

3731 `<NegotiableInformationItem xpath="/CollaborationProtocolAgreement/Start">`

3732 `<ValueWithPreferenceMeasure>`

3733 `<EndPoints>`

3734 `<EarliestStart>1998-01-31T14:20:00.011Z</EarliestStart>`

3735 `<LatestEnd>2002-11-21T14:20:00.000-05:00</LatestEnd>`

3736 `</EndPoints>`

3737 `</ValueWithPreferenceMeasure>`

3738 `</NegotiableInformationItem>`

3739 `<!--Nothing new here, when compared to the previous NegotiableInformationItem except that this gives an example of a`

3740 `piecewise linear preference function. I suspect that this might be over engineering at this point 1998-04-07T18:39:09Z t -->`

3741 `<NegotiableInformationItem xpath="/CollaborationProtocolAgreement/End">`

3742 `<ValueWithPreferenceMeasure>`

3743 `<EndPoints>`

3744 `<EarliestStart>1998-04-07T00:18:39.090Z</EarliestStart>`

3745 `<LatestEnd>2002-11-31T13:20:00.000-05:00</LatestEnd>`

```

3746     </EndPoints>
3747     <PreferenceFunction>
3748       <PiecewiseLinearPiece>
3749         <x-coordinate1>1998-04-07T18:39:09Z </x-coordinate1>
3750         <y-coordinate1>15</y-coordinate1>
3751         <x-coordinate2>2000-11-31T13:20:00.000-05:00 </x-coordinate2>
3752         <y-coordinate2>30</y-coordinate2>
3753       </PiecewiseLinearPiece>
3754     <PiecewiseLinearPiece>
3755       <x-coordinate1>2000-11-31T13:20:00.000-05:00 </x-coordinate1>
3756       <y-coordinate1>30</y-coordinate1>
3757       <x-coordinate2>2002-11-31T13:20:00.000-05:00 </x-coordinate2>
3758       <y-coordinate2>60</y-coordinate2>
3759     </PiecewiseLinearPiece>
3760   </PreferenceFunction>
3761 </ValueWithPreferenceMeasure>
3762 </NegotiableInformationItem>
3763 <!--Note the ConversationConstraints is an element that might be present or absent, and hence a party might negotiate
3764 the presence or absence of this element. There are four cases to consider A party (a) insists that an element must be
3765 present (b) insists that an element is absent (c) is ok with the element being present or absent, but has a preference for one
3766 or the other (d) is o.k with the element being present or absent, and is completely agnostic.-->
3767 <NegotiableInformationItem xpath="/CollaborationProtocolAgreement/ConversationConstraints">
3768   <PresentOrNot value="MustBePresent"/>
3769 </NegotiableInformationItem>
3770 <!-- Note that invocationLimit is an attribute of ConversationConstraints that may or may not be present. So first of all,
3771 the presence or absence of this attribute may be negotiable. Then, the value of this attribute may also be negotiable -->
3772 <NegotiableInformationItem xpath="/CollaborationProtocolAgreement/ConversationConstraints/@invocationLimit">
3773   <IntegerValues>
3774     <RangeInfo preferenceOrder="SmallerPreferred">
3775       <EndPoints>
3776         <SmallestValue>1</SmallestValue>
3777         <LatestValue>5</LatestValue>
3778       </EndPoints>
3779     </RangeInfo>
3780   </IntegerValues>
3781 </NegotiableInformationItem>
3782 <!--concurrentConversations is similar to invocationLimit. I am including this just for the sake of illustrating another use of
3783 the schema -->
3784 <NegotiableInformationItem
3785 xpath="/CollaborationProtocolAgreement/ConversationConstraints/@concurrentConversations">
3786   <IntegerValues>
3787     <PresentOrNot value="MustBePresent"/>
3788     <RangeInfo>
3789       <EndPoints>
3790         <SmallestValue>2</SmallestValue>
3791         <LatestValue>8</LatestValue>
3792       </EndPoints>
3793       <PreferenceFunction>
3794         <FunctionDefinedByEquation> x**2-2*x+3</FunctionDefinedByEquation>
3795       </PreferenceFunction>
3796     </RangeInfo>
3797   </IntegerValues>
3798 </NegotiableInformationItem>
3799 <!--The partyInfo element raises a number of interesting issues. In the CPA, there can be exactly two partyInfo
3800 elements, hence there is no negotiation on these once a CPA has been formed. Since this NDD refers to
3801 a CPA, there can be no negotiation on this element. We will forget this for the time being and see what would be the case if
3802 the document referred to by this NDD were a CPP. A CPP can have multiple PartyInfo elements and one among these must
3803 be chosen to form the CPA. Thus the issue here is to associate a preference order between several elements at the same
3804 level. A simple way of doing this in a CPP is as follows: -->
3805 <NegotiableInformationItem xpath="/CollaborationProtocolProfile/PartyInfo[2]">
3806   <Preference value="1"/>
3807 </NegotiableInformationItem>
3808 <NegotiableInformationItem xpath="/CollaborationProtocolProfile/PartyInfo[1]">
3809   <Preference value="3"/>
3810 </NegotiableInformationItem>
3811 <NegotiableInformationItem xpath="/CollaborationProtocolProfile/PartyInfo[3]">

```

```

3812     <Preference value="2"/>
3813 </NegotiableInformationItem>
3814 <!--This is included to provide an example where the cardinality of an element may be negotiable. It might be the case
3815 that the number of PartyId elements within a partyInfo element is negotiable (because, say of limitations the underlying
3816 system has of handling a large number of partyIds)-->
3817     <NegotiableInformationItem xpath="/CollaborationProtocolAgreement/PartyInfo/PartyId">
3818         <Cardinality>
3819             <RangeInfo>
3820                 <EndPoints>
3821                     <SmallestValue>1</SmallestValue>
3822                     <LatestValue>5</LatestValue>
3823                 </EndPoints>
3824             </RangeInfo>
3825         </Cardinality>
3826     </NegotiableInformationItem>
3827 <!--Example of a boolean value-->
3828     <NegotiableInformationItem
3829 xpath="/CollaborationProtocolAgreement/PartyInfo/CollaborationRole/ServiceBinding/Service/CanSend/ThisPartyActionBind
3830 ing/BusinessTransactionCharacteristics/@isNonRepudiationRequired">
3831         <BooleanValue preference="TruePreferred">
3832             <PresentOrNot value="MustBePresent"/>
3833         </BooleanValue>
3834     </NegotiableInformationItem>
3835 <!--Example of negotiating a duration-->
3836     <NegotiableInformationItem
3837 xpath="/CollaborationProtocolAgreement/PartyInfo/CollaborationRole/ServiceBinding/Service/CanSend/ThisPartyActionBind
3838 ing/BusinessTransactionCharacteristics/@timeToAcknowledgeReceipt">
3839         <DurationWithPreference preferenceOrder="SmallerPreferred">
3840             <MinimumDuration>PT5M</MinimumDuration>
3841             <MaximumDuration>PT6M</MaximumDuration>
3842         </DurationWithPreference>
3843     </NegotiableInformationItem>
3844 <!--This is how would express that one element of an enumeration must be present. Note that the possible values of the
3845 enumeration is defined in the cpp-cpa schema and need not be repeated here-->
3846     <NegotiableInformationItem
3847 xpath="/CollaborationProtocolAgreement/PartyInfo/DeliveryChannel/MessagingCharacteristics/@syncReplyMode">
3848         <OrderedValue>
3849             <PresentOrNot value="MustBePresent"/>
3850             <Value> signalsOnly</Value>
3851         </OrderedValue>
3852     </NegotiableInformationItem>
3853 </NegotiationDescriptor>
3854 <!--Notes
3855 0. The top element of an NDD document is named NegotiationDescriptor. The NegotiationDescriptor element contains
3856 NegotiationInformationItem elements for each item that is negotiable. (
3857 1. The documentLocation attribute of NegotiationDescriptor element is a uri that points to the document for which this >
3858 NDD document pertains to. In particular, the xpath attribute of a NegotiableInformationItem element is an xpath of
3859 this document. The documentLocation attribute is a required attribute.

```

- 3860
3861
3862
3863
3864 Non-Negotiable elements and Attributes
3865 =====
- 3866 1. CollaborationProtocolAgreement
 - 3867 2. CollaborationProtocolAgreement/@schemaLocation
 - 3868 3. CollaborationProtocolAgreement/Status
 - 3869 4. CollaborationProtocolAgreement/PartyInfo/@partyName (Since this is set by each party, it is difficult to see how this
3870 would be negotiable. If it is, it would be similar to /CollaborationProtocolAgreement/@cpaid)
 - 3871 5. CollaborationProtocolAgreement/PartyInfo/@defaultMshChannelId and
3872 CollaborationProtocolAgreement/PartyInfo/@defaultMsPackageId (Again, if these are negotiable, it would be as a result of
3873 the negotiation algorithm recognizing that the default values are not reasonable. Again, only the value can be negotiated, as
3874 in /CollaborationProtocolAgreement/@cpaid)
 - 3875 6. CollaborationProtocolAgreement/PartyInfo/PartyRef/@xlink:type(always simple)
 - 3876 7. CollaborationProtocolAgreement/PartyInfo/CollaborationRole/ProcessSpecification
 - 3877 8. CollaborationProtocolAgreement/PartyInfo/CollaborationRole/ProcessSpecification@name

Automated Negotiation Specification

3878 9. CollaborationProtocolAgreement/PartyInfo/CollaborationRole/ProcessSpecification@xlink:type (always simple)
3879 10.
3880 CollaborationProtocolAgreement/PartyInfo/CollaborationRole/ProcessSpecification/ds:Reference/ds:Trasforms/ds:Transform
3881 /@ds:Algorithm (fixed by the spec)
3882
3883 Elements and attributes similar to others in the sample
3884 =====
3885 0. CollaborationProtocolAgreement/PartyInfo/PartyId/@type (similar to /CollaborationProtocolAgreement/@version", with an
3886 enumeration that enumerates the range of understood naming systems)
3887 1. CollaborationProtocolAgreement/PartyInfo/PartyRef (similar to CollaborationProtocolAgreement/PartyInfo/PartyId)
3888 2. CollaborationProtocolAgreement/PartyInfo/PartyRef/@xlink:href (similar to /CollaborationProtocolAgreement/@cpaid)
3889 3. CollaborationProtocolAgreement/PartyInfo/PartyRef/@type (similar to CollaborationProtocolAgreement/@version)
3890 4. CollaborationProtocolAgreement/PartyInfo/PartyRef/@schemaLocation
3891 (similar to /CollaborationProtocolAgreement/@cpaid)
3892
3893 5. CollaborationProtocolAgreement/PartyInfo/CollaborationRole (similar to negotiating the cardinality of
3894 /CollaborationProtocolAgreement/PartyInfo/PartyId)
3895
3896 6. CollaborationProtocolAgreement/PartyInfo/CollaborationRole/ProcessSpecification@version (similar to
3897 /CollaborationProtocolAgreement/@version)
3898
3899 7. CollaborationProtocolAgreement/PartyInfo/CollaborationRole/ProcessSpecification@xlink:href and uid
3900 ((similar to /CollaborationProtocolAgreement/@cpaid)
3901
3902
3903 !->
3904

3905 **Appendix G Examples of Negotiation-Message Instance**
 3906 **Documents (Non-Normative)**

3907 **Example of Offer Message Instance Document**

3908 [THIS APPENDIX HAS BEEN REPLACED IN FULL.](#)

3909
 3910 Negotiation Message Offer

```

3911 <?xml version="1.0" encoding="UTF-8"?>
3912 <tp:NegotiationMessage xmlns:tp="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpa-negot-1_0.xsd"
3913 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:cppa="http://www.oasis-open.org/committees/ebxml-
3914 cppa/schema/cpp-cpa-2_0.xsd" xsi:schemaLocation="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpa-
3915 negot-1_0.xsd negotiationMsg.xsd" businessMsgId="busMsg001" binding="false" negotiationDialogId="negotDialog001"
3916 offerId="offer001" messageType="Offer">
3917 <tp:NCPA uri="http://www.companya.com/ncpa/myncpa.xml"/>
3918 <tp:CPATemplateId id="uri:companyA-and-companyB-CPA1" cppa:version="1.0"/>
3919 <cppa:SecurityDetails cppa:securityId="ID">
3920 <cppa:SecurityPolicy/>
3921 </cppa:SecurityDetails>
3922 <tp:InitiatingParty>
3923 <cppa:PartyId cppa:type="urn:oasis:names:tc:ebxml-cppa:partyid-type:duns">123456789</cppa:PartyId>
3924 <CPPId id="companya-cpp123456789" cppa:version="1.0"/>
3925 </tp:InitiatingParty>
3926 <tp:RespondingParty>
3927 <cppa:PartyId cppa:type="urn:oasis:names:tc:ebxml-cppa:partyid-type:duns">987654321</cppa:PartyId>
3928 <CPPId id="companyb-cpp987654321" cppa:version="1.0"/>
3929 </tp:RespondingParty>
3930 <tp:BPSSBusinessDocumentName name="CPA_Offer_Doc"/>
3931 <ExpirationDate>2002-12-20T00:00:00Z</ExpirationDate>
3932 <tp:BusinessDocuments>
3933 <CPATemplateDoc>
3934 <NDD>
3935 <Uri>http://www.companya.com/proposedncpa/ncpa.xml</Uri>
3936 </NDD>
3937 <CPATemplate>
3938 <Uri>http://www.companya.com/proposedcpa/companya-companyb-cpa1234.xml</Uri>
3939 </CPATemplate>
3940 </CPATemplateDoc>
3941 </tp:BusinessDocuments>
3942 <tp:NegotiationContent>
3943 <tp:AcceptedItem xpath="/"/>
3944 </tp:NegotiationContent>
3945 </tp:NegotiationMessage>
3946
3947
3948
```

3949 **Example of Counter-Offer Message Instance Document**

3950 The text file for the example of the counter-offer *Message* instance document is available at:

3951

3952 [THIS APPENDIX HAS BEEN REPLACED IN FULL.](#)

3953

3954 Negotiation Message Counter Offer

```

3955 <?xml version="1.0" encoding="UTF-8"?>
3956 <tp:NegotiationMessage xmlns:tp="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpa-negot-1_0.xsd"
3957 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:cppa="http://www.oasis-open.org/committees/ebxml-
3958 cppa/schema/cpp-cpa-2_0.xsd" xsi:schemaLocation="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpa-
```

```

3959 negot-1_0.xsd negotiationMsg.xsd" businessMsgId="busMsg002" binding="false" inresponseTo="busMsg001"
3960 negotiationDialogId="negotDialog001" offerId="offer001" messageType="CounterOffer">
3961   <tp:NCPA uri="http://www.companya.com/ncpa/myncpa.xml"/>
3962   <tp:CPATemplateId id="uri:companyA-and-companyB-CPA1" cppa:version="1.0"/>
3963   <cppa:SecurityDetails cppa:securityId="ID">
3964     <cppa:SecurityPolicy/>
3965   </cppa:SecurityDetails>
3966   <tp:InitiatingParty>
3967     <cppa:PartyId cppa:type="urn:oasis:names:tc:ebxml-cppa:partyid-type:duns">123456789</cppa:PartyId>
3968     <CPPId id="companya-cpp123456789" cppa:version="1.0"/>
3969   </tp:InitiatingParty>
3970   <tp:RespondingParty>
3971     <cppa:PartyId cppa:type="urn:oasis:names:tc:ebxml-cppa:partyid-type:duns">987654321</cppa:PartyId>
3972     <CPPId id="companyb-cpp987654321" cppa:version="1.0"/>
3973   </tp:RespondingParty>
3974   <tp:BSSBusinessDocumentName name="CPA_Counter_Offer_Doc"/>
3975   <ExpirationDate>2003-05-20T00:00:00Z</ExpirationDate>
3976   <tp:BusinessDocuments>
3977     <CPATemplateDoc>
3978       <NDD>
3979         <Uri>http://www.companyb.com/proposedncpa/ncpa.xml</Uri>
3980       </NDD>
3981       <CPATemplate>
3982         <Uri>http://www.companyb.com/proposedcpa/companya-companyb-cpa1234.xml</Uri>
3983       </CPATemplate>
3984     </CPATemplateDoc>
3985   </tp:BusinessDocuments>
3986   <tp:Offer-CounterOffer-Acceptance-Time>2003-03-20T00:00:00Z</tp:Offer-CounterOffer-Acceptance-Time>
3987   <tp:NegotiationContent>
3988     <tp:AcceptedItem xpath="/CollaborationProtocolAgreement/PartyInfo[0]"/>
3989     <tp:AcceptedItem xpath="/CollaborationProtocolAgreement/PartyInfo[1]/Certificate"/>
3990     <tp:AcceptedItem xpath="/CollaborationProtocolAgreement/PartyInfo[1]/SecurityDetails"/>
3991     <tp:AcceptedItem xpath="/CollaborationProtocolAgreement/PartyInfo[1]/DeliveryChannel"/>
3992     <tp:AcceptedItem xpath="/CollaborationProtocolAgreement/PartyInfo[1]/Transport"/>
3993     <tp:AcceptedItem xpath="/CollaborationProtocolAgreement/PartyInfo[1]/DocExchange"/>
3994     <tp:UpdatedItem
3995       xpath="/CollaborationProtocolAgreement/PartyInfo[1]/CollaborationRole/ServiceBinding/Cansend[0]/ThisPartyActionBindin
3996       g/BusinessTransactionCharacteristics@isNonRepudiationRequired" originalValue="true" proposedValue="false"
3997       itemStatus="Preferred"/>
3998   </tp:NegotiationContent>
3999 </tp:NegotiationMessage>
4000
4001

```

4002 **Appendix H Glossary of Terms**

4003 This appendix contains definitions of terms created by this specification. For definitions of
4004 terms created by the CPPA Specification[ebCPP] and related terms that are part of the general
4005 ebXML vocabulary, see [ebCPP].

4006
4007 **CPA Negotiation Process:** The process by which a *Collaboration Protocol Agreement (CPA)* is
4008 formed based on information provided by two *Parties* interested doing *Business*. The
4009 *Negotiation Process* includes the *Negotiation Protocol*, defined in this specification, and the
4010 private negotiation process at each *Party*.

4011
4012 **CPA Template:** A *CPA Template* is a *CPA* with open fields. The schema for a *CPA Template* is
4013 the normal *CPP-CPA* schema. The means of identifying open fields in the *CPA Template* is
4014 defined in this specification.

4015
4016 **Negotiation BPSS Instance Document:** The XML instance document that defines the
4017 *Negotiation-Protocol* choreography. This XML instance document conforms to the ebXML
4018 *Business Process Specification Schema* specification[ebBPSS].

4019
4020 **Negotiation CPA (NCPA):** The *CPA* that governs the *Negotiation Protocol*.

4021
4022 **Negotiation Descriptor Document (NDD):** A *Negotiation Descriptor Document (NDD)*
4023 describes what is negotiable in a *CPP* or a *CPA Template*.

4024
4025 **Negotiation Dialog:** A single instance of the *Negotiation Protocol* that negotiates one *CPA*
4026 from the initial proposal until the *CPA* is successfully completed or the negotiation terminates
4027 without success.

4028
4029 **Negotiation-Dialog Identifier:** A unique identifier that distinguishes each *Negotiation Dialog*
4030 from all others that may be in progress between two *Parties*.

4031
4032 **Negotiation Message:** The *Negotiation Protocol* consists of exchanges of *Messages* that contain
4033 the details of offers and counter offers. This specification defines the schema and semantics of
4034 each *Message*.

4035
4036 **Negotiation Protocol:** The *Negotiation Protocol* defines the exchange of data between both
4037 parties in the negotiation (and perhaps with a negotiation service). The format of these *Messages*
4038 and the choreography of their exchanges are defined by a *Negotiation CPA* and its corresponding
4039 BPSS instance document.

4040
4041 **Offer Identifier:** The *Offer Identifier* is a unique identifier associated with each offer and
4042 counter offer.

4043