1

2
3

Automated Negotiation of Collaboration-Protocol Agreements Specification Version 0.04

4
5
6

7

OASIS ebXML Collaboration Protocol Profile and Agreement Technical Committee

8
9
10
11   Date TBD
12

Status of this Document

14
15   This document specifies an ebXML SPECIFICATION for the eBusiness community.
16
17   Distribution of this document is unlimited.
18
19   The document formatting is based on the Internet Society's Standard RFC format.
20
21   *This version:*
22   URL TBD
23
24   *Errata for this version:*
25   URL TBD
26
27   *Previous version***:**
28   URL TBD
29

1 Automated Negotiation Sub Team Members

31 Martin Sachs, IBM, team lead
32 Jamie Clark, McClure-Moynihan, Inc.
33 Brian Hayes, Collaborative Domain
34 Neelakantan Kartha, Sterling Commerce
35 Kevin Liu, SAP
36 Heiko Ludwig, IBM
37 Monica Martin, Drake Certivo, Inc.
38 Dale Moberg, Cyclone Commerce
39 Himagiri Mukkamala, Sybase
40 Peter Ogden, Cyclone Commerce
41 Yukinori Saito, ECOM
42 Nikola Stojanovic, Encoda Systems
43 Jean Zheng, Vitria

Negotiation.spec.20Jan03.doc   1/24/2003 6:30 PM

44 **2 Table of Contents**

151 ## 3 Introduction

152 ### 3.1 Summary of Contents of Document

153 This document contains a specification for automatically negotiating the contents of an ebXML
154 *Collaboration Protocol Agreement* (*CPA*)[ebCPP]. This specification is a component of the suite
155 of ebXML specifications.
156
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 conditions that must be met before negotiation can begin.
166 - Section 10 discusses negotiability of elements and attributes in the *CPA*.
167 - Section 11 defines and discusses the *Negotiation Descriptor Document (NDD)* that is used to
168 describe offers and counter offers.
169 - Section 12 defines the contents of the *Negotiation Message*s.
170 - Section 13 defines the *Negotiation Protocol* including the ebXML *Business Process
171 Specification Schema*[ebBPSS] instance *Document* that is used to describe the Negotiation
172 Transactions and their choreography.
173 - Section 14 discusses negotiation algorithms.
174 - The appendices include XML Schemas for the *NDD* and *Negotiation Messages*, the
175 negotiation BPSS instance *Document*, examples of an *NDD* instance *Document* and
176 negotiation *Message* instance *Documents*, non-normative aspects of *CPA* composition, and a
177 glossary of terms.

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 15.
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.

3.4 Versioning of the Specification, Schema, and Related Documents

225 *DALE MOBERG: VERSIONING SHOULD ADHERE TO THE CONVENTIONS IN THE*
226 *CPPA SPECIFICATION AS CLOSELY AS POSSIBLE. BECAUSE MULTIPLE XSDS ARE*
227 *INVOLVED, HOWEVER, WE WILL NEED TO ALLOW EACH TO UNDERGO*
228 *VERSIONING. A MATRIX THAT MAINTAINS COMPATIBILITY INFO MAY BE*
229 *NEEDED RELATING THE BPSS FOR NEGOTIATION, NCPA VARIANTS, MESSAGING*
230 *XSD, NDD XSD.*
231

3.5 Definitions

233 Technical terms related to the subject of this specification are defined in Appendix H.
234 Technical terms related to *Collaboration Protocol Profiles* and *Agreements* and to the overall
235 vocabulary of ebXML are defined in [ebCPP].

236 **3.6 Audience**

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

241 **3.7 Assumptions**

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

244 **3.8 Related Documents**

245 Related documents include ebXML specifications on the following topics:
246 • ebXML Collaboration Protocol Profile and Agreement Specification[ebCPP]
247 • ebXML Business Process Specification Schema[ebBPSS]
248 • ebXML Message Service Specification[ebMS]
249
250 See Section 15 for the complete list of references.

251 **3.9 Acknowledgments**

252 • To Duane Nickull, XML Global, for his ebXML Automatic CPA Negotiation proposal, Feb,
253   14, 2001.
254 • To The ebXML *Business Process* Team, for its automated contract negotiation pattern in
255   [bpPATT].

256 4   Design Objectives

257 This specification defines the protocol, *Messages*, and *Business Documents* associated with
258 automatically negotiating the contents of a *CPA*. It does NOT define negotiation algorithms in
259 detail. The negotiation algorithm is part of the private process at each *Party* and MAY embody
260 private or proprietary strategies. This specification does define the rules that ensure
261 interoperability between two *Parties'* negotiation algorithms.
262
263 Following are the objectives for the design of this specification.
264
265 • The design is based on negotiating the contents of a *CPA* starting with a *CPA Template* (draft
266   *CPA*) that one prospective trading partner sends to the other as an initial offer. See Section
267   6.2 for a discussion of *CPA Template* and draft *CPA*. A *CPA Template* contains elements and
268   attributes that need to be negotiated with a prospective trading partner. A *Party* can publish a
269   *CPA Template* in a *Registry* or can create one from its *CPP* and the prospective trading
270   partner's *CPP*.
271 • The specification defines the *Negotiation Protocol* transactions and choreography by means
272   of an ebXML *Business Process Specification Schema*[ebBPSS] instance *Document*.
273 • The *Negotiation Protocol* is governed by a *Negotiation CPA* (*NCPA*). The *NCPA* is a
274   standard ebXML *CPA* that defines a minimal set of function that all *Parties* can be expected
275   to support without *Parties* having to negotiate the *NCPA* before negotiating the *CPA* for their
276   *Business Collaboration*.
277 • Avoid requiring changes to the CPPA and BPSS specifications, at least for version 1 of the
278   negotiation spec.
279 • Use deterministic algorithms
280 • The *Negotiation Process* SHOULD converge rapidly.
281   ♦ The process SHOULD either succeed or fail.
282   ♦ The process SHOULD invoke human intervention on failure
283   ♦ The design SHOULD avoid deadlock such as iterative loops that don't advance the state
284   of the negotiation. An example is reiteration over the same offer or counter offer that was
285   previously rejected by either or both parties.
286   ▪ The specification SHOULD state rules that avoid such iterative loops even if it is
287   decided that automatic detection of loops is out of scope for version 1.
288 • It MUST be absolutely clear at any point in the negotiation which *Party* (i.e., only one *Party*)
289   has the initiative to send the next request (counter offer).
290   ♦ The design SHOULD avoid race conditions in which both parties simultaneously send an
291   a counter offer. The choreography should make this an error condition.
292
293   NOTE: It is probably not possible to avoid or detect the case where two *Parties* send
294   each other initial offers. This condition should be recognized by people.
295
296 • The design SHOULD minimize the amount of state that has to be saved.
297 • Offer rejection semantics SHOULD be strong; rejection SHOULD not be a tactical
298   maneuver.

Deleted: However use of the BPSS instance *Document* is not normative and other choreographies MAY be substituted by particular groups of Parties (e.g. industry vertical organizations).

299   5 System Overview

300   The *CPA Negotiation Protocol* begins when one *Party* makes an initial offer to a second *Party*.
301   The initial offer consists of a *CPA Template* and a *Negotiation Descriptor Document (NDD)* that
302   describes what is negotiable in the *CPA Template*.
303
304   In the *CPA Negotiation Protocol*, a *CPA Template* is verified as suitable for both *Parties* and
305   modified until a suitable *CPA* is constructed. It might also be discovered that agreement cannot
306   be reached until one *Party* (or both) acquires additional software capabilities. The term "*CPA*
307   *Template*" was chosen to emphasize its use as the starting point for *CPA* negotiation. In general,
308   a *CPA Template* constitutes a proposal about an overall binding of a *Business Process* to a
309   delivery agreement with some items left open; negotiation is then used to arrive at detailed
310   values for the open items in order to achieve a final agreement. The *NDD* identifies what items
311   have to be negotiated and defines ranges or sets of acceptable values for those items.
312

313   5.1 Main Components of CPA Negotiation

314   Figure 1 illustrates the main components of *CPA* negotiation.
315
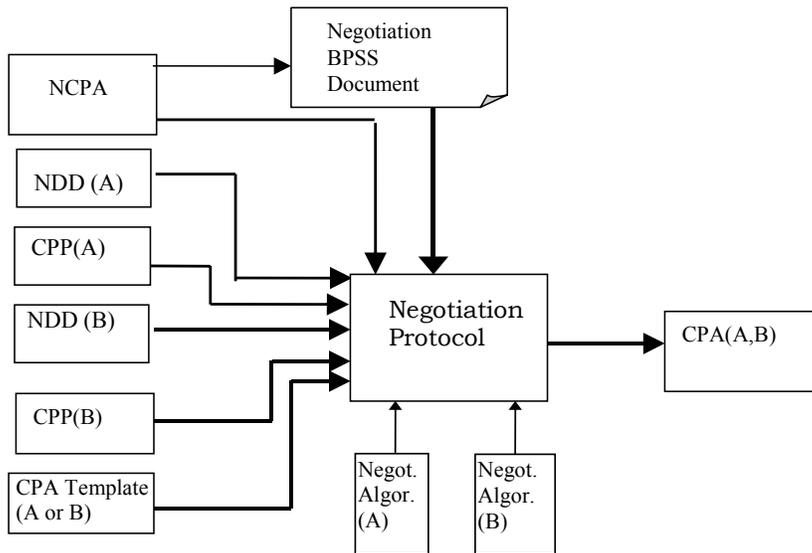
316
317

318   **Figure 1, Components of *CPA* Negotiation**

319
320   The following entities are shown in the figure:
321
322   • NCPA: The Negotiation CPA controls the Negotiation Protocol.

323   • Negotiation BPSS Instance *Document*: An ebXML *Business Process Specification*
324   *Schema*[ebBPSS] instance *Document* that is used to define the negotiation collaborative
325   protocol. This BPSS instance *Document* is referenced from an *NCPA*.
326   • *CPP*: Parties A and B publish their *CPPs* in an ebXML *Registry*[ebRS] or otherwise
327   exchange them when they discover each other.
328   • *CPA Template*: A *CPA* in which some items remained to be filled in by one or the other
329   *Party*, or negotiated between them.
330   • *NDD*: The *Negotiation Descriptor Document*, a *Document* associated with a *CPP* or a *CPA*
331   *Template* that defines what is negotiable, ranges of numeric values, etc. The *NDD* is used in
332   the *Negotiation Protocol*.
333   • *Negotiation Messages*: The *Messages* used to exchange offer and counter-offer information
334   between negotiating *Parties*.
335   • *Negotiation Protocol*: The collaborative protocol that produces a negotiated *CPA*. Although
336   shown as a single box in this figure, the *Negotiation Protocol* is executed between the two
337   *Parties* or between each *Party* and an intermediary.
338   • Negotiation algorithm: The negotiation algorithm is the private process at each negotiating
339   *Party* that implements that *Party's* private negotiation strategy. Note that the *Negotiation*
340   *Protocol* is distinct from the negotiation algorithm. The former is the public protocol,
341   captured by the BPSS instance *Document*. Each *Party* uses its negotiation algorithm, in
342   conjunction with the *CPA Template*, *CPPs* and the *NDD*, to arrive at an offer or counter offer
343   in the *Negotiation Protocol*. The negotiation algorithm is out of scope for version 1 of this
344   specification. See Section 14 for additional discussion.
345

346 Two *Parties* can negotiate a *CPA* as follows. First, they publish their *CPPs* in an ebXML
347 *Registry*, or similar *Registry*, so that potential trading partners can discover them. A *Party* MAY
348 publish an *NDD* along with the *CPP*. This *NDD* describes what is negotiable in the *CPP*.
349

350 When *Party* B discovers *Party* A as a potential trading partner, *Party* B composes a *CPA*
351 *Template* from its own *CPP* and *Party* A's *CPP*. If *Party* A published an *NDD* along with its
352 *CPP*, *Party* B MAY use the information in *Party* A's *NDD* along with its own *NDD* in
353 composing the *NDD* for the initial offer.
354

355 Alternatively, *Party* A MAY publish a *CPA Template* and *NDD*. In that case, *Party* B creates an
356 initial offer by filling in basic information about itself (e.g. its *Party* ID and transport endpoint
357 address). It then creates a new *NDD* by adding its own negotiability information to that from
358 *Party* A's *NDD*.
359

360 In order to negotiate, *Parties* A and B MUST obey the rules defined by an *NCPA*. The *NCPA*
361 MAY be virtual or real. "Virtual" means that the two *Parties* MUST configure their systems to
362 conform to the *NCPA* defined in Appendix C but there might not be a physical *NCPA* document.
363 The *Parties* can obtain each other's endpoint address by out-of-band means such as phone, fax,
364 or discovery in an ebXML or other registry.
365

366 *DALE MOBERG: WE SHOULD LEAVE ROOM FOR BOTH HTTP POST AND SOAP*
367 *1.1/1.2 BINDINGS FOR THE MESSAGES EXCHANGED. WE SHOULD LEAVE ROOM*
368 *FOR WSDL DESCRIPTIONS OF THE MESSAGE EXCHANGES CONSTRUED AS*

Deleted: MAY be

Deleted: have to establish an *NCPA* between themselves. The following procedure can be used.¶

Negotiation.spec.20Jan03.doc   1/24/2003 6:30 PM

369 | ***SERVICES ALSO. IN FACT, I WOULD FAVOR HAVING A NON-NORMATIVE***
370 | ***APPENDIX WITH A WSDL FOR NEGOTIATION EVENTUALLY.***

372 | "Real" means that the *Parties* create and deploy a specific *NCPA* using, for example, the
373 | following procedure:

375 | 1. *Parties* A and B publish *NCPA Templates* (that they are willing to abide by) in a *Registry*.
376 | They are *NCPA Templates* (as opposed to *NCPAs*) because some information (such as the
377 | prospective trading partner's *Party* ID and endpoint address) is missing from an *NCPA*
378 | template. In many cases, a *Party's NCPA Templates* might differ from each other only with
379 | regard to which of several negotiation BPSS instance *Documents* they refer to.
380 | 2. *Party* B discovers *Party* A and wants to conduct trade.
381 | 3. *Party* B chooses an *NCPA* template of *Party* A that it can live with (say, by looking at the
382 | BPSS instance *Document* pointed to by this *NCPA* template).
383 | 4. *Party* B then fills in this *NCPA* template with its own name, endpoint address, etc. (so that
384 | now it becomes an *NCPA*) and sends it, along with a draft *CPA* and an initial *NDD* to *Party*
385 | A to start the *Negotiation Protocol*.

387 | ***DOES STEP 4 ABOVE WORK? IS IT NECESSARY FOR PARTY A TO RECEIVE AND***
388 | ***DEPLOY THE NCPA BEFORE PARTY B CAN SEND AN INITIAL OFFER?***

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

393 | **5.2 Overview of CPA Negotiation**

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

Deleted: These *NCPA Templates* are to be distinguished from regular *CPAs* by *Registry* metadata.

Deleted: of the

Negotiation.spec.20Jan03.doc   1/24/2003 6:30 PM
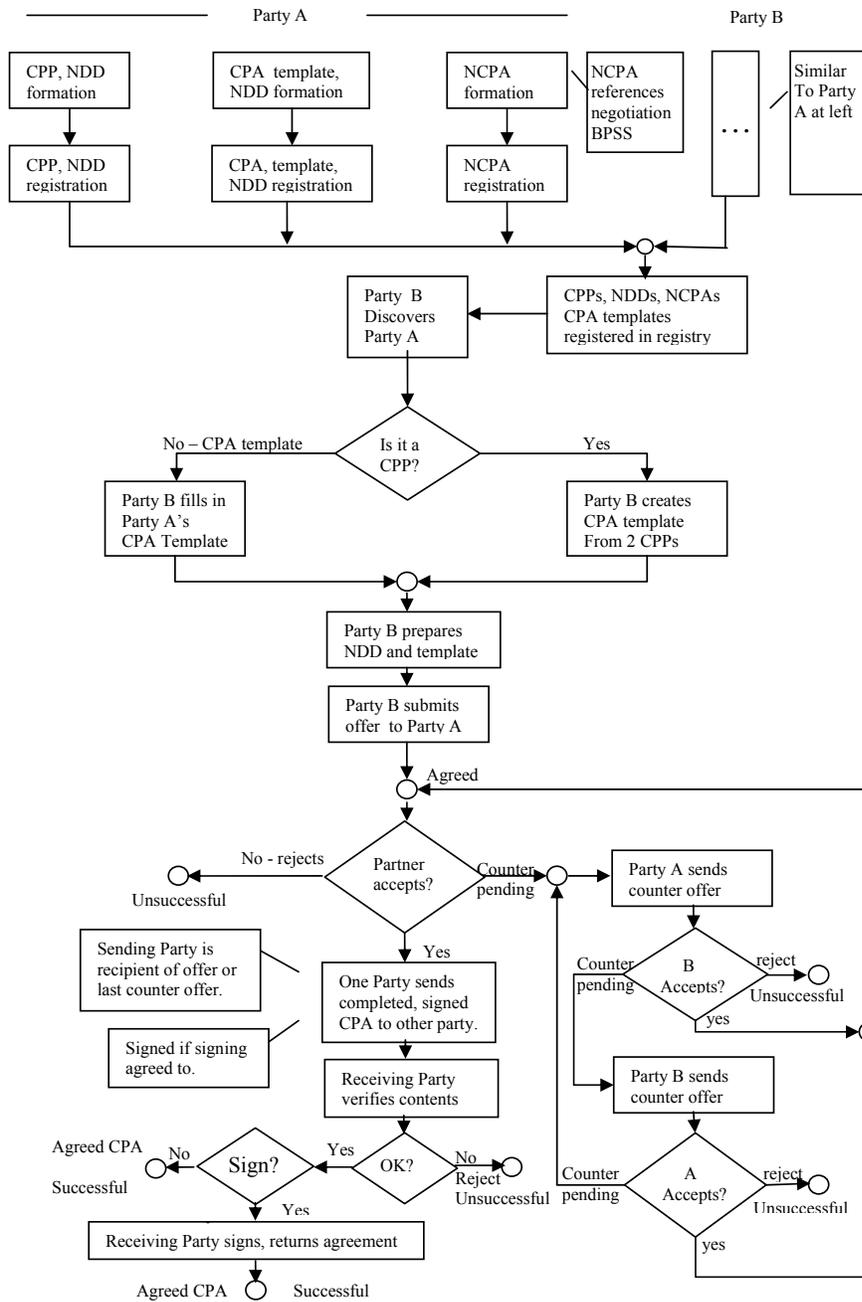
Figure 2, Negotiation Process

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

5.3 CPP and NDD Formation and Editing

435 These are pre-discovery steps that are out of scope for the negotiation specification, they are
436 included here in the interest of completeness. Following is a non-normative list of the elements
437 of *CPP* and *NDD* formation.
438 • *CPP* template (a prototype CPP that could be used for creating CPPs).
439 ♦ Supplied with software installation (configured options)
440 ♦ Edited to reflect preferences
441 • *NDD* formation.
442 ♦ Although *NDD* formation is out of scope, the *NDD* schema is a key component of this

Automated Negotiation Specification   **Page 13 of 97**

443   specification.
444 • Tool for custom *CPP* formation
445 • Tool for *CPA* and *CPA Template* formation.
446 • Tool for *NDD* formation
447 • Service(s) for supplying *CPPs* or *CPA Templates*
448   ♦ UDDI advertised, SOAP, ebXML, simple HTTP GET, and so on.
449 • ebXML *Registry* submission (publication)
450
451 In principle, a *Party* SHOULD be able to publish both a *CPP* and a *CPA Template*. However,
452 this would lead to a problem that a given prospective trading partner might find either one. If a
453 *Party* intends that some prospective trading partners negotiate with a *CPP* while other are
454 expected to accept a *CPA Template*, then the *Party* SHOULD probably publish only the *CPP* and
455 decide whether to send a *CPA Template* based on its knowledge of who the prospective trading
456 partner is.

457 **5.4 Discovery of CPPs and CPA Templates**

458 The discovery process is out of scope for the negotiation specification; it is included here in the
459 interest of completeness. Following are some points concerning the discovery process.
460 • The minimum requirement is to be able to perform an HTTP GET of a CPP from a URL
461   obtained by means outside the scope of this specification.
462 • UDDI ebXML *Registry* bootstrap. This permits CPPs to be advertised in either UDDI or the
463   ebXML Registry.
464 • Search and retrieval in ebXML *Registry* or similar *Registry*.
465 • Well-known address of the *Registry*.
466 • Can a *Registry* provide further function, perhaps as value-added services?
467   ♦ Notification of *CPP* expirations?
468   ♦ Accept filled-out *CPA Templates*?

469 **5.5 Negotiation through an Intermediary**

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

473 # 6 CPA Template

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

475 ## 6.1 CPA Template and Draft CPA

476 The *Negotiation Protocol* defined in this specification is based on the use of a *CPA* that is
477 incomplete in that items that are negotiable or MUST be filled in by the *Party* that receives an
478 initial offer. Negotiable items can have "dummy" values that will later be replaced by the agreed
479 values arrived at during the *Negotiation Process*. Such an incomplete *CPA* can be categorized as
480 a *CPA Template* or a draft *CPA*.
481
482 A *CPA Template* will normally contain dummy values for the *Party*-specific values of the *Party*
483 to which the *CPA Template* is being sent as well as dummy values for other items that the
484 offering *Party* considers negotiable. A draft *CPA*, on the other hand, will typically have been
485 formed by pruning and combining *CPPs* of each of the *Parties* in the *Negotiation Process*, and
486 so can contain all "real" values. That is, using a *CPA Template* will typically require a counter
487 proposal while using a draft *CPA,* the *Party* making the initial offer might only be asking for
488 approval of the draft *CPA* rather than offering to negotiate some items. For convenience, both
489 kinds of *Document*, though having different origins, will be referred to by the term "*CPA*
490 *Template*" because the process of negotiation proceeds the same way for either *CPA Templates*
491 or draft *CPAs*. Sensible use of *CPA Templates* requires that the dummy values be indicated as
492 negotiable and that acceptance does not occur until the dummy values have been replaced. In this
493 specification, the *NDD* is the means of indicationg what is negotiable.
494
495 A *CPA Template* can encompass a wide range of negotiating possibilities. At one end of the
496 range, it might amount simply to a take-it-or-leave-it offer, its *NDD* indicating only those items
497 that MUST be filled in to customize it to the other *Party*. At the other end of the spectrum, its
498 *NDD* might indicate that virtually everything is negotiable.
499
500 In the simplest case, the accompanying *NDD* might be very simple and would simply indicate
501 which elements and attributes need to be completed by the prospective trading partner, such as
502 *Party* ID and transport endpoint address. For this case, the *NDD* facilitates identifying the items
503 to be filled in, avoiding the need to label the items to be filled in within the *CPA Template* and
504 the need to parse the *CPA Template* to find those items.

505 ## 6.2 Advantages of Starting Negotiation with a CPA Template

506 If negotiation is performed with the two *Parties' CPPs* and an *NDD* for each, everything in the
507 *CPPs* is potentially negotiable and has to be considered during the *Negotiation Process*. The
508 process of composing a *CPA Template* from two *CPPs* will often narrow down the amount of
509 negotiation relative to the negotiation possibilities expressed in the *NDDs* that accompany the
510 *CPPs*. The reason is that many of the differences between the two *CPPs* can be "mechanically"
511 resolved by finding compatible choices and matching values of some elements or attributes. For
512 example, there might be only one transport protocol that is common to the two *Parties*. After the
513 *CPA Template* is constructed, a new *NDD* MUST be constructed that includes only the items in
514 the *CPA Template* that remain to be negotiated.
515

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

524   **6.3 CPA Template composition**

525   Composition of a *CPA Template* is the same as composing any *CPA* from two *CPPs*. Appendix
526   E, "*CPA* Composition (Non-Normative)", of [ebCPP] contains a detailed discussion of *CPA*
527   composition from two *CPPs*.

Negotiation.spec.20Jan03.doc   1/24/2003 6:30 PM

7   CPP and CPA Template Content

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

7.1 Validation of CPP and CPA Template

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

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

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

THE ABOVE MATERIAL WILL BE EXTENDED TO ENCOMPASS ALL NEGOTIABILITY PATTERNS THAT ARE IDENTIFIED.

7.2 Preference Order

Enumerations MUST always be stated in preference order (highest preference first). In most cases, preference order is REQUIRED by the CPPA specification[ebCPP]. Following are examples:
- ***PartyId*** elements under the same ***PartyInfo*** element.
- CanSend and CanReceive elements under the ServiceBinding element (NEED TO VERIFY THIS)
- AccessAuthentication elements under the same TransportSender element
- EncryptionAlgorithm elements under the same TransportClientSecurity or TransportServerSecurity element.
- ***TransportProtocol*** elements under the same ***Transport*** element
- ***AnchorCertificate*** elements under the same ***Certificate*** element

7.3 Conflicts between two Parties' Preferences

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

7.4 CPA Period of Validity

The values of the ***Start*** and ***End*** elements in the *CPA Template* SHOULD be consistent with each other (start time MUST precede end time) and SHOULD be consistent with the expiration times of all the certificates. It is preferable that the *CPA* expire before any of its certificates

Automated Negotiation Specification   **Page 17 of 97**

568   expire. All of these times are negotiable but it will simplify matters if the times in the *CPA*
569   *Template* are mutually consistent. If the **Start** and **End** elements do not appear in the *CPPs*; they
570   MUST be added when the C*PA Template* is composed from the *CPPs*.

Negotiation.spec.20Jan03.doc   1/24/2003 6:30 PM

8 Negotiation CPA (NCPA)

The purpose of this section is to:
- Explain how to construct the *Negotiation CPA* such that it does not have to be negotiated;
- Explain the negotiation aspects of the *NCPA*. Principally, these aspects are the elements that define the interface between a *CPA* and the BPSS instance *Document*, i.e., the **CollaborationRole**, **ProcessSpecification**, and **Role** elements.

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

IT IS ESSENTIAL THAT EVERYONE REVIEW THE AND DISCUSS THE NCPA INSTANCE DOCUMENT.

DALE MOBERG: SOME SIMPLE PARTS MAY BE MISSING FROM THE NCPA.

DALE MOBERG: I DO NOT THINK THAT WE HAVE DOCUMENTED BOTH ASYNC AND SYNC PATTERNS FOR NEGOTIATION.

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

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

THIS MATERIAL WILL BE EXPANDED AS NEEDED.

8.1 Document Exchange

The following rules eliminate the need for negotiating the *Document*-exchange specifications for the *NCPA*:
- Omit the following child elements of the ***ebXMLSenderBinding*** and ***ebXMLReceiverBinding*** elements: ***ReliableMessaging***, ***PersistDuration***, ***xxxNonRepudiation***, and ***xxxDigitalEnvelope***. This means that reliable *Messaging* and *Message* security are not used.
  THIS SPECIFICATION NEEDS TO STATE WHETHER OR NOT THE NAMESPACESUPPORTED ELEMENT IS REQUIRED OR MUST BE OMITTED. THE NAMESPACESUPPORTED ELEMENT CAN ALSO BE OMITTED UNLESS THE MESSAGE STRUCTURE USED FOR NEGOTIATION REQUIRES IDENTIFYING NAMESPACES FOR BODY PARTS.
- In the ***MessagingCharacteristics*** elements, specify the value "never" for the attributes

Negotiation.spec.20Jan03.doc   1/24/2003 6:30 PM

614   *ackRequested*, *ackSignatureRequested*, and *duplicateElimination* (they are used only with
615   reliable Messaging). For the *actor* attribute, specify either of the permitted values; this
616   attribute is ignored when the value of the *ackRequested* attribute is "never".
617   *THE VALUE OF THE SYNCREPLYMODE ATTRIBUTE SHOULD BE SPECIFIED IN*
618   *THIS NEGOTIATION SPECIFICATION. IT SHOULD NOT HAVE TO BE*
619   *NEGOTIATED.*

620

621   *DALE MOBERG: SYNCREPLY MODE IN NCPA IS VERY [EBMS] SPECIFIC. WE HAVE*
622   *TO BE CAREFUL TO ALLOW VARIOUS MESSAGING AND TRANSFER PROTOCOL*
623   *BINDINGS AND INITIALLY PICK ONE AS A DEFAULT AND OTHERS AS OPTIONAL.*
624   *HTTP POST DEFAULT? SOAP 1.1 DEFAULT?*

625

626   *THE FOLLOWING IS AN ALTERNATIVE THAT WOULD REQUIRE DEFINING A NEW*
627   *BINDING IN THE CPPA SPECIFICATION.*
628   *Messaging* could be specified to use basic SOAP or W3C XML Protocol (when available). In
629   this context, "basic" means that values or choices that normally have to be negotiated will either
630   be omitted or will be given fixed values by this specification.

631   **8.2 Transport**

632   •   Use HTTP PUT or POST to send a proposed *CPA* to a URL.
633   •   The response to an offer or counter offer is always synchronous. This avoids the need for the
634   responder to know the URL for a response.

635

636   See [ebMS]. Appendices B.2.4 and B.2.5 regarding synchronous HTTP connections.

637

638   **8.3 Packaging**

639   *COMPLETION OF THE PACKAGING DEFINITION (E.G. SIMPLEPART*
640   *DEFINITIONS) AWAITS COMPLETION OF THE NDD AND NEGOTIATION MESSAGE*
641   *SCHEMAS.*

642   **8.4 Security of Negotiation Protocol**

643   *THE FOLLOWING ARE PRIMARILY BOOTSTRAP ISSUES. MORE DISCUSSION AND*
644   *DECISIONS ARE NEEDED.*

645

646   *WE NEED TO DECIDE WHICH OF THESE ITEMS ARE IN SCOPE FOR VERSION 1*
647   *AND WHICH MIGHT BE FUTURE WORK.*

648

649   *HAVE WE DEFINED WHAT IS NEEDED TO NEGOTIATE THE SECURITYDETAILS*
650   *ELEMENT?*

651

652   *MONICA MARTIN: THIS SECTION IS A BIT NEBULOUS IN THAT IT REFERENCES*
653   *TRUST AND SECURITY ISSUES BUT DOESN'T SAY THAT THIS PROCESS WILL OR*
654   *WON'T BE USED. IF IT IS OUTSIDE THE SCOPE OF V1.0, STATE ACCORDINGLY.*

655

656   •   If both Parties have the same trust model, negotiation can proceed in a secure fashion.

657   ♦ An initial negotiation of trust anchors and other security matters might be needed.
658   Consider exchanging this information dynamically, using *Message* exchanges.. The
659   might be slower, but simpler, than putting it in the *NCPA*. This might involve human
660   intervention to evaluate and accept the proposed trust model and then to configure the
661   systems to use it for negotiation.
662 • One *Party* might have to add a new trust anchor proposed by the other *Party*.
663 • The signing certificate need not be the same as the others.
664 • Certificate validity.
665 • Are self-signed certificates permitted?
666 • For the initial version of the specification, omit *Document*-exchange certificates.
667 • Signing of *Negotiation Messages* has to be covered either in the *NCPA* or in the initial
668   security negotiation mentioned above.

8.5 Explanation of NCPA Example

670 The text of the *NCPA* example is in Appendix C.

671

672 ***TO BE SUPPLIED.***

Negotiation.spec.20Jan03.doc   1/24/2003 6:30 PM

673 ## 9 Pre-Conditions for Negotiation

674 This section discusses conditions that MUST be met before negotiation. If these conditions are
675 not met, a successful outcome is unlikely. The discussions relate to *CPPs* or a *CPA Template* as
676 appropriate
677
678 The two partners MUST agree on what *Negotiation Protocol* to follow, i.e. what *NCPA* to use
679 for negotiation. (The *NCPA* identifies the *Negotiation BPSS Instance Document* to be used.)
680
681 There MUST be a minimum level of matching (i.e. compatibility) between two *CPPs*.
682 • There MUST be at least one transport protocol in common.
683 • There MUST be a minimum level of compatibility between at least one **DocumentExchange**
684   element in each *CPP **(DETAILS TO BE DETERMINED).***
685 ***THIS LIST WILL BE EXPANDED.***
686
687 See Section 7 for related information.
688
689 *MONICA MARTIN: HOW WILL WE ACCOUNT FOR SELF SIGNNG FOR THE*
690 *CERTIFICATE AUTHORITY REFERENCE RELATED TO DELIVERY CHANNELS?*

Negotiation.spec.20Jan03.doc   1/24/2003 6:30 PM

691 # 10 Negotiability of CPA Elements and Attributes

692 *THIS SECTION IS BASED ON THE WORK BEING DONE WITH THE CPA ELEMENT*
693 *AND ATTRIBUTE NEGOTIABILITY SPREADSHEET.*
694
695 This section discusses the negotiability of the different elements and attributes in the *CPA* and is
696 concerned mostly with composing a *CPA* from two *CPPs*. It focuses on those cases that involve
697 special considerations.

698 ## 10.1 Enumerations

699 There are several cases of enumerations:
700 • Some enumerations are laid out in the *CPP* instance *Documents* (e.g. certificates).
701 • Some enumerations are laid out in the CPPA schema itself.
702 • Some enumerations might be defined only in the text of the CPPA specification and would
703   have to be put into the *NDD* schema.
704 • Some enumerations are not listed in full anywhere (e.g. the W3C forms of encryption
705   algorithm name)
706 • Some might be defined elsewhere, perhaps as a set of URIs.
707
708 In some cases, especially those that are defined in the CPPA schema, only the items in an
709 enumeration that are acceptable to the *Party* that is preparing the *NDD* instance *Document* have
710 to be listed in the *NDD*. An example is the versions of the specification that are acceptable to the
711 *Party*.
712
713 The CPPA schema itself is input to the *Negotiation Process*. Therefore, enumerations that are
714 defined in full in the CPPA schema don't necessarily have to be defined in full in the *NDD*
715 schema.

716 ## 10.2 CollaborationRole element and its child elements

717 The normal case is that the two *CPPs* are being composed into a *CPA Template* specify the same
718 BPSS instance *Document*. In version 1 of this specification, the contents of the BPSS instance
719 *Document* cannot be negotiated using the negotiation functions defined in this specification.
720 Two prospective trading partners SHOULD agree on the same BPSS instance *Document* and
721 assignment of *Roles* before beginning to negotiate the *CPA*. In many cases, agreement will be
722 established by the fact that the two prospective trading partners have compatible
723 **CollaborationRole** subtrees in their *CPPs*. The following considerations relate to establishing
724 compatible **CollaborationRole** subtrees.
725 • If both *CPPs* specify the same *Role* (e.g. both specify "buyer"), the situation cannot be
726   resolved automatically. Human contact is needed and one *CPP* MUST be changed to specify
727   the other *Role*.
728 • If both *CPPs* specify both *Roles* (i.e. two **CollaborationRole** elements with opposite *Roles*),
729   this cannot be resolved automatically. Human contact is needed and the two *Parties* MUST
730   agree on which *Party* plays which *Role*.
731 • If *CPP* A specifies one *Role* and *CPP* B specifies both *Roles*, chose the *Role* in *CPP* B which
732   is opposite to the *Role* specified in *CPP* A.

Automated Negotiation Specification   **Page 23 of 97**

733 • If both *CPPs* specify more than one BPSS instance *Document* but there is only one in
734   common to the two *Parties*, use that one.
735 • If both *CPPs* specify more than one BPSS instance *Document* that is in common to both of
736   them, human contact is needed to decide whether all the common ones are to be used in the
737   *Business Collaboration* or which one is to be used.
738
739 From the viewpoint of *CPA* composition and negotiation, the best practice is to include only one
740 BPSS instance *Document* in each *CPP*.
741
742   NOTE: A *Party* can describe the *Business Collaboration* using any desired alternative to
743   the ebXML *Business Process Specification Schema*. When an alternative *Business-
744   Collaboration* description is used, the *Parties* to a *CPA* MUST agree on how to interpret
745   the *Business-Collaboration* description and how to interpret the elements in the *CPA* that
746   reference information in the *Business-Collaboration* description. The affected elements
747   in the *CPA* are the **Role** element, the **CanSend** and **CanReceive** elements, the
748   **ActionContext** element, and some attributes of the **BusinessTransactionCharacteristics**
749   element. The two *Parties* also have to come to a common understanding of how to
750   negotiate the negotiable elements and attributes whose interpretations are changed by the
751   use of the alternative *Business Collaboration* description.
752

753 **10.3 Elements or Attributes whose Cardinality Includes Zero**

754 Regarding elements or attributes whose cardinalities include zero (omission), the main
755 negotiable thing is "presence or absence". However, if it is agreed to include (one or more of)
756 that element or attribute, it is then necessary to negotiate the value (or child elements in the case
757 of an element) of each one that is included. **PersistDuration** is an example. If the two parties
758 agree to include it, they then have to negotiate its value.

759 **10.4 Values**

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

763 **10.5 Items that are Referred to**

764 *NEGOTIATION OF ITEMS THAT ARE REFERRED TO (E.G. BY IDREF) IS AN OPEN*
765 *QUESTION.*

766 **10.6 Transport Endpoints**

767 Transport endpoints are not really negotiable since any *Party* can define whatever endpoints it
768 chooses. There might be issues of matching endpoint characteristics. One example is the
769 endpoint type. *Parties* might need to negotiate what endpoint types are used.
770
771 *IT WAS NOT CLEAR TO THE SUBTEAM HOW MUCH USE WILL BE MADE OF*
772 *ENDPOINT TYPES OTHER THAN "ALL PURPOSE". FOR ITEMS WHOSE WIDE USE*
773 *IS NOT CERTAIN, IT MAY BE BETTER NOT TO DESIGN IN DETAIL IN THE FIRST*
774 *VERSION. INSTEAD, WE COULD INCLUDE A NON-NORMATIVE NOTE ON*
775 *WHATEVER WE UNDERSTAND ABOUT EACH SUCH ITEM AND LEAVE IT FOR*

Automated Negotiation Specification   **Page 24 of 97**

776 *FUTURE VERSIONS TO CONSIDER THE NEED TO NEGOTIATE IT.*

777 **10.7 Security**

778 *THESE POINTS NEED FURTHER DISCUSSION AND DECISIONS.*

779 • Negotiation on certificates might require human contact.
780 • A *Party's* unwillingness to handle the proposed trust model is a reason for failure of the
781   negotiation.

782 **10.7.1 Trust Anchors and Related Matters**

783 This section discusses the kinds of negotiation that might take place for aligning ***SecurityDetails***
784 and ***TrustAnchors*** with various ***CertificateRefs***.
785
786 *DOES THIS SPECIFICATION DEFINE WHAT IS NEEDED FOR NEGOTIATING*
787 *TRUST ANCHORS?*
788
789 There are 3 major levels for alignments in public-key infrastructure (PKI)*. ALIGNMENTS OF*
790 *OTHER SECURITY CREDENTIALS ALSO NEED TO BE DISCUSSED HERE.*
791
792 1. Transport-level security
793 2. *Messaging*-level security
794 3. Application-level security
795
796 For transport-level security, (transient) encryption and authentication alignment are needed. Both
797 server-side and client-side SSL or TLS need to have the trust anchors synchronized with
798 corresponding certificates.
799
800 For *Messaging*-level (persistent) security, digital envelopes and non-repudiation (of origin and/or
801 receipt) by means of digital signatures require alignment.
802
803 For application-level (persistent) security, digital envelopes and non-repudiation (of origin
804 and/or receipt) by means of digital signatures require alignment.
805
806 Failure to validate a certificate need not prevent formation of a *CPA Template*. First, the sender's
807 signing certificate can be a self-signed certificate. If so, a reference to this self-signed certificate
808 can be added to the receiver's ***TrustAnchors*** and ***AnchorCertificateRef*** lists. This proposal
809 amounts to proposing to agree to a direct trust model, rather than a hierarchical model involving
810 certificate authorities. Second, a proposal to add a trusted root might be made, again by
811 appropriate revision of the ***TrustAnchors*** element.
812
813 As a result of the *CPA Template* formation process, various details could be up for negotiation.
814 *OTHER DETAILS ABOUT ALGORITHMS OR STRENGTHS NEED TO BE ADDED.*
815
816 First, a change to the PKI might be proposed. For the self-signed certificate addition option, the
817 negotiatee might want to:
818
819 1. Reject adding a self-signed certificate and indicate rejection of the security function resting

Automated Negotiation Specification   **Page 25 of 97**

820   on this PKI alignment
821 2. Insist on the proposer getting a certificate from an existing CA.
822 3. Propose issuing another certificate signed by an acceptable authority.
823
824 For case 1, the negotiation "space" would involve a change in the value of an attribute under
825 ***BusinessTransactionCharacteristics***.
826
827 For case 2, the negotiatee would have to indicate rejection of the *CPA Template* and indicate that
828 until the *CPP* certificate value changes, there will be no forward progress. The proposer would
829 have to go out and get a new certificate.
830
831 For case 3, the negotiatee would propose a different certificate issued by its own CA. The
832 negotiatee would have to install it and use it for this transaction. This is not yet a common
833 practice, though it is logically possible. This would involve one side being a CA for the *Business*
834 *Process* and the ability of the other side to use more than one certificate for its existing key-pair.
835 The *CPA* proposed to do this would go outside of anything strictly derivable from the *CPP* (only
836 the old X.509 certificate would be used to put together a new X.509 certificate from a new
837 issuer).
838
839 Next, for the PKI trust anchor certificate addition option, the negotiatee might want to:
840
841 1. Reject adding a new CA to its trust anchors and indicate rejection of the security function
842   resting on this PKI alignment.
843 2. Insist on the proposer getting a certificate from some already trusted existing CA.
844 3. Propose accepting another certificate signed by its own signing authority.
845 4. Propose a different trust anchor either higher or lower in the validation chain than the one
846   proposed by the other side.
847
848 Again, as for adding a self-signed certificate, for case 1, the negotiation "space" would involve a
849 change in the value of an attribute under the ***BusinessTransactionCharacteristics*** element. For
850 case 2, the response would have to be rejection with a call for a change in *CPP*. For case 3, the
851 negotiatee proceeds as described in case 3 above.
852
853 The new case 4 is logically possible but still exotic. In effect, the negotiation SHOULD not
854 matter to the other side, because it is just an adjustment to which trust anchor is added to one
855 side's PKI trust list and the certificate used would still validate to the alternative trust anchor. Yet
856 it would reflect a slight change in security details.

857 **10.8 Discussion of Various Elements and Attributes**

858 *RULES FOR ADDITIONAL ELEMENTS AND ATTRIBUTES PROBABLY HAVE TO BE*
859 *ADDED.*
860
861 ***cpaid:*** The value of the ***cpaid*** attribute can be negotiated. In order to negotiate the value of the
862 ***cpaid*** attribute, it SHALL be a URI.
863
864 ***Start*** and ***End*** elements: The value of the ***Start*** element MUST precede the value of the ***End***

Deleted: Id
Deleted: aId
Deleted: Id

865   element and the times stated in the **Start** and **End** elements MUST NOT be outside the certificate
866   validity periods. If the values of the **Start** and **End** elements are negotiable, the *CPP* SHALL
867   specify the earliest acceptable start time and the latest acceptable end time.
868   ***IS THE ABOVE DEFINITION CORRECT AND ACCEPTABLE?***
869
870   **Status** element: The **Status** element is not negotiable; its value identifies the state of the
871   negotiation. The negotiation algorithm is responsible for changing the state at appropriate times.
872
873   **defaultMSHChannelId**: Since a delivery channel contains both *Parties'* properties, the two
874   *Parties* have to agree on both *Parties'* default delivery channels. ***MORE DISCUSSION IS***
875   ***NEEDED ON THIS SUBJECT.***
876
877   **defaultMSHPackageId**: *A USE CASE IS NEEDED*.
878
879   **PartyId** type: The **type** attribute of the **PartyId** element identifies the naming system to which
880   the **PartyId** belongs (e.g. DUNS). The *Negotiation Process* SHOULD select one possible
881   **PartyId** type for each *Party* and eliminate any others that are in the *CPPs*. Each *Party's* **PartyId**
882   type MUST be understandable by the other *Party*. Eliminating the others ensures that each *Party*
883   will always use the same **PartyId** for the other *Party*.
884
885   **PartyRef: THE TYPE ATTRIBUTE OF PARTYREF NEEDS A USE CASE FOR**
886   **NEGOTIABILITY.** One possible reason to negotiate is that a *Party* might not be able to
887   understand the other *Party's* **PartyRef** *Document*. For example, the geographical contexts might
888   not match. While negotiating the contents of the **PartyRef** *Document* is out of scope for this
889   specification, negotiating the contents might lead to negotiating the schema (type), which is in
890   scope.
891
892   **CollaborationRole**: the cardinality is one or more.
893
894   **version** attribute of the **ProcessSpecification** element: The two *Parties' CPPs* might specify the
895   same BPSS instance *Document* but different versions of it.
896
897   ***THE VERSION ATTRIBUTE OF THE BPSS PROCESSSPECIFICATION ELEMENT IS***
898   ***ACTUALLY THE VERSION OF THE SPECIFICATION. THE BPSS SPECIFICATION***
899   ***DOES NOT DEFINE A VERSION ATTRIBUTE OF A BPSS INSTANCE DOCUMENT.***
900   ***DECISIONS HAVE TO BE MADE ON HOW TO RESOLVE THE DISCREPANCY***
901   ***BETWEEN [EBCPP] AND [EBBPSS] AND THEN TO REWRITE REFERENCES TO THE***
902   ***VERSION ATTRIBUTE IN THIS SPECIFICATION. ONE SOLUTION IS TO CORRECT***
903   ***[EBCPP] TO REFER TO THE VERSION OF [EBBPSS] AND CHANGE THIS***
904   ***SPECIFICATION ACCORDINGLY. ANOTHER IS TO DEFINE A BPSS INSTANCE***
905   ***DOCUMENT VERSION AND CORRECT BOTH [EBCPP] AND THIS SPECIFICATION***
906   ***TO AGREE WITH THE BPSS SPECIFICATION. THE LATER SOLUTION IS FOR A***
907   ***FUTURE VERSION OF ALL THREE SPECIFICATIONS.***
908
909   **name** attribute of the **ProcessSpecification** element: This is not negotiable unless a future
910   version of [ebBPSS] provides for more than one **ProcessSpecification** element in a BPSS

911   instance *Document*.

912

913   ***THE SUBTEAM HAS RECENTLY DISCUSSED THE POSSIBILITIES IN NEGOTIATING***
914   ***ABOUT WHICH BPSS INSTANCE DOCUMENT TO USE. FOLLOWING ARE THE***
915   ***POSSIBLITIES DISCUSSED:***
916   ***- ANY NEGOTIATION ABOUT WHICH BPSS INSTANCE DOCUMENT TO USE IS FOR***
917   ***THE FUTURE.***
918   ***- PERMIT NEGOTIATING ON THE NAME ATTRIBUTE OF THE BPSS INSTANCE***
919   ***DOCUMENT, I.E. ON WHICH BPSS INSTANCE IS TO BE USED.***
920   ***- PERMIT NEGOTIATING ON THE VERSION OF THE BPSS SPECIFICATION TO BE***
921   ***USED. (SEE 'VERSION ATTRIBUTE' ABOVE.***
922   ***- REQUIRE THAT THE CHOICE OF BPSS INSTANCE DOCUMENT SHOULD BE***
923   ***LIMITED TO CHOICES THAT DO NOT REQUIRE CHANGES IN THE ELEMENTS AND***
924   ***ATTRIBUTES OF THE COLLABORATIONROLE ELEMENT.***

925

926   ***DURING THE DISCUSSION, IT WAS POINTED OUT THAT EVEN SUBSTITUTING A***
927   ***VERY SIMILAR BPSS INSTANCE DOCUMENT FOR ANOTHER COULD AFFECT***
928   ***WHAT IS DEFINED IN THE ACTIONCONTEXT ELEMENT. ONE POSSIBILITY IS TO***
929   ***STATE THAT THE ALTERNATIVE BPSS INSTANCE DOCUMENTS MUST BE***
930   ***STRUCTURALLY SIMILAR AND TO WARN THAT EVEN SO, CHANGES MIGHT BE***
931   ***NEEDED IN THE CONTENTS OF THE ACTIONCONTEXT ELEMENT; THE PARTIES***
932   ***WILL HAVE TO BE PREPARED TO NEGOTIATE ABOUT THE ACTIONCONTEXT***
933   ***ELEMENT***

934

935   ***DALE MOBERG; LOOK INTO WHAT DEPENDENCIES NEGOTIATION OF OTHER*** | **Deleted:** . |
936   ***ITEMS IN THE CPA HAS ON NEGOTIATING THE BPSS INSTANCE DOCUMENT***
937   ***NAME, ESPECIALLY ACTIONCONTEXT. ARE THERE THINGS THAT MIGHT HAVE***
938   ***TO BE NEGOTIATED THAT ARE NOT COVERED BY THE CURRENT DRAFT?***

939

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

945

946   ***Role***: The two *Parties* have to have opposite *Roles* in a *Business Collaboration*. This MUST be
947   validated. ***THERE IS NO KNOWN USE CASE FOR NEGOTIATING IT.***

948

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

952

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

955

956   ***PackageId*** might be negotiable.

957
958 *ActionContext*: This is not negotiable. If BPSS is not being used, ignore the *ActionContext*
959 element.
960
961 *CollaborationActivity*: This allows a *Party* to specify a complete path inside the BPSS instance
962 *Document*. Its value is completely determined by the structure of the BPSS instance *Document*
963 and is therefore not negotiable.
964
965 *channelId*: The *Parties* can negotiate which delivery channels to use or add new ones.
966
967 *Certificate*: An enumeration of *keyinfo* types might be useful to help decide which certificates
968 are acceptable.
969
970 *DeliveryChannel*: Cardinality is negotiable. It is suggested that a new delivery channel be
971 created rather than modifying an existing one.
972
973 *Signing the CPA*: Negotiation of signing is accomplished by negotiating presence of the *CPA*
974 *Signature* element and its child *ds: Signature* elements. See Section 13.13 for details.
975
976 *Comment*: [ebCPP] states that all comments in both *CPPs* SHALL be included in the *CPA*
977 unless the *Parties* agree otherwise. Therefore, each *Comment* element is separately negotiable.
978 Since comments are arbitrary text strings, negotiation about *Comment* elements MUST be by
979 human to human contact.

980 # 11 Negotiation Descriptor Document

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

984

985 The *NDD* identifies the *CPP* or *CPA Template*. The *CPP* or *CPA Template* does not identify the
986 *NDD* since a *Party* might have many different *NDDs* associated with the same *CPP* or *CPA*
987 *Template*. These could be for different *Negotiation Processes*, different categories of partner,
988 etc.

989

990 ## 11.1 Use of NDD

991 • An *NDD* can be placed in a *Registry* along with the *CPP*. The *NDD* and *CPP* would have to
992 be connected by *Registry* metadata. Alternatively, a *Party* might choose not to include an
993 *NDD* in the *Registry*. Instead, when a *Party* is discovered by a prospective trading partner,
994 the *NDDs* can be exchanged prior to the opening step of the negotiation. This permits a *Party*
995 to send an *NDD* that it considers appropriate for the particular prospective trading partner.
996 • An *NDD* is sent from the *Party* making the initial offer to the other *Party* during initialization
997 of the *Negotiation Protocol*. After that, the *NDD* is not modified during negotiation and is
998 not again sent from one *Party* to the other. All information about the state of negotiation of
999 the negotiable items is exchanged in the *Negotiation Messages*.

1000

1001 NOTE: This means that an item which is initially not negotiable cannot be made negotiable
1002 during the *Negotiation Protocol*.

1003 ## 11.2 General Principles of Contents of NDD

1004 The *NDD* has been defined in an abstract manner to enable it to be applied to any kind of XML
1005 agreement. This avoids the need to define a new *NDD* schema for each kind of *Document* to be
1006 negotiated.

1007

1008 NOTE: The abstract level of the *NDD* is an opportunity for tool vendors to produce *NDD*
1009 composition tools. Such a tool would have a GUI that would tailor the view of the *NDD* to
1010 the specific kind of *Document* to be negotiated. The tool would reference the schema of the
1011 *Document* being negotiated along with the *NDD* being constructed. This will supply the tool
1012 with sufficient information to make the views understandable by someone who is composing
1013 an *NDD*. This would enable that person to communicate with the tool in terms of the
1014 specifics of the *Document* to be negotiated. The tool could then construct the *NDD* instance
1015 *Document* in accord with the *NDD* schema.

1016

1017 The *NDD* references both the *CPA Template* and the *CPPA* XML schema.

1018

1019 The *NDD* consists of a variable length (cardinality 1 or more) set of [XPATH] expressions, each
1020 of which refers to a negotiable element or attribute.

1021

1022 With each XPATH expression, the negotiability of the element or attribute is defined by child
Automated Negotiation Specification   **Page 30 of 97**

1023 elements. These child elements represent the negotiability characteristics of the element or
1024 attribute identified by the XPATH statement. Examples are:
1025 • Cardinality (range of permitted cardinalities)
1026 • For a numeric value, minimum, maximum, and negotiation step size
1027 • For choices, XPATH statements, ID attribute values, qnames, element values, etc. which
1028   identify the specific choices within the *Document* being negotiated. Examples in the *CPA*
1029   are certificates, delivery channels, transport protocols, and signature algorithms.
1030
1031 The following rules define what is negotiable at the point referenced by an XPATH expression:
1032
1033 1. If the XPATH expression references a non-leaf element, that element, and the whole sub tree
1034   below that element, are negotiable.
1035 2. If the XPATH expression references any attribute, it means that only that attribute is
1036   negotiable and doesn't imply anything about the containing element or the rest of the sub tree
1037   descended from the element containing that attribute.
1038 3. If the XPATH expression references a leaf element, only that element and its contained
1039   attributes are negotiable.
1040

1041 ## 11.3 Composition of an NDD for a CPA Template

1042 Formally, the negotiation defined in this specification begins when one *Party* presents an initial
1043 offer, consisting of a *CPA Template* and its *NDD*, to another *Party*. However, the following
1044 RECOMMENDATIONS on constructing an *NDD* for a *CPA Template* might facilitate the
1045 *Negotiation Process*.
1046
1047 If the initiating *Party* has access to the other *Party's NDD* that goes with its *CPP*, the initiating
1048 *Party* SHOULD use both its and the other *Party's NDD* to establish the *NDD* and *CPA Template*
1049 to be used as the initial offer. The new *NDD* would be a composite of the two sets of
1050 requirements that is acceptable to both *Parties* as a starting point in negotiation.
1051
1052 If *Party* A is composing the *NDD* of the *CPA Template*, *Party* A SHOULD exclude from the
1053 new *NDD* anything that *Party* A understands (from *Party* B's *NDD*) is not negotiable or is
1054 unacceptable to *Party* B. For example, for an enumeration, the new *NDD* SHOULD include
1055 only those choices that are common to both of the original *NDDs*. For a range of values, *Party* A
1056 SHOULD put in the new *NDD* only the common range. If, for some element, *Party* A had
1057 specified values of 1-9 and Party B had specified values of 3-12, the new *NDD* SHOULD
1058 specify values 3-9. The intersection process might identify items with no common ground,
1059 making successful negotiation unlikely.
1060
1061 *Party* A SHOULD NOT include items in the new *NDD* that were not in *Party* B's original *NDD*
1062 because *Party* B did not intend to negotiate on the items that it did not put in its original *NDD*.
1063 For those items that were not in *Party* B's initial *NDD*, *Party* A MUST either accept what is in
1064 *Party* B's *CPP* or recognize that there is an irreconcilable conflict.
1065

1066 Note that it is not mandatory for *Party* A to take *Party* B's *NDD* into account in composing the
1067 *NDD* for the *CPA Template* since incompatibilities will anyway be removed during the exchange
1068 of counter offers. However, taking *Party* B's *NDD* into account will speed up convergence (or
1069 recognition of fatal incompatibilities) and reduce the possibilities of unnecessary rejects during
1070 negotiation. In other words, composing a CPA *Template* and combined *NDD* before starting
1071 negotiation simplifies the *Negotiation Process* by:
1072 1. Removing subjects from negotiation that can be handled by simple matching.
1073 2. Quickly recognizing the existence of fatal incompatibilities. For fatal incompatibilities,
1074   human to human contact to resolve the incompatibilities is RECOMMENDED.

1075 **11.4 Explanation of Contents of NDD**

1076 This section discusses the schema and example of an *NDD* instance *Document*. See Appendix A
1077 for the schema and Appendix F for an example of an instance *Document*.
1078
1079 The ***NegotiationDescriptor*** element is the top element of an *NDD*. It is is a container element,
1080 that contains one or more instances of the negotiable parts called ***NegotiableInformationItem***
1081 elements. The ***documentLocation*** attribute of the ***NegotiationDescriptor*** element is a URI that
1082 points to the XML *Document* to which this *NDD Document* corresponds. For instance, if the
1083 *NDD* pertains to a *CPA*, the ***documentLocation*** attribute points to that *CPA*.
1084
1085 Each ***NegotiableInformationItem*** element contains an ***xpath*** attribute that identifies the
1086 negotiable information item with respect to the *Document* pointed to by the ***documentLocation***
1087 attribute of the ***NegotiationDescriptor*** element.
1088
1089 Each negotiable information item (which could be an XML element or an attribute) is one of the
1090 following types, depending on what kind of negotiation that one needs to perform on this
1091 negotiable information item.
1092
1093 1. Value: For negotiating the value of the item.
1094 2. UnorderedValue: For negotiating the presence or absence of a member of a set of unordered
1095   values.
1096 3. OrderedValue: For negotiating to choose among the members of a set of ordered values,
1097   where the preference is of a simple kind (namely, a preference for earlier values or later
1098   values in the set).
1099 4. ValuesWithPreferenceMeasure: For negotiating to choose among the members of a set of
1100   values, where the preference measure is of a more complicated nature (for instance,
1101   expressed by a piecewise linear function or a function defined by an equation).
1102 5. PresentOrNot: For negotiating the presence or absence of a value. This type allows one to
1103   express that a *Party* (a) insists that a value MUST be present; (b) insists that a value is
1104   absent; (c) is ok with the value being present or absent, but has a preference for one or the
1105   other or (d) is o.k with the value being present or absent, and has no preference.
1106 6. IntegerValues: For expressing (a) whether an integer value is present or not (as in
1107   PresentOrNot) and then (b) the choice between different integer values using simple
1108   preference measures (such as smaller ones being preferred or larger ones being preferred) or
1109   more complicated preference measures (such as those expressible via piecewise linear
1110   functions). This type is provided mostly for convenience, since there are many entries in a
1111   *CPP* or *CPA* that impose these kinds of negotiation requirements.

Automated Negotiation Specification   **Page 32 of 97**

7. Preference: For expressing preference among values of a similar nature (such as multiple elements at the same level, e.g., the ***PartyInfo*** element)
8. Cardinality: Similar to IntegerValues.
9. BooleanValues: For expressing (a) whether a Boolean-valued item is present or not and then (b) for expressing preference for either true or false as the value of the boolean-valued item.
10. DurationWithPreference: For expressing (a) whether a duration-valued item is present and then (b) to give maximum and minimum possible values of the duration and to express a preference for smaller values or larger values.

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

THE FOLLOWING ARE OPEN QUESTIONS:

PIECEWISELINEARPIECE ELEMENT: THE SPEC SHOULD EXPLAIN THE FUNCTION OF X1 AND Y1. PERHAPS WE COULD GIVE THESE ELEMENTS SELF-EXPLANATORY NAMES.

PIECEWISELINEARPIECE (SCHEMA): COULD X1 AND Y1 BE TYPED MORE STRONGLY THAN "XS:STRING" AND "XS:NMTOKEN"? I GUESS THAT THIS IS NOT POSSIBLE IF THEY ARE TO BE USED WITH A VARIETY OF DATATYPES. COULD WE STATE A RULE THAT RELATES THE TYPE OF EACH INSTANCE OF X1 AND Y1 TO THE TYPE OF THE ELEMENT THAT THEY ARE WORKING WITH (E.G. DATETIME FOR THE START AND END ELEMENTS)?

FOR GIVING THE ENDPOINTS FOR ELEMENTS LIKE START/END, THE TYPE IS CURRENTLY SET TO STRING BEACUSE XML SPY DOES NOT SEEM TO VALIDATE DATETIME ENTRIES CORRECTLY, BUT SHOULD BE CHANGED TO DATETIME LATER

DO WE NEED AN ADDITIONAL CONSTRUCT IN THE NEDD TO IDENTIFY AN ELEMENT OR ATTRIBUTE FOR WHICH THE OFFEREE MUST SUPPLY A VALUE (E.G. PARTYID) BUT IS NOT NEGOTIABLE?

SEE OTHER OPEN QUESTIONS IN THE NEGOTIABILITY SECTION.

12 Negotiation Messages

A *Negotiation Message* includes the details of a offer or a counter offer, identification of the *NDD* and *CPA Template* being negotiated, and information that controls the *Negotiation Protocol*. Some *Messages* include the *NDD* and the *CPA Template* or their URLs.

This section defines and discusses the details in the *Negotiation Message* in terms of the individual XML elements and attributes. The discussion is illustrated with XML fragments.

See Appendix B for the complete *Negotiation-Message* XML Schema. See Appendix G for examples of *Negotiation-Message* instance *Documents*.

12.1 Negotiation Message Structure

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

```xml
<NegotiationMessage
        xmlns:tp="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpa-negot-1_0.xsd"
        xmlns:xsi=http://www.w3.org/2001/XMLSchema-instance xmlns:cppa="http://www.oasis-
        open.org/committees/ebxml-cppa/schema/cpp-cpa-2_0.xsd"
        xsi:schemaLocation="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpa-negot-
        1_0.xsd"
        businessMsgId="busMsg002"
        binding="false"
        inresponseTo="busMsg001"
        negotiationDialogId="negotDialog001"
        offerId="offer001"
        messageType="CounterOffer"
        error="ExpiredCPP">
  <NCPA uri="http://..." />
  <CPAIdentity>
        …
  </CPAIdentity>
  <cppa:SecurityDetails cppa:securityId="ID">
        …
  </cppa:SecurityDetails>
  <InitiatingParty>
        …
  </InitiatingParty>
  <RespondingParty>
        …
  </RespondingParty>
  <BPSSBusinessDocumentName name="CPA_Counter_Offer_Doc" />
  <ExpirationDate>…</ExpirationDate>
  <BusinessDocuments>
        …
  </BusinessDocuments>
  <NegotiationContent>
        …
  </NegotiationContent>
```

```
1196    <ResponseToURL>
1197    …
1198    </ResponseToURL>
1199    <Comment/>
1200  </NegotiationMessage>
```

12.1.1 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:

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

The *NegotiationMessage* element contains the following attributes:

- a REQUIRED *businessMsgId* attribute that uniquely identifies the current *Business Message* within the scope of one *Negotiation Dialog*,
- a REQUIRED *negotiationDialogId* attribute that uniquely identifies an ongoing *Negotiation Dialog* that connects multiple offer/counter-offer transactions that pertain to the same *CPA Template*,
- an IMPLIED *offerId* attribute that uniquely identifies each instance of an offer or counter offer,
- an IMPLIED *inresponseTo* attribute that identifies the unique *Business Message* of the previous offer or counter-offer that this *Business Message* is responding to. Its value can be null for the initiating offer of the *Negotiation Dialog*,
- a REQUIRED *binding* attribute that indicates whether the current *Message* is legally binding,
- a REQUIRED *messageType* attribute that indicates the status of current negotiation. The legal values for the *messageType* attribute are:
  - ♦ "Offer"
  - ♦ "CounterOffer",
  - ♦ "CounterOfferPending"
  - ♦ "Rejected"
  - ♦ "Accepted"
  - ♦ "Expired"
  - ♦ "SinglePartySigned"
  - ♦ "Signed"
  - ♦ "Unsigned"

IN THE NEGOTIATION BPSS INSTANCE DOCUMENT, THE "CPA FINAL RESPONSE DOC" DOCUMENT IS USED FOR BOTH ACCEPTANCE AND REJECT. THIS REQUIRES A SUCCESS/FAILURE INDICATOR IN THE MESSAGE AND THAT INDICATOR WILL HAVE TO BE CHECKED BY THE RECIPIENT WHENEVER THE MESSAGE IS "CPA FINAL RESPONSE DOC." MESSAGETYPE IS INTENDED TO BE

Negotiation.spec.20Jan03.doc   1/24/2003 6:30 PM

THAT IDICATOR. HOWEVER, IT ISN'T CLEAR THAT THE WAY MESSAGETYPE IS DEFINED IS CONSISTENT WITH THE BPSS INSTANCE DOCUMENT. WHEN THE CHOREOGRAPHY CODE RECEIVES THE CPA FINAL RESPONSE DOC" MESSAGE, IT IS EXPECTING TO TEST THE MESSAGETYPE ELEMENT ONLY FOR "REJECTED" OR "ACCEPTED". ARE WE CERTAIN THAT THESE ARE THE ONLY TWO VALUES THAT MESSAGETYPE COULD HAVE IN THE CPA FINAL RESPONSE DOC MESSAGE? I BELIEVE THAT MOST OF THE VALUES CORRESPOND TO SPECIFIC BPSS BUSINESS DOCUMENT NAMES, SO IT MAY BE OK IT IS. ONE VALUE THAT IS OF CONCERN IS "EXPIRED". COULD "EXPIRED" BE A VALID VALUE OF THE MESSAGETYPE ELEMENT IN THE CPA FINAL RESPONSE DOC MESSAGE? IF SO, THE CHOREOGRAPHY SOFTWARE WON'T KNOW WHAT TO DO WITH IT. ISN'T "EXPIRED" AS A VALUE OF MESSAGETYPE THE SAME AS "EXPIREDOFFER" (THE VALUE OF THE ERROR ATTRIBUTE)? CAN WE SIMPLY DELETE "EXPIRED" FROM THE PERMISSIBLE VALUES OF MESSAGETYPE? (PRESUMABLY THIS REQUIRES A FIX TO THE XSD AND MAYBE TO THE INSTANCE DOCUMENT EXAMPLES.)

DALE MOBERG: THE ABOVE SET OF VALUES MAY BE INCOMPLETE. MARTY SACHS: THE ERROR ELEMENT IS A MODIFIER TO "REJECTED". POSSIBLY, THE MISSING SET OF VALUES IS GIVEN BY THE ERROR ELEMENT.

- An IMPLIED **error** attribute that identifies the error code in case of rejection for a CounterOffer.

The *NegotiationMessage* element SHALL consist of the following child elements:
- One REQUIRED *NCPA* element to identify the current *NCPA*
- One REQUIRED *CPATemplateId* element to identify the *CPA* that is being negotiated.
- One REQUIRED *cppa:SecurityDetails* element to describe the security requirements of current negotiation.
- One REQUIRED *InitiatingParty* element to describe the initiating *Party* of the current *Negotiation Dialog*.
- One REQUIRED *RespondingParty* element to describe the responding *Party* of the current *Negotiation Dialog*.
- One REQUIRED *BPSSBusinessDocumentName* element that indicates the name of the *Business Document* this offer relates to.
- One REQUIRED *ExpirationDate* element that specifies the date when this offer or counter offer expires. Its XML data type is "dateTime".
- One REQUIRED *BusinessDocuments* element that describes the *CPA Template*.
- One REQUIRED *NegotiationContent* element that itemizes accepted, rejected, updated elements within the current *CPA*.
- One IMPLIED *ResponseToURL* element that identifies the return (http) address where the responding *Party* can send its response asynchronously. See Section 12.1.10 for more information.
- One IMPLIED *Comment* element that can be used to record free text. For example, if the counter offer is a rejection, the *Party* can list its contact information here for a more traditional negotiation that will involve humans.

Automated Negotiation Specification   **Page 36 of 97**

12.1.2 NCPA element

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

12.1.3 CPATemplateId element

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

12.1.4 cppa:SecurityDetails element

The *cppa:SecurityDetails* element is defined in detail in the "*SecurityDetails* element" section of [ebCPP].

12.1.5 InitiatingParty element

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

The *CPPId* element has two attributes:
- A REQUIRED *id* attribute, which SHALL contain the correct value as specified in [ebCPP] for the *cppid* attribute of the *CollaborationProtocolProfile* element in a *CPP* document.
- A REQUIRED *version* attribute. See [ebCPP] for the definition of the *version* attribute of the *CollaborationProtocolProfile* element.

The *CPPNDD* element contains either a child *BinaryDoc* element or a child *Uri* element. The *BinaryDoc* element has the xml type "base64Binary". Its value is the *NDD Document*. The *Uri* element SHALL have a value that is a URI that conforms to [RFC2396] and identifies the location of the *NDD* xml *Document*.

12.1.6 RespondingParty element

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

The *CPPId* element has two attributes:
- A REQUIRED *id* attribute that SHALL contain the value specified for the *cppid* attribute of the *CollaborationProtocolProfile* element in a *CPP* document. See [ebCPP].
- A REQUIRED *version* attribute. See [ebCPP] for the definition of the *version* attribute of the *CollaborationProtocolProfile* element.

The *CPPNDD* element contains either a child *BinaryDoc* element or a child *Uri* element. The

Deleted: *CPATemplateId* **has**

Automated Negotiation Specification   **Page 37 of 97**

BinaryDoc element has the xml type "base64Binary". Its value is the *NDD* Document. The *Uri* element SHALL have a value that is a URI that conforms to [RFC2396] and identifies the location of the *NDD* xml *Document*.

12.1.7 BPSSBusinessDocumentName element

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

- "CPA_Offer_Doc"
- "CPA_Accept_Offer_Doc"
- "CPA_Counter_Pending_Offer_Doc"
- "CPA_Counter_Offer_Doc"
- "CPA_Reject_Offer_Doc"
- "CPA_Final_Doc"
- "CPA_Final_Response_Doc"
- "CPA_Final_Response_Doc_Signed"

> **Formatted:** Default Text

DALE MOBERG EXPRESSED CONCERN ABOUT WHAT SPACE IN ENUMERATED VALUES. HIS ORIGINAL COMMENT WAS ABOUT A VALUE IN THE MESSAGETYPE ENUMERATION. THAT WHITE SPACE WAS A TYPOGRAPHICAL ERROR IN THE TEXT AND HAS BEEN REMOVED. IS THERE ANY CONCERN ABOUT THE ABOVE LIST? HERE, THE WHITE SPACE MATCHES THE WHITE SPACE IN THE DOCUMENT NAMES IN THE BPSS INSTANCE DOCUMENT. REMOVING THE WHITE SPACE WILL AFFECT THE MESSAGE SCHEMA AND EXAMPLES, BPSS INSTANCE DOCUMENT AND MANY INSTANCES OF THE ABOVE NAMES IN THIS SPECIFICATION.

12.1.8 BusinessDocuments element

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

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

Throughout one negotiation process, the most up-to-date *CPATemplate* SHALL be available for the other *Party*. Therefore, each *Party* SHALL include either the updated *CPA Template* or a URI reference to it in each message. The receiving *Party* can reference it while examining the

1372 *NegotiationContent* element, which identifies the delta between the current copy of
1373 **CPATemplate** and the one that was sent out earlier. See Section 13.10 for a definition of
1374 "updated *CPA Template".*

1375 **12.1.9 NegotiationContent element**

1376 For an initial offer, the **NegotiationContent** element can be empty. If the offering *Party* had
1377 created the initial *CPA Template* and *NDD* by modifying information in the other *Party's CPP*
1378 and *NDD,* the **NegotiationContent** element in the initial offer SHALL describe the changes made
1379 by the offering *Party* to the information in the other *Party's CPP* or *CPA Template* when
1380 forming the *CPA Template* of the initial offer. For counter offers within this *Negotiation Dialog,*
1381 the **NegotiationContent** element SHALL describe the sending *Party's* proposed modifications to
1382 the *CPA Template.*

1384 The **NegotiationContent** element SHALL list all items accepted by the sending *Party* since the
1385 start of the *Negotiation Dialog* (including the ones being accepted by this *Message*). The
1386 **NegotiationContent** element does not contain items accepted by the other *Party* since the start of
1387 the negotiation.

1389 The **NegotiationContent** element SHALL list all items updated, deleted, or inserted by the
1390 sending *Party* since the sending *Party* received the previous offer or counter offer. Any item that
1391 has been deleted by one party can no longer be re-inserted in future counter offers.

1393 It is up to the receiver of this *Message* to decide whether to continue negotiate, accept, or reject
1394 changes listed within the **NegotiationContent** element.

1396 The **NegotiationContent** element contains the following child elements. In the following
1397 information, the Xpath of the item points to the item in the most recent update of the *CPA*
1398 *Template,* i.e. the one that is included in or referenced by this *Message.*

1400 *THE DESCRIPTIONS OF ACCEPTED, DELETED, UPDATED, AND INSERTED*
1401 *SHOULD BE WRITTEN IN TERMS OF THE NAMES OF THE ELEMENTS (E.G.*
1402 *ACCEPTEDITEM) AND THEIR ATTRIBUTES, AS WE HAVE DONE EVERYWHERE*
1403 *ELSE.*

1405 • Accepted Items[1](1…n): These are the items that have been accepted by the sending Party
1406   during all exchanges prior to this *Message* and within the same *Negotiation Dialog.*
1407   ♦ Xpath of item
1408   ♦ The status that indicates acceptance of this item is "Required" or "Preferred".

1409   **Formatted:** Default Text

1410 *SINCE ALL ACCEPTED ITEMS ARE LISTED, THE MESSAGE RECIPIENT CNNOT*
1411 *QUICKLY TELL WHAT NEW ACCEPTANCES ARE IN THE MESSAGE. SHOULD*
1412 *THERE BE A WAY TO FLAG THE NEW ACCEPTANCES?*

1414 • Deleted Items (0…n)

---

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

Negotiation.spec.20Jan03.doc   1/24/2003 6:30 PM

1415 ◆ Xpath of item
1416 ◆ The status that indicates deletion of this item is "Required" or "Preferred".
1417 • Updated Items (0…n)
1418 ◆ Xpath of item
1419 ◆ Original value of item
1420 ◆ Proposed value of item
1421 ◆ The status that indicates update of this item is "Required" or "Preferred".
1422 • Inserted Items (0…n)
1423 ◆ Xpath of item
1424 ◆ Proposed value of item
1425 ◆ The status that indicates addition of this item is "Required" or "Preferred".
1426 *WE NEED TO EXPLAIN WHAT KINDS OF ITEM CAN BE INSERTED, UNDER WHAT*
1427 *CONDITIONS, AND HOW TO EXPRESS THE POSSIBLITY OF INSERTION IN THE*
1428 *NDD.*
1429
1430 Any of the items under the *NegotiationContent* element can be either a leaf node or non-leaf
1431 node. A non-leaf node indicates that the entire subtree under that node is subject to the
1432 corresponding change action. In other words, the entire subtree has been accepted, deleted,
1433 updated, or inserted though for update, not all items within the subtree have necessarily changed.
1434 See Figure 3 and Figure 4 for an example of how these elements can be used in a negotiation.
1435
1436 *WE NEED A PARAGRAPH THAT DESCRIBES THE KEY POINTS ILLUSTRATED IN*
1437 *THESE TWO FIGURES.*
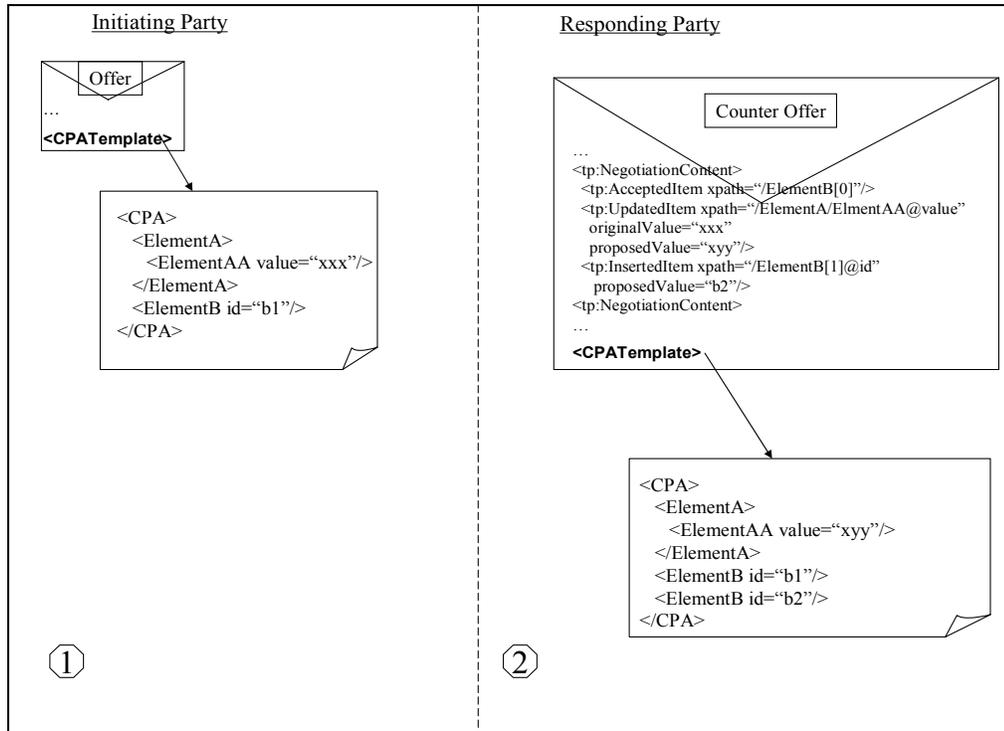
Figure **3**, **Example of how to use NegotiationContent Element - Step 1**
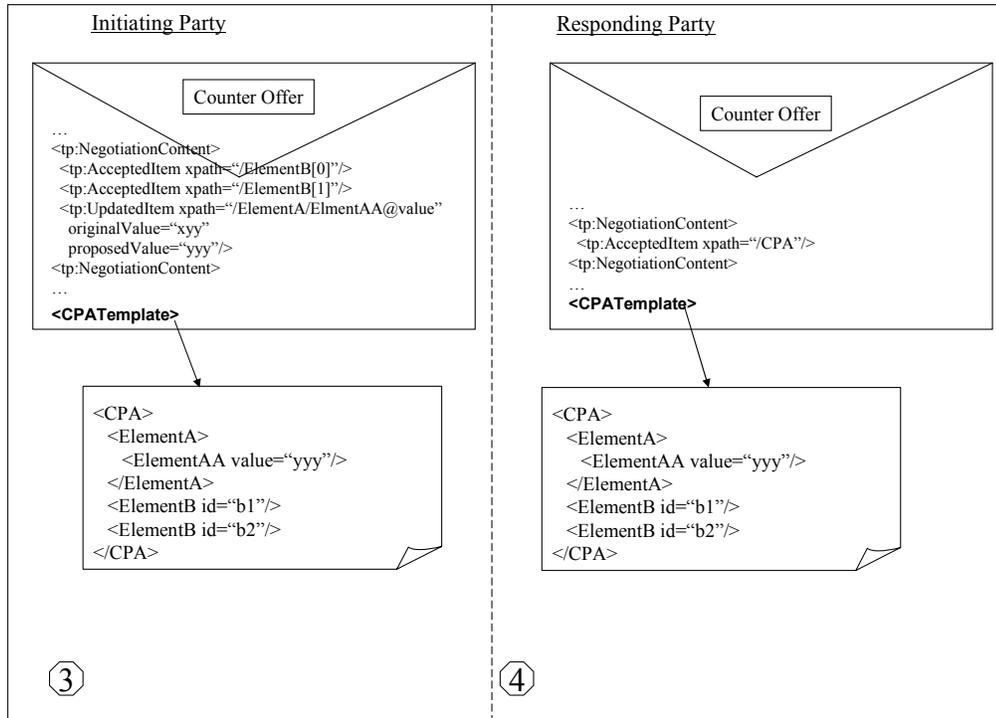
Figure 4, Example of how to use the NegotiationContent Element, Step 2

When the *NegotiationMessage* element has its *messageType* attribute set to "CounterOfferPending", the counter offer might consist of:

- Deleted elements and attributes.
- Inserted elements and attributes.
- Re-ordered elements using a [XPATH]-based list of changes with status of "required" or "preferred".
- Changed values of elements and attributes.
- Rejection: with reason(s) for rejection. See Section 12.4 for additional information. Rejection is final. It ends the *Negotiation Dialog* and the two *Parties* SHOULD make human to human contact to resolve their incompatibilities.

FOLLOWING ARE ADDITIONAL CONSIDERATIONS ABOUT THE RESPONSE MESSAGE.
- CONSIDER PHYSICALLY PACKAGING THE RESPONSE MESSAGE WITH THE COUNTER OFFER IF ONE IS BEING ISSUED, IN ORDER TO SAVE MESSAGE TRAFFIC.
        + CAN THIS BE DONE USING EXISTING BUSINESS SIGNALS FOR THE RESPONSE INDICATOR (IN ORDER TO AVOID CPPA CHANGES)?
- IT WAS SUGGESTED THAT THIS PACKAGING MIGHT BE UNNECESSARY COMPLEX, ESPECIALLY FOR VERSION 1).

Negotiation.spec.20Jan03.doc   1/24/2003 6:30 PM

12.1.10 ResponseToURL element

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

12.2 CPA ID, Negotiation Dialog ID, Unique Business Message ID, and InResponseTo

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

The *Negotiation-Dialog Identifier* is used to identify a particular *Negotiation-Dialog* thread. See the discussion of the *Negotiation-Dialog Identifier* in Section 13.3. The value of the *id* attribute of the *CPATemplateId* element SHALL NOT be used as the *Negotiation-Dialog Identifier*. See the discussion in Section 13.3.

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

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

12.3 Offer and Counter Offer

THE PARAGRAPH WHICH WAS HERE HAS BEEN MOVED TO SECTION 13.10 (OFFER AND COUNTER OFFER).

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

THE PARAGRAPH WHICH WAS HERE HAS BEEN MOVED TO SECTION 13.10 (OFFER AND COUNTER OFFER).

12.4 CPA Offer Rejected

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

The process of composing the *CPA* from *CPPs* will detect many error conditions before the *Negotiation Process* begins. Others might be discovered during the *Negotiation Process*. Examples are mismatched Process Specification *Document* and mismatched delivery channel requirements. See the *CPA*-composition appendix of [ebCPP] for information about error

Automated Negotiation Specification   **Page 43 of 97**

conditions that can be detected during composition of a *CPA Template*.

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

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

- *CPA Template* contents. Examples:
    - ♦ `"ExpiredCPP"`: Expired *CPP*
    - ♦ `"UnableToFulfillSecurityRequirements"`: Unable to fulfill security requirements
    - ♦ `"ProposedSecurityPolicyInadequate"`: Proposed security policy is inadequate
    - ♦ `"OurOfSequenceCounterOffer"`: Out-of-sequence counter offer
    - ♦ `"FailedSignatureValidation_CPATemplate"`: The signature on the *CPA Template* failed validation.
    - ♦ `FailedSignatureValidation_CPA"`: The signature on the agreed *CPA* failed validation
        - The *CPA* is not signed until it is agreed to.
    - ♦ `"UnsupportedPackaging"`: proposed *Packaging* not supported
    - ♦ `"UnsupportedSignal"`: unable to support signals requested (BPSS instance *Document*)
- Business relationship
    - ♦ `"UnsupportedBusinessRelationship"`: *CPA* unsupported without existing *Business* relationship.
- Negotiation Process
    - ♦ `"FailedToConverge"`: In the judgment of the rejecting *Party*, too many counter offers were tried with no forward progress toward convergence.

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

    - ♦ `"PreviouslyRejectedCPA"`: Proposed *CPA* previously received and not accepted.
- `"ExpiredOffer"`: The current offer's validity interval has expired.
- `"FormatError"`: CPA format problems
    - ♦ Examples: parsing error, data invalid
- `"UnknownSystemError"`: Internal System Error

12.5 CPA Offer accepted

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

1547

1548 *THERE IS ONE MINOR PROBLEM WITH THE ABOVE PARAGRAPH. AT THE END, A*
1549 *COMPLETED CPA IS EXCHANGED AND THAT CPA IS CARRIED BY OR*
1550 *REFERENCED BY THE CPATEMPLATE ELEMENT IT MIGHT BE ARGUED THAT*
1551 *THE NAME OF THE ELEMENT IN THAT CASE SHOULD BE COMPLETEDCPA AND*
1552 *NOT CPATEMPLATE. HOWEVER, NOTHING ELSE WOULD CHANGE EXCEPT THE*
1553 *NAME OF THE ELEMENT, AND THERE IS PLENTY OF CONTEXT, INCLUDING*
1554 *BUSINES DOCUMENT NAME, TO RECOGNIZE THAT IT IS A COMPLETED CPA, SO*
1555 *MAYBE IT IS SUFFICIENT SIMPLY TO EXPLAIN IN THE TEXT.*

1556

Negotiation.spec.20Jan03.doc   1/24/2003 6:30 PM

13 Negotiation Protocol

13.1 General Principles of Negotiation Protocol

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

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

13.2 CPA Identifier

When a *Party* creates a *CPA Template*, that *Party* SHALL assign a valid value to the *cpaid* attribute in the *CPA Template*. See Section 10.8 regarding negotiability of the *cpaid* attribute.

13.3 Negotiation-Dialog Identifier

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

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

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

13.4 Offer Identifier

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

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

Deleted: cpaId

Deleted: cpaId

1598   NOTE: With ebXML *Messaging*, the **messageId** and **refToMessageId** attributes in the
1599   *Message* header could serve the purpose of the *Offer Identifier*. However, to enable
1600   alternative *Messaging* protocols, such as "vanilla SOAP", which do not have these
1601   identifiers, the *Offer Identifier* is defined at the application level.

13.5 Negotiation Status

1602

1603   The **Status** element in the *CPA* records the state of the composition and *Negotiation Protocol*.
1604   The states of its **value** attribute progress as follows:

1605   • "Proposed" – This value is in the *CPA Template* sent with the initial offer and remains
1606   unchanged until an agreed *CPA* is completed.
1607   • "Agreed" – This value is in the completed *CPA* that is sent from one *Party* to the other for
1608   validation if the *Parties* had agreed not to sign the *CPA*. This is the final state.
1609   • "Signed" – If the *Parties* had agreed to sign the *CPA*, the *CPA* sent from one *Party* to the
1610   other *Party* is signed by the sending *Party* and the value of the **value** attribute is "Signed".
1611   This is the final state.

1612

1613   NOTE: Because the **Status** element is included in the first *Party's* signature, the value of the
1614   **value** attribute cannot be changed when the second *Party* signs.

13.6 ebXML Conversation

1615

1616   A single *Negotiation Dialog* corresponds to a single ebXML *Conversation.*

1617

1618   ***THIS SECTION MUST DEFINE THE BEGINNING AND END OF A CONVERSATION***
1619   ***WITH RESPECT TO THE CHOREOGRAPHY DEFINED IN THE BPSS INSTANCE***
1620   ***DOCUMENT. THE RECEIVING SYSTEM MUST BE ABLE TO ASSOCIATE THE***
1621   ***CONVERSATION ID OF AN INCOMING MESSAGE TO A NEGOTIATION INSTANCE***
1622   ***IDENTIFIER.***

1623

1624   For use with *Message* services, such as "vanilla SOAP", that have no *Conversation* construct, the
1625   *Negotiation-Dialog Identifier* serves the purpose of a *Conversation* identifier at the application
1626   level

13.7 Negotiation CPA

1627

1628   Prior to the initial offer, a *Negotiation CPA* MUST be activated between the two negotiating
1629   *Parties*. See Section 5.1 for a possible scenario.

13.8 Initial Offer

1630

1631   A *Party* (B) can create and send an initial offer to another *Party* (A) in different ways, depending
1632   on whether *Party* B is starting with *Party* A's *CPP* or *CPA Template*.
1633   • If *Party* B discovered the *CPP* of *Party* A (a potential trading partner), *Party* B composes a
1634   *CPA Template* from its *CPP* and *Party* A's *CPP*. *Party* B then prepares an *NDD* that
1635   describes what is negotiable in the *CPA Template*. If *Party* A had also published an *NDD*,
1636   *Party* B SHOULD take that *NDD* into account in preparing the *NDD* for the initial offer.
1637   • If *Party* B discovered the *CPA Template* and *NDD* of *Party* A, *Party* B modifies the *CPA*
1638   *Template* to include information about itself, makes other modifications to negotiable items
1639   in the *CPA Template* that are indicated in the *Party* A's *NDD*, and prepares a new *NDD* to go
1640   with the modified *CPA Template*.

Automated Negotiation Specification   **Page 47 of 97**

1641

1642 In either case, *Party* B is also responsible for inserting into the *CPA Template* the **Start**, **End**, and
1643 other elements that are present in a *CPA* but not in a *CPP*.

1644

1645 If *Party* B creates the initial offer by modifying *Party* A's published *CPP* or *CPA Template*,
1646 *Party* A SHOULD include a list of changes (Accepted, Deleted, Updated, Inserted) in the initial-
1647 offer *Message* (Negotiation Content section) in addition to the initial-offer information

1648

1649 *Party* B then submits the new *CPA Template* and *NDD* to *Party* A as an initial offer.

1650

1651 It is RECOMMENDED that the *CPA Template* in an initial offer be signed by the offering *Party*.

13.9 Simultaneous Initial Offers

1653 Two *Parties* might simultaneously discover each other and send each other initial offers. Since
1654 the two initial offers will cause creation of two independent *Negotiation Dialogs*, this race
1655 condition might only be discoverable and resolvable at the application level. Human contact will
1656 be necessary to decide which *Negotiation Dialog* to proceed with.

13.10 Offer and Counter Offer

1658 When a *Party* proposes an offer or counter offer, the details of the offer or counter offer are
1659 expressed in a negotiation *Message*. The original *NDD* SHALL NOT be altered during the
1660 course of the negotiation.

1661

1662 If *Party* A initiates the *Negotiation Dialog* by sending *Party* B an offer, *Party* B sends back a
1663 counter offer. In order to counter this counter offer, *Party* A sends another counter offer to *Party*
1664 B. In other words, only the initiating *Message* is an offer; the rest of the negotiation will be
1665 conducted by exchanging counter offers. Each counter offer message contains or references an
1666 updated version of the *CPA Template* that contains all changes up to, and including, the latest
1667 changes accepted prior to this *Message.*

1668

1669 Throughout the *Negotiation Dialog*, each *Party* can terminate the negotiation by sending "CPA
1670 offer Rejected" in responding to an incoming offer or counter offer. Human to human contact is
1671 encouraged after "CPA offer Rejected" is sent in order to resolve any impasse before initiating a
1672 brand new *Negotiation Dialog*.

1673

1674 A counter offer SHALL only refer to items that are listed in the *NDD*. Any offer or counter offer
1675 that is outside the limits defined in the *NDD* MUST be rejected.

1676

1677 A counter offer SHALL NOT propose a wholesale change of subject matter. For example a
1678 counter offer SHALL NOT propose changes in the *Roles* of the participants.

1679

1680 A *Party* that wishes to propose a different BPSS instance *Document* SHALL reject the received
1681 offer or counter offer and can then issue its own initial offer including the desired BPSS instance
1682 *Document*.

1683

1684 A counter offer SHALL NOT introduce a new *NDD*. To introduce a new *NDD*, a *Party* SHALL
1685 reject the received offer or counter offer and can then issue its own initial offer including the

1686   desired *NDD*.
1687
1688   When responding to an offer or counter offer, a *Party* SHALL indicate in its counter offer, which
1689   items in the prior offer or counter offer it accepted.
1690
1691   If a counter offer contains only indications of acceptance of items, the *Party* that sent it is
1692   indicating acceptance of the *CPA* as modified by the prior steps in the negotiation.
1693
1694   Once agreement has been reached on any part of the *CPA,* those elements and attributes SHALL
1695   NOT be reopened for negotiation.
1696

1697   13.10.1 Responses to Offer and Counter Offer

1698   A number of responses can be given to an offer or counter offer. The responses fall into the
1699   following categories:
1700   • Acceptance: Acceptance of an offer or counter offer means that the *Party* that received the
1701   offer or counter offer is accepting all remaining open items and hence the two *Parties* have
1702   reached agreement.
1703   • Counter offer pending: The *Party* that received the offer or counter offer wishes to negotiate
1704   further on some or all open items and is going to send its own counter offer.
1705   • Rejection: The *Party* that received the offer or counter offer believes that agreement cannot
1706   be reached. Human contact is REQUIRED in order to resove the incompatibilities.
1707   • The responses are discussed in Section 12.4.

1708   13.10.2 Offer-Counter Offer Acceptance Time

1709   A maximum time (interval) for acceptance is associated with each offer or counter offer. The
1710   acceptance interval is a business-level timeout; processing it is independent of any *Document*-
1711   exchange or transport-level *Message*-loss recovery rules. When the acceptance interval expires
1712   without a response, the initiator SHALL record the current *Negotiation Dialog* as terminated.
1713
1714   *CONSIDER WHETHER THE BPSS TIME TO PERFORM ATTRIBUTE (BINARY*
1715   *COLLABORATION) CAN BE USED TO REPRESENT THE ACCEPTANCE INTERVAL.*

1716   13.11 Time to Complete Negotiation

1717   *IS THERE A WAY OF SPECIFYING THE MAXIMUM TIME TO COMPLETE A*
1718   *NEGOTIATION FROM INITIAL OFFER TO COMPLETION? IS THERE A BPSS TIME*
1719   *ATTRIBUTE THAT CAN BE USED? BPSS ATTRIBUTES CANNOT BE NEGOTIATED*
1720   *WITHOUT NEGOTIATING THE NEGOTIATION CPA. WHAT ABOUT A TIME THAT*
1721   *COULD BE EXPRESSED IN THE NDD AND NEGOTIATED? SHOULD THIS BE LEFT*
1722   *FOR THE FUTURE?*

1723   13.12 Conclusion of Negotiation

1724   The negotiation concludes when agreement has been reached. This might happen either by one
1725   *Party* accepting the initial offer or following an exchange of counter offers.
1726
1727   If agreement is reached on the initial offer, and the *Party* that received the initial offer does not
1728   have to add any information to the *CPA Template*, the negotiation concludes immediately. The

Automated Negotiation Specification   **Page 49 of 97**

1729 *Party* that received the initial offer SHALL send a *Message* indicating acceptance and the *CPA*
1730 *Template* becomes the agreed *CPA*. If signing is included in the initial offer, the offering *Party*
1731 SHALL sign the *CPA Template* before sending it. The receiving *Party* SHALL then sign and
1732 return the *CPA*. At this point, the *Parties* are ready to deploy the *CPA* into their run-time systems
1733 and commence business. If the second *Party* does not agree to sign, and signing is negotiable, it
1734 SHALL respond with a counter offer that excludes signing instead of accepting the initial offer.
1735
1736 When agreement has been reached following exchanges of counter offers, the *Party* that received
1737 and accepted the final counter offer SHALL send the completed *CPA* (or its URL) to the other
1738 *Party* for approval. The receiving *Party* SHALL respond, indicating either approval or rejection.
1739 If signing was agreed to, the sending *Party* SHALL sign the *CPA* before sending it. The
1740 receiving *Party* SHALL check that the new *CPA* conforms to its understanding of the contents of
1741 the *CPA*. The receiving *Party* can also validate the first *Party's* signature. If the receiving *Party*
1742 approves the *CPA*, the receiving *Party* SHALL sign the *CPA* over the first *Party's* signature and
1743 return it to the first *Party*. Otherwise the receiving *Party* SHALL respond indicating rejection.
1744
1745 The *Party* that received the completed *CPA* SHALL respond in one of the following ways:
1746 • *Message* indicating that a completed *CPA* was received (**BusinessDocument** name = "CPA
1747   Final Response Doc")
1748   ♦ A separate indicator in the *Message* distinguishes between accept and reject.
1749 • *Message* that sends a completed *CPA* signed by the sender (**BusinessDocument** name =
1750   "CPA Final Response Doc Signed").
1751   ♦ Used when signing was agreed to and the received *CPA* was signed by the sending *Party*.
1752
1753 Following are some reasons for rejecting the received *CPA*:
1754 • The final *CPA* does not agree with the recipient's understanding of the contents of the *CPA*
1755   (some kind of state-tracking mismatch).
1756 • The signature on the final *CPA* cannot be validated.
1757 • The final *CPA* was not signed although signing was agreed to.
1758 •
1759 When signing by both *Parties* was agreed to, the *Party* that received the double-signed *CPA*
1760 SHALL test for the following conditions:
1761 • The double-signed *CPA* is acceptable.
1762 • The double-signed *CPA* is rejected. Reasons to reject this *CPA* include:
1763   ♦ The second signature on the double-signed *CPA* cannot be validated.
1764   ♦ An acknowledgment was received when a double-signed *CPA* was expected.
1765
1766 Acceptance and rejection of the double-signed *CPA* are indicated by business signals. See
1767 Section 13.14.2 for details.
1768
1769 Rejection at this stage is a fatal condition and the *Negotiation Dialog* SHALL be terminated. It is
1770 RECOMMENDED that the two *Parties* confer to resolve the discrepancy and then renegotiate
1771 the *CPA*. If the resolution of the discrepancy was successful, the renegotiation will generally
1772 consist of one *Party* sending a new offer that the other *Party* can accept without a counter offer.

Automated Negotiation Specification   **Page 50 of 97**

13.13 Signing the CPA

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*.

Negotiation of signing is accomplished by negotiating the presence of the *CPA* **Signature** element and its child ***ds:Signature*** elements. Following are the outcomes:

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

It is important to understand that the ***ds:Signature*** elements MUST be incorporated into the *CPA* one at a time, as the *Parties* sign. The **Signature** element MUST NOT be inserted into the *CPA* until the first *Party* signs. If it is incorporated earlier, the *CPA* will fail validation against the CPPA XML Schema because there will be no child ***ds:Signature*** elements.

If the *Parties* agreed to third-*Party* signing, they SHALL obtain the third *Party's* signature before commencing to do business under the *CPA*. The means of obtaining the third *Party's* signature are not defined in this specification.

MONICA MARTIN COMMENTED: CAN WE DIFFERENTIATE HERE THAT THE DIGITAL SIGNATURE IS A LEGAL SIGNING THAT MAY IMPACT THE BUSINESS RELATIONSHIP? ARE THERE ANY OTHER LEGAL CRITERIA, OR IS THIS A 'LEGAL SIGNING?' PERHAPS YOU SHOULD ACKNOWLEDGE THAT A BUSINESS LEVEL AGREEMENT WOULD BE THE DECIDING FACTOR WHETHER OR NOT THIS NEGOTIATION IS LEGALLY BINDING. YOU SPEAK ABOUT LEGALITIES IN SECTION 13.14.

THE SUBTEAM NEEDS TO DECIDE WHETHER IT IS APPROPRIATE TO DISCUSS LEGAL MATTERS IN THIS SPECIFICATION, WHICH IS REALLY ABOUT THE MECHANICS OF THE NEGOTIATION PROTOCOL. [EBCPP] IS ALSO SILENT ON LEGALITIES.

THE FOLLOWING COMMENTS ARE FROM KARTHA 10/31/02. THE LINE NUMBERS HAVE BEEN REWRITTEN AS DISPLACEMENTS FROM THE SECTION HEADINGS IN THIS DRAFT AND EXPRESSED AS "SECTION NUMBER PLUS DISPLACEMENT".

Line 13.14.1+22…24: I was expecting a Success element with a fromBusinessState value set to CPAOfferBTA (from the text in lines 13.14.1+19…21). However, in the business process instance document, there is no such Success element (under the binary collaboration CPA Negotiation BC). What is present, however, is a transition element with a fromBusinessState of CPAOfferBTA (line AppendixD+27), which transitions to the toBusinessState CPAFinalBTA. This is misleading.

Line 13.14.1+43…44: For clarity, say:"…RequestingBusinessActivity of the BusinessTransaction CPACounterOfferBT. Also, note a typo on line 13.14.1+44, the name of the RequestingBusinessActivity is

<table>
<tr><td>1820</td><td><strong style="color:red">CPA_Counter_offer_BT_ReqBA (not BT).</td></tr>
</table>

1820 **CPA_Counter_offer_BT_ReqBA (not BT).**
1821
1822
1823 **Line 13.14.1+48: Is the binary collaboration still CPA_Negotiation_BC? Or**
1824 **CPA_Negotiation_Counter_Offer_BC (as it seems fro the explanation of the transition element? If the latter,**
1825 **it might be good to point out that the business state now is CPA Counter Offer 1 BTA (once we are in in the**
1826 **CPA_Negotiation_Counter_Offer_BC) so that one can look for the correct fromBusinessState in the Success,**
1827 **Failure and Transition elements.**
1828
1829 **Line 13.14.2+5: This seems incorrect. If the initial offer is accepted, the business state seems to be CPA Final**
1830 **BTA (see line AppendixD+27). Similarly, in line 13.14.2+7, the "ReqBA" part of**
1831 **"ReqBA_Final_CPA_ …) should be deleted.**
1832
1833

13.14 BPSS Instance Document for Automated Negotiation

1835

THE FOLLOWING ARE STILL OPEN MATTERS FOR THE NEGOTIATION BPSS.
- TWO OF THE THE BUSINESS DOCUMENT NAMES CONTAIN THE CHARACTERS
"DOC" IN THE VALUE OF THE NAME ATTRIBUTE AND "doc" IN THE VALUE OF
THE NAMEID ATTRIBUTE. THEY ARE "CPA Final Response DOC/doc" AND "CPA
Final Response DOC/doc Signed". THIS IS NOT NECESSARILY A PROBLEM BUT IF
COMPARISONS OF TEXT STRINGS ARE CASE-SENSITIVE, IT COULD CAUSE SOME
CONFUSION OR PROGRAMMING ERRORS. IT WOULD BE BETTER TO USE "doc" IN
BOTH THE NAME AND THE NAMEID ATTRIBUTE AS IS THE CASE WITH ALL THE
OTHER BUSINESS DOCUMENTS.
- THERE ARE NO CONDITION TESTS FOR REJECTION CONDITIONS IN THE
EXCHANGE OF THE FINAL CPAS. REASONS CAN INCLUDE:
    + THE FINAL CPA DOES NOT AGREE WITH THE RECIPIENT'S
    UNDERSTANDING OF WHAT SHOULD BE IN IT (SOME KIND OF STATE-
    TRACKING MISMATCH).
    + THE FINAL CPA WAS NOT SIGNED ALTHOUGH SIGNING WAS AGREED TO.
    THE SIGNATURE ON THE FINAL CPA CANNOT BE VALIDATED.
    THE SECOND SIGNATURE ON THE DOUBLE-SIGNED CPA CANNOT BE
    VALIDATED.
    + AN ACKNOWLEDGMENT WAS RECEIVED WHEN A DOUBLE-SIGNED CPA
    WAS EXPECTED.

- THE "CPA Final Response doc" DOCUMENT IS USED FOR BOTH ACCEPTANCE AND
REJECT. EXCEPT FOR THIS CASE, A MESSAGE RECEIPIENT CAN DETERMINE
SUCCESS OR FAILURE FROM THE BUSINESS DOCUMENT NAME IN THE
MESSAGE. FOR "CPA Final Response doc" WE NEED A SEPARATE
SUCCESS/FAILURE INDICATOR IN THE MESSAGE, THAT INDICATOR HAS TO BE
CHECKED, AND HANDLING OF THE CONDITION IS OUTSIDE THE
CHOREOGRAPHY.

The choreography of the *Negotiation Protocol* is defined by an instance *Document* of the
ebXML *Business Process Specification Schema*[ebBPSS]. The BPSS instance *Document* for

1867 automated negotiation is in Appendix D.

1868

1869   NOTE: Although in [ebCPP], the use of a BPSS instance *Document* to describe
1870   choreography is not required, this specification depends intimately on the selected
1871   choreography description. Therefore, the use of a BPSS instance *Document* is normative
1872   for this version of this specification.

1873

1874

1875 This BPSS instance *Document* defines the negotiation choreography beginning with an exchange
1876 of an offer and response.

1877 • If the response to the offer is "accept offer", the choreography transitions to the final *CPA*
1878   exchange (see below).

1879 • If the response to the offer is "reject offer", the choreography immediately concludes.

1880 • If the response to the offer is "counter offer pending", the choreography then goes into an
1881   alternation of counter offer and response between the two *Parties* which continues until:

1882   ♦ A response of "accept offer" causes the choreography to transition to the final *CPA*
1883   exchange.

1884   ♦ A response of "reject offer" concludes the choreography.

1885

1886 Several *Business Document* names are defined directly under the **ProcessSpecification** element
1887 and referenced in various places as described below.

1888

1889 The BPSS instance *Document* defines initiator and responder *Role* names for each binary
1890 collaboration, collaboration activity, and binary transaction activity. For simplicity in the
1891 explanation below, in most cases, the terms "initiator" and "responder" are used. For each stage
1892 of the choreography, the *NCPA* associates *Role* names with actual *Parties* in the **Action** elements
1893 under the **CollaborationRole** elements.

1894 **13.14.1 Offer-Counter-Offer Choreography**

1895 A counter offer is a requesting *Document* in a new *Business Transaction*, not a response to an
1896 offer. To issue a counter offer, the recipient of an offer SHALL reply "counter pending offer"
1897 and then issue the counter offer as a new *Business Transaction*. This avoids a race condition
1898 with respect to which *Party* sends the next *Message*. It also avoids any need to for the two
1899 *Parties* to switch *Roles*.

1900

1901 ***THE FOLLOWING DISCUSSION HAS TO BE UPDATED TO CORRESPOND TO***
1902 ***CHANGES IN THE BPSS INSTANCE DOCUMENT THAT WERE MADE SINCE THIS***
1903 ***SECTION WAS WRITTEN.***

1904

1905 The choreography begins with the "CPA_Offer_BTA" **BusinessTransactionActivity** element
1906 under the "CPA_Negotiation_BC" **BinaryCollaboration** element. A *CPA* offer *Message* is sent
1907 from the"CPA_Negotiation_Initiator_Role" *Party* to the "CPA_Negotiation_Responder-Role"
1908 *Party* by means of the "CPA_Offer_BT" *Business Transaction*. The "**CPA**_Offer_ReqBA"
1909 **RequestingBusinessActivity** sends the "CPA_Offer_Doc" *Message* from the initiator *Party* to
1910 the responder *Party*. The "CPA_Offer_BT_RespBA" **RespondingBusinessActivity** then sends
1911 the response *Message* from the responder *Party* to the initiator *Party*. This *Message* is then

1912 evaluated as defined by the **Success**, **Failure**, and **Transition** elements under the
1913 "CPA_Negotiation_BC" binary collaboration. These are the elements whose ***fromBusinessState***
1914 attribute has the value "CPA Offer BTA". The value of the **expression** attribute in each of these
1915 elements is the name of the response *Message*, as follows:
1916 • **Success** element: If the response *Message* is "CPA Accept Offer Doc", the proposed *CPA*
1917   has been accepted by the responder *Party* and the choreography transitions to the final *CPA*
1918   exchange.
1919 • **Failure** element: If the response *Message* is "CPA Reject Offer Doc", the proposed *CPA* has
1920   been unconditionally rejected by the responder *Party* and the choreography concludes.
1921 • **Transition** element: If the response *Message* is "CPA Counter Pending Offer Doc", the
1922   responder *Party* will send a counter offer as the next *Business Transaction*. The
1923   ***toBusinessState*** attribute of the **Transition** element identifies "CPA Counter Offer CA" (the
1924   name of the ***CollaborationActivity*** element) as the next state in the choreography.
1925
1926 If the response *Message* to the "CPA_Offer_Doc" *Message* was "CPA Counter Pending Offer
1927 Doc", the transition described above takes place and takes the choreography to state "CPA
1928 Counter Offer CA", i.e. to the ***CollaborationActivity*** element named "CPA Counter Offer CA".
1929 This ***CollaborationActivity*** element references the "CPA_Negotiation_CounterOfferBC"
1930 ***BinaryCollaboration*** element.
1931
1932 The initial request *Message* is under the "CPA_Counter_Offer_1_BTA"
1933 ***BusinessTransactionActivity*** element, which is the "from" state for the following. The *Party*
1934 which received the original *CPA* offer is now the initiator in this ***BusinessTransactionActivity***.
1935 The *Message* is sent from the "CPA_Negotiation_CounterOfferInitiator_Role" *Role* to the
1936 "CPA_Negotiation_CounterOfferResponder_Role" *Role* by means of the
1937 "CPA_Counter_Offer_BT" ***BusinessTransaction***. The *Message*,"CPA_Counter_Offer_Doc", is
1938 identified in the "CPA_Counter_Offer_ReqBA" ***RequestingBusinessActivity***. The response
1939 *Message* is sent by means of the "CPA_Counter_Offer_BT_RespBA"
1940 ***RespondingBusinessActvity***. One of three response *Messages* can be sent, as discussed below.
1941 The response *Message* is then evaluated as defined by the **Success**, **Failure**, and **Transition**
1942 elements under the "CPA_Negotiation_BC" ***BinaryCollaboration***.
1943 • **Success** element: If the response *Message* is "CPA Accept Offer Doc", the proposed *CPA*
1944   has been accepted by the responder *Party* and the choreography transitions to the final *CPA*
1945   exchange.
1946 • **Failure** element: If the response *Message* is "CPA Reject Offer Doc", the proposed *CPA* has
1947   been unconditionally rejected by the responder *Party* and the choreography concludes.
1948 • **Transition** element: If the response *Message* is "CPA Counter Pending Offer Doc", the
1949   responder *Party* will send a counter offer as the next *Business Transaction*. The
1950   ***toBusinessState*** attribute of the **Transition** element identifies the "CPA Counter Offer 2
1951   BTA" *Business Transaction* Activity as the next state in the choreography.
1952
1953 If the above transition takes place, it means that the *Party* that was the responder now becomes
1954 the initiator to supply a counter offer to the counter offer. The "CPA Counter Offer 2 BTA"
1955 ***BusinessTransactionActivity*** is now performed in the same manner as the
1956 "CPA_Counter_Offer_1_BTA" ***BusinessTransactionActivity***, described above.
1957

Automated Negotiation Specification   **Page 54 of 97**

1958   The choreography then iterates between the "CPA_Counter_Offer_1_BTA"
1959   ***BusinessTransactionActivity*** and the "CPA Counter Offer 2 BTA" ***BusinessTransactionActivity***
1960   until a success or failure is achieved. Success causes the choreography to transition to the final
1961   *CPA* exchange. Failure ends the choreography.

1962   **13.14.2 Final CPA exchange**

1963   When either the initial offer or a counter offer is accepted in full, the choreography transitions to
1964   the"*CPA* Final BT" *Business Transaction*. The purpose of this *Business Transaction* is for the
1965   *Party* that accepted the offer or counter offer to send the completed *CPA* to the other *Party*.
1966
1967   If the initial offer was accepted, the next *Business* state is the "*CPA* Final BTA Init Responder"
1968   *Business Transaction* activity, which references the "*CPA* Final BT" *Business Transaction*. The
1969   initiator *Party* for the "Req BA Final_*CPA*_BT_ReqBA" requesting *Business* activity sends the
1970   "*CPA* Final Doc" *Message*, containing the *CPA* or its URL, to the other (responder) *Party*. The
1971   responder *Party* checks the *CPA* and performs the responding *Business* activity conveying one
1972   of:
1973   • The "CPA Final Response Doc" *Message*, to acknowledge receipt of an acceptable *CPA* or
1974   rejection. Acceptance and rejection are indicated by values of the status indicator in the
1975   *Negotiation Message*.
1976   ***THE ABOVE SENTENCE NEEDS TO BE REVISED TO USE THE CORRECT NAME***
1977   ***OF THE STATUS ELEMENT OR ATTRIBUTE WHEN THE MESSAGE SCHEMA IS***
1978   ***COMPLETED.***
1979   • The "CPA Final Response Doc Signed" *Message*, to acknowledge receipt of an acceptable
1980   signed *CPA* and return that *CPA* with the responder *Party's* signature over the initiator
1981   *Party's* signature.
1982
1983   The *Party* that receives the final (double signed) *CPA* SHOULD test it for possible error
1984   conditions as described in Section 13.12. The *Party* that received the double-signed *CPA*
1985   SHALL reply with the AcceptanceAcknowledgment business signal if the *CPA* is acceptable or
1986   with the Exception business signal if the *CPA* is not acceptable. These signals are instance
1987   *Documents* of the business signals defined in [ebBPSS]. Examples of these instance *Documents*
1988   are in Appendix E .
1989
1990   If a counter offer was accepted in full, the choreography transitions to the "*CPA* Final BTA Init
1991   Initiator" *Business Transaction* activity. That *Business Transaction* activity references the "*CPA*
1992   Final BT" *Business Transaction* and proceeds as for acceptance of an initial offer.
1993
1994   ***THE FOLLOWING COMMENT FROM MONICA MARTIN NEEDS TO BE RESOLVED:***
1995   ***WE SHOULD LOOK AT THIS SECTION CLOSELY AS THERE HAS BEEN A GREAT***
1996   ***DEAL OF DISCUSSION ABOUT THE RECEIPTACK AND ACCEPTANCEACK IN***
1997   ***RECENT DAYS FOR UMM, BPSS, AND BCPS. THERE IS NO ACCEPTANCEACK***
1998   ***REQUIRED FROM THE INITIATOR AFTER RECEIPT OF THE RESPONSE FROM***
1999   ***THE RESPONDER. TO EXPLAIN, THE CONTEXT OF A TYPICAL BUSINESS***
2000   ***TRANSACTION IS THAT IF THE RESPONDER SENDS A RESPONSE AND THE***
2001   ***POSSIBILITY EXISTS THAT IT CHANGES THE TERMS AND CONDITIONS FOR THE***
2002   ***INITIATOR. IF THERE IS A RECEIPTACK AND AN ACCEPTANCEACK, THEN THIS***

Negotiation.spec.20Jan03.doc   1/24/2003 6:30 PM

2003 *AFFECTS THE BUSINESS RELATIONSHIP AND HAS LEGAL IMPLICATIONS. IT IS*
2004 *ACCEPTABLE TO SEND A RECEIPTACK FROM THE INITIATOR TO THE*
2005 *RESPONDER TO INDICATE RECEIPT BUT NOT THE ACCEPTANCEACK. SEE BRIAN*
2006 *HAYES ON SOME OF THIS DISCUSSION. IT IS ONLY THE INITIATOR THAT CAN*
2007 *SEND AN ACCEPTANCEACK. AS WE ARE CHANGING ROLES IN THIS*
2008 *'COLLABORATION,' PERHAPS WE SHOULD JUST BE A BIT CLEARER IN THIS*
2009 *SECTION. I'VE ALSO INCLUDED TWO .GIF THAT SHOWS OUR WORKING*
2010 *INITIATING AND RECEIVING STATE MACHINE DIAGRAMS. LATTER IS NOT*
2011 *COMPLETE (FROM BCPS).*

2012 **13.14.3 Negotiation Business Signals**

2013
2014 *ADD DISCUSSION OF THE CONTENTS OF THE BUSINESS SIGNALS.*

2015
2016 *IS IT FEASIBLE TO CONSTRUCT EXAMPLES OF INSTANCE DOCUMENTS OF THE*
2017 *BUSINESS SIGNALS THAT ARE SPECIFIED IN THE NEGOTIATION BPSS INSTANCE*
2018 *DOCUMENT?*

2019 **13.14.4 State Diagrams**

2020 The choreography is illustrated by the state diagram shown in Figure 5 and Figure 6.
2021
2022 *THE STATE DIAGRAMS NEED SOME CORRECTIONS:*
2023 *- IN Figure 5, THE DOCUMENT NAMES IN THE CENTER DO NOT ALWAYS AGREE*
2024 *WITH THE NAMES IN THE INSTANCE DOCUMENT.*
2025 *- IN Figure 6:*
2026 *   + UNLIKE Figure 5, THE DOCUMENT NAMES ARE NOT USED.*
2027 *   THE RETURN OF THE DOUBLE-SIGNED CPA IS NOT SHOWN. PRESUMABLY*
2028 *   + IT IS ANOTHER OUTPUT FROM THE "RECEIVED FINAL CPA" STATE.*
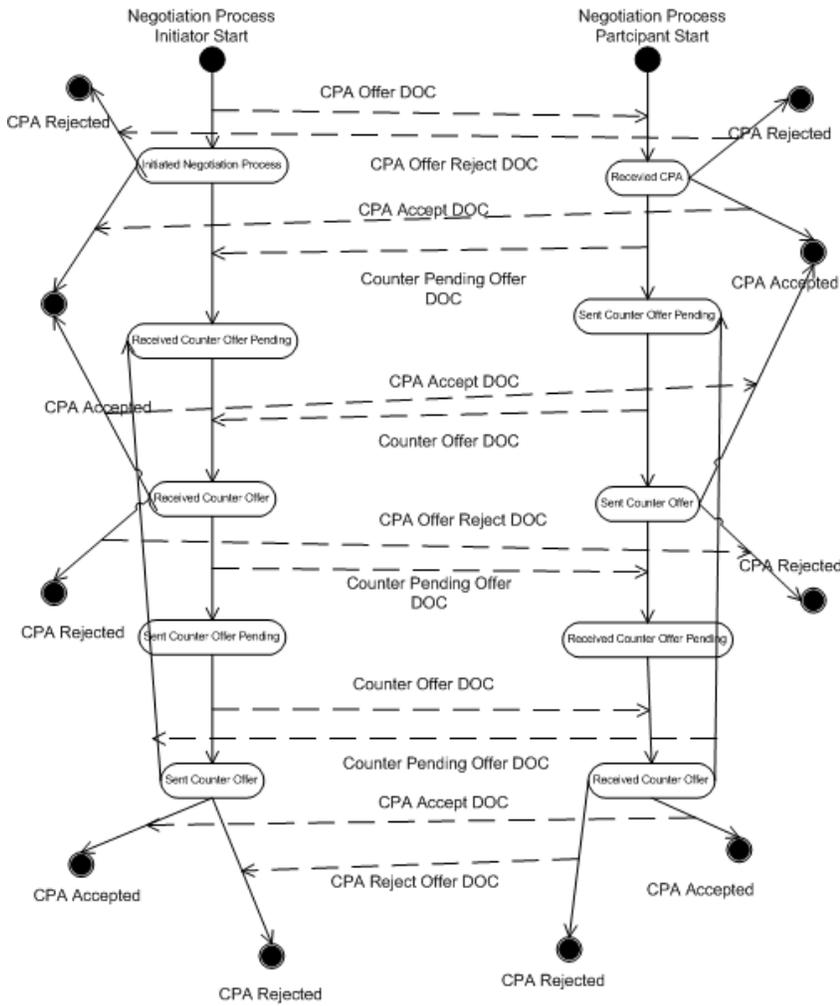
2029

Figure 5, State Diagram for Initial Offer and Counter Offers
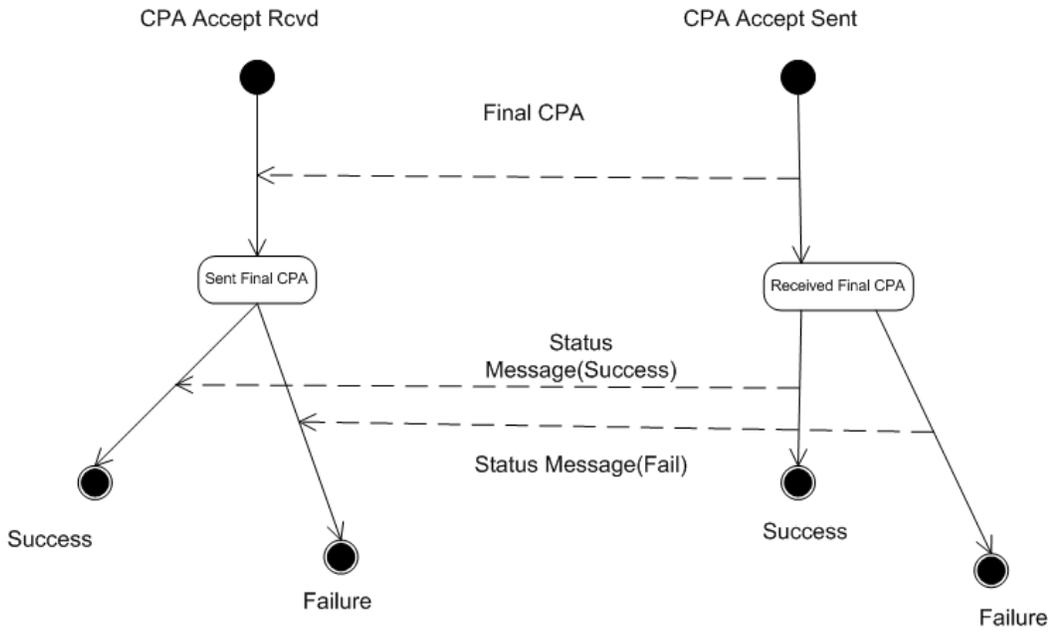
2031

2032 **Figure 6, State Diagram for Final Transaction**

14 Negotiation Algorithm

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

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

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

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

14.1 CPPs and NDDs

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

15 References

VERSION NUMBERS AND URLS TBD.
CHECK THE URLS THAT ARE HERE FOR CORRECTNESS.

[bpPATT] ebXML E-Commerce Patterns, version 1.0,
http://www.ebxml.org/specs/bpPATT.pdf

[ebBPSS] ebXML Business Process Specification Schema

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

[ebMS] ebXML Message Service Specification, version 2.0.

[ebRS] ebXML Registry Services Specification

[RFC2119] Key Words for use in RFCs to indicate Requirement Levels, Internet Engineering
Task Force RFC 2119, http://www.ietf.org/rfc/rfc2119.txt

[RFC2396] Uniform Resource Identifiers URI): General Syntax, Internet Engineering Task
Force RFC 2396, http://www.ietf.org/rfc/rfc2396.txt

[SOAPATTACH] SOAP Messages with Attachments, John J. Barton, Hewlett Packard Labs;
Satish Thatte and Henrik Frystyk Nielsen, Microsoft, Published Oct 09 2000.
http://www.w3.org/TR/2000/NOTE-SOAP-attachments-20001211

[XML] Extensible Markup Language (XML), World Wide Web Consortium,
http://www.w3.org/XML.

[XMLDSIG] XML Signature Syntax and Processing, Worldwide Web Consortium,
http://www.w3.org/TR/xmldsig-core/

[XMLENC] XML Encryption Syntax and Processing, Worldwide Web Consortium,
http://www.w3.org/TR/2002/CR-xmlenc-core-20020304/

[XMLNS] Namespaces in XML, Worldwide Web Consortium,
 http://www.w3.org/TR/REC-xml-names/

[XPATH] XML Path Language (XPath) Version 1.0,
http://www.w3.org/TR/xpath

2107 **16 Conformance**

2108 In order to conform to this specification, an implementation:
2109   a) SHALL support all the functional and interface requirements defined in this specification,
2110   b) SHALL NOT specify any requirements that would contradict or cause non-conformance
2111   to this specification.
2112
2113 A conforming implementation SHALL satisfy the conformance requirements of the applicable
2114 parts of this specification.
2115
2116 The objective of conformance testing is to determine whether an implementation being tested
2117 conforms to the requirements stated in this specification. Conformance testing enables vendors to
2118 implement compatible and interoperable systems. Implementations and applications SHALL be
2119 tested using available test suites to verify their conformance to this specification.
2120
2121 Publicly available test suites from vendor neutral organizations such as OASIS and the U.S.A.
2122 National Institute of Science and Technology (NIST) SHOULD be used to verify the
2123 conformance of implementations, applications, and components claiming conformance to this
2124 specification. Open-source reference implementations might be available to allow vendors to test
2125 their products for interface compatibility, conformance, and interoperability.

2126 **16.1 NDD and Negotiation Messages**

2127 An implementation of a tool or service that creates or maintains ebXML instance *Documents* of
2128 the *NDD* and *Negotiation Messages* SHALL be determined to be conformant by validation of the
2129 instance *Documents*, created or modified by said tool or service, against the XML
2130 Schema[XMLSCHEMA-1] definition of these *Documents* in Appendix A and Appendix B,
2131 respectively, and available from
2132
2133 *URLS TO BE SUPPLIED*
2134
2135 by using two or more validating XML Schema parsers that conform to the W3C XML Schema
2136 specifications[XMLSCHEMA-1, XMLSCHEMA-2].

2137 **16.2 NCPA Instance Document**

2138 An implementation of a tool or service that creates or maintains *NCPA* instance *Documents*
2139 SHALL be determined to be conformant by validation of the *NCPA* instance *Documents*, created
2140 or modified by said tool or service, against the XML Schema[XMLSCHEMA-1] definition of
2141 the *CPA* in [ebCPP]and available from
2142
2143   http://www.oasis-open.org/committees/ebxml-cppa/schema/cpp-cpa-2_0.xsd
2144
2145 by using two or more validating XML Schema parsers that conform to the W3C XML Schema
2146 specifications[XMLSCHEMA-1, XMLSCHEMA-2].

2147 **16.3 Negotiation BPSS Instance Document**

2148 An implementation of a tool or service that creates or maintains negotiation BPSS instance

2149 *Documents* SHALL be determined to be conformant by validation of the BPSS instance
2150 *Documents*, created or modified by said tool or service, against the XML
2151 Schema[XMLSCHEMA-1] definition of the BPSS in available from

2152
2153 ***URL TO BE SUPPLIED.***
2154
2155 by using two or more validating XML Schema parsers that conform to the W3C XML Schema
2156 specifications[XMLSCHEMA-1, XMLSCHEMA-2].

16.4 Negotiation Business Signals

2158 An implementation of a tool or service that creates or maintains negotiation *Business*-signal
2159 instance *Documents* SHALL be determined to be conformant by validation of the *Business*-
2160 signal instance *Documents*, created or modified by said tool or service, against the XML
2161 Schema[XMLSCHEMA-1] definition of the *Business* signals and available from

2162
2163 ***URL TO BE SUPPLIED.***
2164
2165 by using two or more validating XML Schema parsers that conform to the W3C XML Schema
2166 specifications[XMLSCHEMA-1, XMLSCHEMA-2].

2167 # 17 Disclaimer

2168 The views and specification expressed in this document are those of the authors and are not
2169 necessarily those of their employers. The authors and their employers specifically disclaim
2170 responsibility for any problems arising from correct or incorrect implementation or use of this
2171 design.

18 Contact Information

OTHERS TBD

Martin W. Sachs (sub team lead)
IBM T. J. Watson Research Center
P.O.B. 704
Yorktown Hts, NY 10598
USA
Phone: 914-784-7287
email: mailto:mwsachs@us.ibm.com

Dale W. Moberg (author)
Cyclone Commerce
8388 E. Hartford Drive
Scottsdale, AZ 85255
USA
Phone: 480-627-2648
email: mailto:dmoberg@cyclonecommerce.com

2190   **Notices**

2191   *NEED TO DETERMINE OF UN/CEFACT HAS TO BE MENTIONED.*

2192
2193   Portions of this document are copyright © 2002 OASIS.

2194

2195
2196
2197   OASIS takes no position regarding the validity or scope of any intellectual property or other
2198   rights that might be claimed to pertain to the implementation or use of the technology described
2199   in this document or the extent to which any license under such rights might or might not be
2200   available; neither does it represent that it has made any effort to identify any such rights.
2201   Information on OASIS's procedures with respect to rights in OASIS specifications can be found
2202   at the OASIS website. Copies of claims of rights made available for publication and any
2203   assurances of licenses to be made available, or the result of an attempt made to obtain a general
2204   license or permission for the use of such proprietary rights by implementers or users of this
2205   specification, can be obtained from the OASIS Executive Director.

2206
2207   OASIS invites any interested party to bring to its attention any copyrights, patents or patent
2208   applications, or other proprietary rights which may cover technology that may be required to
2209   implement this specification. Please address the information to the OASIS Executive Director.

2210
2211   OASIS has been notified of intellectual property rights claimed in regard to some or all of the
2212   contents of this specification. For more information consult the online list of claimed rights.

Appendix A XML Schema for Negotiation Descriptor Document (Normative)

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

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

```xml
<?xml version="1.0" encoding="UTF-8"?>
<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"/>
                                <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="EnumeratedValues">
        <sequence>
            <element name="PresentOrNot" type="tns:PresentOrNotType" minOccurs="0"/>
            <element name="Value" type="xs:string" maxOccurs="unbounded"/>
        </sequence>
    </complexType>
    <!--For also stating that the enumerted values have some order associated with them-->
    <complexType name="OrderedEnumeratedValues">
        <complexContent>
            <extension base="tns:EnumeratedValues">
                <attribute name="preference" use="optional">
```

```
2270                    <simpleType name="orderName">
2271                        <restriction base="xs:string">
2272                            <enumeration value="EarlierPreferred"/>
2273                            <enumeration value="LaterPreferred"/>
2274                        </restriction>
2275                    </simpleType>
2276                </attribute>
2277            </extension>
2278        </complexContent>
2279    </complexType>
2280    <!--For giving the endpoints for elements like Start/End. The type is currently set to string beacuse XML spy does
2281 not seem to validate dateTime entries correctly, but should be changed to dateTime later -->
2282    <complexType name="DateEndPointsType">
2283        <sequence>
2284            <element name="EarliestStart" type="xs:string"/>
2285            <element name="LatestEnd" type="xs:string"/>
2286            <!--TODO: Make this xs:dateTime -->
2287        </sequence>
2288    </complexType>
2289    <!--For giving a type of preference function. Currently, the preference function can be one of two: (a) a piecewise
2290 linear function,  (b) an arbtrary function expressed as a string (such as x*x+ y*y) .-->
2291    <complexType name="PreferenceFunctionType">
2292        <sequence>
2293            <choice>
2294                <element name="PiecewiseLinearPiece" maxOccurs="unbounded">
2295                    <complexType>
2296                        <sequence>
2297                            <element name="x1" type="xs:string"/>
2298                            <element name="y1" type="xs:NMTOKEN"/>
2299                            <element name="x1" type="xs:string"/>
2300                            <element name="y1" type="xs:NMTOKEN"/>
2301                            <!--TODO: Make dateTime/NMTOKEN for generality?-->
2302                        </sequence>
2303                    </complexType>
2304                </element>
2305                <element name="FunctionDefinedByEquation" type="xs:string"/>
2306                <!--<xs:element name="NoPreference"/> -->
2307            </choice>
2308        </sequence>
2309    </complexType>
2310    <!--For giving a value, and associating with it some preferrence function -->
2311    <complexType name="ValueWithPreferenceMeasureType">
2312        <sequence>
2313            <element name="EndPoints" type="tns:DateEndPointsType"/>
2314            <element name="PreferenceFunction" type="tns:PreferenceFunctionType" minOccurs="0"/>
2315        </sequence>
2316    </complexType>
2317    <!--For specifying a preference for whether an entry must be present or not -->
2318    <complexType name="PresentOrNotType">
2319        <attribute name="value" use="required">
2320            <simpleType>
2321                <restriction base="xs:string">
2322                    <enumeration value="MustBePresent"/>
2323                    <enumeration value="MustBeAbsent"/>
2324                    <enumeration value="PreferredPresent"/>
2325                    <enumeration value="PreferredAbsent"/>
2326                    <enumeration value="Agnostic"/>
2327                </restriction>
2328            </simpleType>
2329        </attribute>
2330    </complexType>
2331    <!--For giving the integer endpoints for elements like retries.  -->
```

Automated Negotiation Specification   **Page 67 of 97**

```
2332        <complexType name="IntegerEndPointsType">
2333            <sequence>
2334                <element name="SmallestValue" type="xs:integer"/>
2335                <element name="LatestValue" type="xs:integer"/>
2336            </sequence>
2337        </complexType>
2338        <!--This associates a preference order to the integer end points or a preference such as Smaller is Preferred-->
2339        <complexType name="IntegerValuesWithPreferenceMeasureType">
2340            <sequence>
2341                <element name="EndPoints" type="tns:IntegerEndPointsType"/>
2342                <element name="PreferenceFunction" type="tns:PreferenceFunctionType" minOccurs="0"/>
2343            </sequence>
2344            <attribute name="preferenceOrder">
2345                <simpleType name="orderName">
2346                    <restriction base="xs:string">
2347                        <enumeration value="SmallerPreferred"/>
2348                        <enumeration value="LargerPreferred"/>
2349                    </restriction>
2350                </simpleType>
2351            </attribute>
2352        </complexType>
2353        <!--This type is for integer values whose (a) presence can be potentially negotiated (b) the values themselves can
2354    be negotiated -->
2355        <complexType name="IntegerValuesType">
2356            <sequence>
2357                <element name="PresentOrNot" type="tns:PresentOrNotType" minOccurs="0"/>
2358                <element name="RangeInfo" type="tns:IntegerValuesWithPreferenceMeasureType"/>
2359            </sequence>
2360        </complexType>
2361        <!--For specifying a preference for whether an entry must be present or not -->
2362        <complexType name="BooleanValuesType">
2363            <sequence>
2364                <element name="PresentOrNot" type="tns:PresentOrNotType" minOccurs="0"/>
2365            </sequence>
2366            <attribute name="preference" use="required">
2367                <simpleType>
2368                    <restriction base="xs:string">
2369                        <enumeration value="TruePreferred"/>
2370                        <enumeration value="FalsePreferred"/>
2371                        <enumeration value="Agnostic"/>
2372                    </restriction>
2373                </simpleType>
2374            </attribute>
2375        </complexType>
2376        <!--For specifying the minimum and maximum allowable durations-->
2377        <complexType name="DurationType">
2378            <sequence>
2379                <element name="PresentOrNot" type="tns:PresentOrNotType" minOccurs="0"/>
2380                <element name="MinimumDuration" type="xs:string"/>
2381                <element name="MaximumDuration" type="xs:string"/>
2382                <!--TODO: Make this xs:duration -->
2383            </sequence>
2384        </complexType>
2385        <!--For also stating that the durations have some preference  associated with them-->
2386        <complexType name="DurationWithPreferenceType">
2387            <complexContent>
2388                <extension base="tns:DurationType">
2389                    <attribute name="preferenceOrder" use="optional">
2390                        <simpleType name="preferenceName">
2391                            <restriction base="xs:string">
2392                                <enumeration value="SmallerPreferred"/>
2393                                <enumeration value="LargerPreferred"/>
```

```
2394                    </restriction>
2395                  </simpleType>
2396                </attribute>
2397              </extension>
2398           </complexContent>
2399        </complexType>
2400    </schema>
```

Negotiation.spec.20Jan03.doc   1/24/2003 6:30 PM

2401 Appendix B XML Schema for Negotiation Messages
2402   (Normative)

2403 The XML Schema for the negotiation *Messages* is available in text form at:

2404

```
2405 <?xml version="1.0" encoding="UTF-8"?>
2406
2407 <!--  This is the schema that corresponds to the version 1.0 CPP/A Negotiation spec
2408   -->
2409 <schema targetNamespace="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpa-negot-1_0.xsd"
2410   xmlns="http://www.w3.org/2001/XMLSchema"
2411   xmlns:cppa="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpp-cpa-2_0.xsd"
2412 xmlns:tns="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpa-negot-1_0.xsd">
2413   <import namespace="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpp-cpa-2_0.xsd"
2414 schemaLocation="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpp-cpa-2_0.xsd"/>
2415   <attributeGroup name="id.grp">
2416     <attribute name="id" type="cppa:non-empty-string" use="required"/>
2417     <attribute ref="cppa:version" use="required"/>
2418   </attributeGroup>
2419   <element name="NegotiationMessage">
2420     <complexType>
2421       <sequence>
2422         <element ref="tns:NCPA"/>
2423         <element ref="tns:CPATemplateId"/>
2424         <element ref="cppa:SecurityDetails"/>
2425         <element ref="tns:InitiatingParty"/>
2426         <element ref="tns:RespondingParty"/>
2427         <element ref="tns:BPSSBusinessDocumentName"/>
2428         <element name="ExpirationDate" type="dateTime"/>
2429         <element ref="tns:BusinessDocuments"/>
2430         <element ref="tns:NegotiationContent"/>
2431         <element maxOccurs="1" minOccurs="0" name="ResponseToURL" type="anyURI"/>
2432         <element maxOccurs="1" minOccurs="0" name="Comment" type="string"/>
2433       </sequence>
2434       <attribute name="businessMsgId" type="ID" use="required"/>
2435       <attribute name="negotiationDialogId" type="cppa:non-empty-string" use="required"/>
2436       <attribute name="offerId" type="cppa:non-empty-string"/>
2437       <attribute name="inresponseTo" type="cppa:non-empty-string"/>
2438       <attribute name="binding" type="boolean" use="required"/>
2439       <attribute name="messageType" type="tns:messageTypeValue.type" use="required"/>
2440       <attribute name="error" type="tns:errorValue.type"/>
2441     </complexType>
2442   </element>
2443   <element name="CPATemplateId">
2444     <complexType>
2445       <attributeGroup ref="tns:id.grp"/>
2446     </complexType>
2447   </element>
2448   <element name="NCPA">
2449     <complexType>
2450       <attribute name="uri" type="anyURI" use="required"/>
2451     </complexType>
2452   </element>
2453   <element name="BPSSBusinessDocumentName">
2454     <complexType>
2455       <attribute name="name" type="tns:bpssBusinessDocumentName.type" use="required"/>
2456     </complexType>
2457   </element>
```

```
2458        <element name="BusinessDocuments">
2459          <complexType>
2460            <choice>
2461              <element name="CPATemplateDoc" type="tns:CPATemplateDoc.type"/>
2462            </choice>
2463          </complexType>
2464        </element>
2465        <element name="NegotiationContent">
2466          <complexType>
2467            <sequence>
2468              <element maxOccurs="unbounded" minOccurs="1" ref="tns:AcceptedItem"/>
2469              <element maxOccurs="unbounded" minOccurs="0" ref="tns:DeletedItem"/>
2470              <element maxOccurs="unbounded" minOccurs="0" ref="tns:UpdatedItem"/>
2471              <element maxOccurs="unbounded" minOccurs="0" ref="tns:InsertedItem"/>
2472            </sequence>
2473          </complexType>
2474        </element>
2475        <element name="AcceptedItem" type="tns:simpleItem.type"/>
2476        <element name="DeletedItem" type="tns:simpleItem.type"/>
2477        <element name="UpdatedItem" type="tns:updatedItem.type"/>
2478        <element name="InsertedItem" type="tns:insertedItem.type"/>
2479        <element name="InitiatingParty" type="tns:partySummary.type"/>
2480        <element name="RespondingParty" type="tns:partySummary.type"/>
2481        <complexType name="simpleItem.type">
2482          <attribute name="xpath" type="tns:xpath.type" use="required"/>
2483        </complexType>
2484        <complexType name="updatedItem.type">
2485          <attribute name="xpath" type="tns:xpath.type" use="required"/>
2486          <attribute name="originalValue" type="cppa:non-empty-string" use="required"/>
2487          <attribute name="proposedValue" type="cppa:non-empty-string" use="required"/>
2488        </complexType>
2489        <complexType name="insertedItem.type">
2490          <attribute name="xpath" type="tns:xpath.type" use="required"/>
2491          <attribute name="proposedValue" type="cppa:non-empty-string" use="required"/>
2492        </complexType>
2493        <complexType name="doc.type">
2494          <choice>
2495            <element name="BinaryDoc" type="base64Binary"/>
2496            <element name="Uri" type="anyURI"/>
2497          </choice>
2498        </complexType>
2499        <complexType name="CPATemplateDoc.type">
2500          <sequence>
2501            <element name="NDD" type="tns:doc.type"/>
2502            <element name="CPATemplate" type="tns:doc.type"/>
2503          </sequence>
2504        </complexType>
2505        <complexType name="partySummary.type">
2506          <sequence>
2507            <element ref="cppa:PartyId"/>
2508            <element name="CPPId">
2509              <complexType>
2510                <attributeGroup ref="tns:id.grp"/>
2511              </complexType>
2512            </element>
2513            <element maxOccurs="1" minOccurs="0" name="CPPNDD" type="tns:doc.type"/>
2514          </sequence>
2515        </complexType>
2516        <simpleType name="xpath.type">
2517          <restriction base="string"/>
2518        </simpleType>
2519        <simpleType name="messageTypeValue.type">
```

Automated Negotiation Specification   **Page 71 of 97**

```
2520        <restriction base="NMTOKEN">
2521            <enumeration value="Offer"/>
2522            <enumeration value="CounterOffer"/>
2523            <enumeration value="CounterOfferPending"/>
2524            <enumeration value="Rejected"/>
2525            <enumeration value="Accepted"/>
2526            <enumeration value="Expired"/>
2527            <enumeration value="SinglePartySigned"/>
2528            <enumeration value="Signed"/>
2529            <enumeration value="Unsigned"/>
2530        </restriction>
2531      </simpleType>
2532      <simpleType name="errorValue.type">
2533        <restriction base="NMTOKEN">
2534            <enumeration value="ExpiredCPP"/>
2535            <enumeration value="UnableToFullfillSecurityRequirements"/>
2536            <enumeration value="ProposedSecurityPolicyInadequate"/>
2537            <enumeration value="OutOfSequenceCounterOffer"/>
2538            <enumeration value="FaildSignatureValidation_CPATemplate"/>
2539            <enumeration value="FailedSignatureValidation_CPA"/>
2540            <enumeration value="UnsupportedBusinessRelationship"/>
2541            <enumeration value="UnsupportedPackaging"/>
2542            <enumeration value="UnsupportedSignal"/>
2543            <enumeration value="FailedToConverge"/>
2544            <enumeration value="PreviouslyRejectedCPA"/>
2545            <enumeration value="ExpiredOffer"/>
2546            <enumeration value="FormatError"/>
2547            <enumeration value="UnknownSystemError"/>
2548        </restriction>
2549      </simpleType>
2550      <simpleType name="bpssBusinessDocumentName.type">
2551        <restriction base="NMTOKEN">
2552            <enumeration value="CPA_Offer_Doc"/>
2553            <enumeration value="CPA_Accept_Offer_Doc"/>
2554            <enumeration value="CPA_Counter_Pending_Offer_Doc"/>
2555            <enumeration value="CPA_Counter_Offer_Doc"/>
2556            <enumeration value="CPA_Reject_Offer_Doc"/>
2557            <enumeration value="CPA_Final_Doc"/>
2558            <enumeration value="CPA_Final_Response_Doc"/>
2559
2560            <enumeration value="CPA_Final_Response_Doc_Signed"/>
2561        </restriction>
2562      </simpleType>
2563 </schema>
2564
```

Appendix C Negotiation CPA Example (Non-Normative)

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

THE NCPA'S PACKAGING DEFINITIONS HAVE TO BE COMPLETED AFTER THE NDD AND MESSAGE SCHEMA ARE COMPLETED.

```xml
<?xml version="1.0"?>
<!-- edited with XML Spy v4.4 U (http://www.xmlspy.com) by Hima Mukkamala (Web Services Architecture WG) -->
<tp:CollaborationProtocolAgreement xmlns:tp="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpp-cpa-
2_0.xsd" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:ds="http://www.w3.org/2000/09/xmldsig#" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xsi:schemaLocation="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpp-cpa-2_0.xsd
                 cpp-cpa-2_0.xsd" tp:cpaid="uri:NegoInit-and-NegoResp-cpa" tp:version="2_0a">
    <tp:Status tp:value="proposed"/>
    <tp:Start>2001-05-20T07:21:00Z</tp:Start>
    <tp:End>2003-05-20T07:21:00Z</tp:End>
    <tp:ConversationConstraints tp:invocationLimit="100" tp:concurrentConversations="10"/>
    <!-- Party info for Negotiation Initiator -->
    <tp:PartyInfo tp:partyName="NegotiationInitiator" tp:defaultMshChannelId="asyncChannelA1"
tp:defaultMshPackageId="NegoInit_MshSignalPackage">
        <tp:PartyId tp:type="urn:oasis:names:tc:ebxml-cppa:partyid-type:duns">123456789</tp:PartyId>
        <tp:PartyRef xlink:href="http://NegoInit.com/about.html"/>
        <!-- This role is for Negotiation Initiator performing the role of Negotiation Initiator -->
        <tp:CollaborationRole>
            <tp:ProcessSpecification tp:version="2.0" tp:name="CPPA-Negotiation" xlink:type="simple"
xlink:href="http://www.oasis-open.org/committees/ebxml-cppa-negot/CPA_Negotiation_BPSS.xml"
tp:uuid="bpid:ebXML:CPPA-Negotiation"/>
            <tp:Role tp:name="CPA Negotiation Initiator" xlink:type="simple" xlink:href="http://www.oasis-
open.org/committees/ebxml-cppa-negot/CPA_Negotiation_BPSS.xm#CPA Negotiation Initiator"/>
            <tp:ServiceBinding>
                <tp:Service>bpid:ebXML:CPPA-Negotiation</tp:Service>
                <!-- This send is for sending the Negotiation Offer -->
                <tp:CanSend>
                    <tp:ThisPartyActionBinding tp:id="NegoInit_ABID1" tp:action="CPA_Offer_BT_ReqBA"
tp:packageId="NegoInit_OfferRequestPackage">
                        <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
                        <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
tp:businessTransactionActivity="CPA Offer BTA" tp:requestOrResponseAction="CPA_Offer_BT_ReqBA"/>
                        <tp:ChannelId>asyncChannelA1</tp:ChannelId>
                    </tp:ThisPartyActionBinding>
                    <tp:OtherPartyActionBinding>NegoResp_ABID1</tp:OtherPartyActionBinding>
                </tp:CanSend>
                <!-- This send is for sending the Receipt Acknowledgment -->
                <tp:CanSend>
                    <tp:ThisPartyActionBinding tp:id="NegoInit_ABID2" tp:action="ReceiptAcknowledgement"
tp:packageId="NegoInit_ReceiptAcknowledgmentPackage">
                        <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="true"
tp:isNonRepudiationReceiptRequired="true" tp:isConfidential="transient" tp:isAuthenticated="persistent"
tp:isTamperProof="persistent" tp:isAuthorizationRequired="true"/>
                        <tp:ChannelId>asyncChannelA1</tp:ChannelId>
                    </tp:ThisPartyActionBinding>
                    <tp:OtherPartyActionBinding>NegoResp_ABID2</tp:OtherPartyActionBinding>
                </tp:CanSend>
                <!-- This send is for send the Final message in the collaboration. This would be the double signed CPA
document or acceptance or reject of the CPA  in the final Response document-->
                <tp:CanSend>
                    <tp:ThisPartyActionBinding tp:id="NegoInit_FinalResponseMessageA"
tp:action="Final_CPA_BT_RespBA" tp:packageId="NegoInit_FinalMessage">
```

Automated Negotiation Specification   **Page 73 of 97**

```
2625                    <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
2626   tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2627   tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2628                    <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2629   tp:businessTransactionActivity="CPA Final BTA" tp:requestOrResponseAction="Final_CPA_BT_RespBA"/>
2630                    <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2631                  </tp:ThisPartyActionBinding>
2632                  <tp:OtherPartyActionBinding>NegoResp_FinalResponseMessageB</tp:OtherPartyActionBinding>
2633                </tp:CanSend>
2634                <!-- This receive is for receiving the response for Negotiation Offer, could be acceptance, reject or counter
2635   offer-->
2636                <tp:CanReceive>
2637                  <tp:ThisPartyActionBinding tp:id="NegoInit_ABID9" tp:action="CPA_Offer_BT_RespBA"
2638   tp:packageId="NegoInit_OfferResponsePackage">
2639                    <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
2640   tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2641   tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2642                    <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2643   tp:businessTransactionActivity="CPA Offer BTA" tp:requestOrResponseAction="CPA_Offer_BT_RespBA"/>
2644                    <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2645                  </tp:ThisPartyActionBinding>
2646                  <tp:OtherPartyActionBinding>NegoResp_ABID9</tp:OtherPartyActionBinding>
2647                </tp:CanReceive>
2648                <!-- This receive is for receiving the Final Response document in the final BTA -->
2649                <tp:CanReceive>
2650                  <tp:ThisPartyActionBinding tp:id="NegoInit_FinalResponseA" tp:action="Final_CPA_BT_ReqBA"
2651   tp:packageId="NegoInit_FinalMessage">
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 Final BTA" tp:requestOrResponseAction="Final_CPA_BT_ReqBA"/>
2657                    <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2658                  </tp:ThisPartyActionBinding>
2659                  <tp:OtherPartyActionBinding>NegoResp_FinalResponseB</tp:OtherPartyActionBinding>
2660                </tp:CanReceive>
2661                <!-- This Receive is for receiving the Receipt Acknowledgment -->
2662                <tp:CanReceive>
2663                  <tp:ThisPartyActionBinding tp:id="NegoInit_ABID3" tp:action="ReceiptAcknowledgment"
2664   tp:packageId="NegoInit_ReceiptAcknowledgmentPackage">
2665                    <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="true"
2666   tp:isNonRepudiationReceiptRequired="true" tp:isConfidential="transient" tp:isAuthenticated="persistent"
2667   tp:isTamperProof="persistent" tp:isAuthorizationRequired="true"/>
2668                    <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2669                  </tp:ThisPartyActionBinding>
2670                  <tp:OtherPartyActionBinding>NegoResp_ABID3</tp:OtherPartyActionBinding>
2671                </tp:CanReceive>
2672                <!-- This Receive is for receiving the Exception -->
2673                <tp:CanReceive>
2674                  <tp:ThisPartyActionBinding tp:id="NegoInit_ABID4" tp:action="Exception"
2675   tp:packageId="NegoInit_ExceptionPackage">
2676                    <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="true"
2677   tp:isNonRepudiationReceiptRequired="true" tp:isConfidential="transient" tp:isAuthenticated="persistent"
2678   tp:isTamperProof="persistent" tp:isAuthorizationRequired="true"/>
2679                    <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2680                  </tp:ThisPartyActionBinding>
2681                  <tp:OtherPartyActionBinding>NegoResp_ABID4</tp:OtherPartyActionBinding>
2682                </tp:CanReceive>
2683              </tp:ServiceBinding>
2684            </tp:CollaborationRole>
2685            <!-- This role is for Negotiation Initiator company performing the role of Negotiation Counter offer responder -->
2686            <tp:CollaborationRole>
2687              <tp:ProcessSpecification tp:version="2.0" tp:name="CPPA-Negotiation" xlink:type="simple"
2688   xlink:href="http://www.oasis-open.org/committees/ebxml-cppa-negot/CPA_Negotiation_BPSS.xml"
2689   tp:uuid="bpid:ebXML:CPPA-Negotiation"/>
2690              <tp:Role tp:name="CPA Negotiation Counter Offer Responder" xlink:type="simple" xlink:href="http://www.oasis-
```

Copyright © OASIS, 2003. All Rights Reserved

```
2691    open.org/committees/ebxml-cppa-negot/CPA_Negotiation_BPSS.xm#CPA Negotiation Counter Offer Responder"/>
2692            <tp:ServiceBinding>
2693                <tp:Service>bpid:ebXML:CPPA-Negotiation</tp:Service>
2694                <!-- This send is for sending the Negotiation Counter Offer in "CPA Counter Offer 2 BTA"-->
2695                <tp:CanSend>
2696                    <tp:ThisPartyActionBinding tp:id="NegoInit_ABID5" tp:action="CPA_Counter_Offer_BT_ReqBA"
2697    tp:packageId="NegoInit_CounterOfferRequestPackage">
2698                        <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
2699    tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2700    tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2701                        <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2702    tp:businessTransactionActivity="CPA Counter Offer 2 BTA"
2703    tp:requestOrResponseAction="CPA_Counter_Offer_BT_ReqBA">
2704                            <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
2705                        </tp:ActionContext>
2706                        <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2707                    </tp:ThisPartyActionBinding>
2708                    <tp:OtherPartyActionBinding>NegoResp_ABID5</tp:OtherPartyActionBinding>
2709                </tp:CanSend>
2710                <!-- This send is for sending the Negotiation Counter Offer Response  in "CPA Counter Offer 1 BTA"-->
2711                <tp:CanSend>
2712                    <tp:ThisPartyActionBinding tp:id="NegoInit_ABID6" tp:action="CPA_Counter_Offer_BT_ReqBA"
2713    tp:packageId="NegoInit_CounterOfferResponsePackage">
2714                        <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
2715    tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2716    tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2717                        <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2718    tp:businessTransactionActivity="CPA Counter Offer 1 BTA"
2719    tp:requestOrResponseAction="CPA_Counter_Offer_BT_RespBA">
2720                            <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
2721                        </tp:ActionContext>
2722                        <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2723                    </tp:ThisPartyActionBinding>
2724                    <tp:OtherPartyActionBinding>NegoResp_ABID6</tp:OtherPartyActionBinding>
2725                </tp:CanSend>
2726                <!-- This send is for sending the Final CPA Response message in CPA_Final_BTA_init_Responder"-->
2727                <tp:CanSend>
2728                    <tp:ThisPartyActionBinding tp:id="NegoCOR_FinalMessageA" tp:action="Final_CPA_BT_ReqBA"
2729    tp:packageId="NegoInit_FinalMessage">
2730                        <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
2731    tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2732    tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2733                        <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2734    tp:businessTransactionActivity="CPA_Final_BTA_init_Responder" tp:requestOrResponseAction="Final_CPA_BT_ReqBA">
2735                            <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
2736                        </tp:ActionContext>
2737                        <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2738                    </tp:ThisPartyActionBinding>
2739                    <tp:OtherPartyActionBinding>NegoCOResp_FinalMessageB</tp:OtherPartyActionBinding>
2740                </tp:CanSend>
2741                <!-- This send is for sending the response to the Final CPA Response message in
2742    CPA_Final_BTA_init_Initiator"-->
2743                <tp:CanSend>
2744                    <tp:ThisPartyActionBinding tp:id="NegoCOR_FinalMessageResponseA"
2745    tp:action="Final_CPA_BT_RespBA" tp:packageId="NegoInit_FinalMessage">
2746                        <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
2747    tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2748    tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2749                        <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2750    tp:businessTransactionActivity="CPA_Final_BTA_init_Initiator" tp:requestOrResponseAction="Final_CPA_BT_RespBA">
2751                            <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
2752                        </tp:ActionContext>
2753                        <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2754                    </tp:ThisPartyActionBinding>
2755                    <tp:OtherPartyActionBinding>NegoCOResp_FinalMessageResponseB</tp:OtherPartyActionBinding>
2756                </tp:CanSend>
```

Automated Negotiation Specification   **Page 75 of 97**

```
2757                        <!-- This send is for sending the Receipt Acknowledgment -->
2758                        <tp:CanSend>
2759                            <tp:ThisPartyActionBinding tp:id="NegoInit_ABID13" tp:action="ReceiptAcknowledgement"
2760        tp:packageId="NegoInit_ReceiptAcknowledgmentPackage">
2761                                <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="true"
2762        tp:isNonRepudiationReceiptRequired="true" tp:isConfidential="transient" tp:isAuthenticated="persistent"
2763        tp:isTamperProof="persistent" tp:isAuthorizationRequired="true"/>
2764                                <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2765                            </tp:ThisPartyActionBinding>
2766                            <tp:OtherPartyActionBinding>NegoResp_ABID13</tp:OtherPartyActionBinding>
2767                        </tp:CanSend>
2768                        <!-- This receive is for receiving the Final CPA message in CPA_Final_BTA_init_Initiator"-->
2769                        <tp:CanReceive>
2770                            <tp:ThisPartyActionBinding tp:id="NegoCOR_FinalMessageA1" tp:action="Final_CPA_BT_ReqBA"
2771        tp:packageId="NegoInit_FinalMessage">
2772                                <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
2773        tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2774        tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2775                                <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2776        tp:businessTransactionActivity="CPA_Final_BTA_init_Initiator" tp:requestOrResponseAction="Final_CPA_BT_ReqBA">
2777                                    <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
2778                                </tp:ActionContext>
2779                                <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2780                            </tp:ThisPartyActionBinding>
2781                            <tp:OtherPartyActionBinding>NegoCOResp_FinalMessageB1</tp:OtherPartyActionBinding>
2782                        </tp:CanReceive>
2783                        <!-- This receive is for receiving the response to the Final CPA message in
2784        CPA_Final_BTA_init_Responder"-->
2785                        <tp:CanReceive>
2786                            <tp:ThisPartyActionBinding tp:id="NegoCOR_FinalMessageResponseA2"
2787        tp:action="Final_CPA_BT_RespBA" tp:packageId="NegoInit_FinalMessage">
2788                                <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
2789        tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2790        tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2791                                <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2792        tp:businessTransactionActivity="CPA_Final_BTA_init_Responder"
2793        tp:requestOrResponseAction="Final_CPA_BT_RespBA">
2794                                    <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
2795                                </tp:ActionContext>
2796                                <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2797                            </tp:ThisPartyActionBinding>
2798                            <tp:OtherPartyActionBinding>NegoCOResp_FinalMessageResponseB2</tp:OtherPartyActionBinding>
2799                        </tp:CanReceive>
2800                        <!-- This receive is for receiving the response forNegotiation Counter Offer, could be accept, reject or again
2801        send a counter offer This happens in "CPA Counter Offer 2 BTA"-->
2802                        <tp:CanReceive>
2803                            <tp:ThisPartyActionBinding tp:id="NegoInit_ABID10" tp:action="CPA_Counter_Offer_BT_RespBA"
2804        tp:packageId="NegoInit_CounterOfferResponsePackage">
2805                                <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
2806        tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2807        tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2808                                <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2809        tp:businessTransactionActivity="CPA Counter Offer 2 BTA"
2810        tp:requestOrResponseAction="CPA_Counter_Offer_BT_RespBA">
2811                                    <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
2812                                </tp:ActionContext>
2813                                <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2814                            </tp:ThisPartyActionBinding>
2815                            <tp:OtherPartyActionBinding>NegoResp_ABID10</tp:OtherPartyActionBinding>
2816                        </tp:CanReceive>
2817                        <!-- This receive is for receiving the Negotiation Counter Offer. This happens in "CPA Counter Offer 1 BTA"--
2818        >
2819                        <tp:CanReceive>
2820                            <tp:ThisPartyActionBinding tp:id="NegoInit_ABID12" tp:action="CPA_Counter_Offer_BT_RespBA"
2821        tp:packageId="NegoInit_CounterOfferRequestPackage">
2822                                <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
```

Automated Negotiation Specification   **Page 76 of 97**

```
2823   tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2824   tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2825                             <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2826   tp:businessTransactionActivity="CPA Counter Offer 1 BTA"
2827   tp:requestOrResponseAction="CPA_Counter_Offer_BT_ReqBA">
2828                             <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
2829                             </tp:ActionContext>
2830                             <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2831                         </tp:ThisPartyActionBinding>
2832                         <tp:OtherPartyActionBinding>NegoResp_ABID12</tp:OtherPartyActionBinding>
2833                     </tp:CanReceive>
2834                     <!-- This Receive is for receiving the Receipt Acknowledgment -->
2835                     <tp:CanReceive>
2836                         <tp:ThisPartyActionBinding tp:id="NegoInit_ABID7" tp:action="ReceiptAcknowledgment"
2837   tp:packageId="NegoInit_ReceiptAcknowledgmentPackage">
2838                             <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="true"
2839   tp:isNonRepudiationReceiptRequired="true" tp:isConfidential="transient" tp:isAuthenticated="persistent"
2840   tp:isTamperProof="persistent" tp:isAuthorizationRequired="true"/>
2841                             <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2842                         </tp:ThisPartyActionBinding>
2843                         <tp:OtherPartyActionBinding>NegoResp_ABID7</tp:OtherPartyActionBinding>
2844                     </tp:CanReceive>
2845                     <!-- This Receive is for receiving the Exception -->
2846                     <tp:CanReceive>
2847                         <tp:ThisPartyActionBinding tp:id="NegoInit_ABID8" tp:action="Exception"
2848   tp:packageId="NegoInit_ExceptionPackage">
2849                             <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="true"
2850   tp:isNonRepudiationReceiptRequired="true" tp:isConfidential="transient" tp:isAuthenticated="persistent"
2851   tp:isTamperProof="persistent" tp:isAuthorizationRequired="true"/>
2852                             <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2853                         </tp:ThisPartyActionBinding>
2854                         <tp:OtherPartyActionBinding>NegoResp_ABID8</tp:OtherPartyActionBinding>
2855                     </tp:CanReceive>
2856                 </tp:ServiceBinding>
2857             </tp:CollaborationRole>
2858             <!-- Certificates used by the "Negotiation Initiator" company -->
2859             <tp:Certificate tp:certId="NegoInit_AppCert">
2860                 <ds:KeyInfo>
2861                     <ds:KeyName>NegoInit_AppCert_Key</ds:KeyName>
2862                 </ds:KeyInfo>
2863             </tp:Certificate>
2864             <tp:SecurityDetails tp:securityId="NegoInit_MessageSecurity">
2865                 <tp:TrustAnchors>
2866                     <tp:AnchorCertificateRef tp:certId="NegoInit_AppCert"/>
2867                 </tp:TrustAnchors>
2868             </tp:SecurityDetails>
2869             <tp:DeliveryChannel tp:channelId="asyncChannelA1" tp:transportId="transportA1"
2870   tp:docExchangeId="docExchangeA1">
2871                 <tp:MessagingCharacteristics tp:syncReplyMode="none" tp:ackRequested="always"
2872   tp:ackSignatureRequested="always" tp:duplicateElimination="always"/>
2873             </tp:DeliveryChannel>
2874             <tp:Transport tp:transportId="transportA1">
2875                 <tp:TransportSender>
2876                     <tp:TransportProtocol tp:version="1.1">HTTP</tp:TransportProtocol>
2877                     <tp:AccessAuthentication>basic</tp:AccessAuthentication>
2878                 </tp:TransportSender>
2879                 <tp:TransportReceiver>
2880                     <tp:TransportProtocol tp:version="1.1">HTTP</tp:TransportProtocol>
2881                     <tp:AccessAuthentication>basic</tp:AccessAuthentication>
2882                     <tp:Endpoint tp:uri="https://www.NegoInit.com/servlets/ebxmlhandler/async" tp:type="allPurpose"/>
2883                 </tp:TransportReceiver>
2884             </tp:Transport>
2885             <tp:DocExchange tp:docExchangeId="docExchangeA1">
2886                 <tp:ebXMLSenderBinding tp:version="2.0"/>
2887                 <tp:ebXMLReceiverBinding tp:version="2.0"/>
2888             </tp:DocExchange>
```

Automated Negotiation Specification   **Page 77 of 97**

```
2889        </tp:PartyInfo>
2890        <!-- Party info for Negotiation Responder -->
2891        <tp:PartyInfo tp:partyName="NegotiationResponder" tp:defaultMshChannelId="asyncChannelB1"
2892   tp:defaultMshPackageId="NegoInit_MshSignalPackage">
2893            <tp:PartyId tp:type="urn:oasis:names:tc:ebxml-cppa:partyid-type:duns">123456789</tp:PartyId>
2894            <tp:PartyRef xlink:href="http://NegoResp.com/about.html"/>
2895            <!-- This role is for Negotiation Responder performing the role of Negotiation Responder -->
2896            <tp:CollaborationRole>
2897                <tp:ProcessSpecification tp:version="2.0" tp:name="CPPA-Negotiation" xlink:type="simple"
2898   xlink:href="http://www.oasis-open.org/committees/ebxml-cppa-negot/CPA_Negotiation_BPSS.xml"
2899   tp:uuid="bpid:ebXML:CPPA-Negotiation"/>
2900                <tp:Role tp:name="CPA Negotiation Responder" xlink:type="simple" xlink:href="http://www.oasis-
2901   open.org/committees/ebxml-cppa-negot/CPA_Negotiation_BPSS.xm#CPA Negotiation Responder"/>
2902                <tp:ServiceBinding>
2903                    <tp:Service>bpid:ebXML:CPPA-Negotiation</tp:Service>
2904                    <!-- This send is for sending the Negotiation Offer Response, this could be accept, pending, response-->
2905                    <tp:CanSend>
2906                        <tp:ThisPartyActionBinding tp:id="NegoResp_ABID9" tp:action="CPA_Offer_BT_RespBA"
2907   tp:packageId="NegoInit_OfferResponsePackage">
2908                            <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
2909   tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2910   tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2911                            <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2912   tp:businessTransactionActivity="CPA Offer BTA" tp:requestOrResponseAction="CPA_Offer_BT_RespBA"/>
2913                            <tp:ChannelId>asyncChannelB1</tp:ChannelId>
2914                        </tp:ThisPartyActionBinding>
2915                        <tp:OtherPartyActionBinding>NegoInit_ABID9</tp:OtherPartyActionBinding>
2916                    </tp:CanSend>
2917                    <!-- This send is for sending the Final Response document in the final BTA -->
2918                    <tp:CanSend>
2919                        <tp:ThisPartyActionBinding tp:id="NegoResp_FinalResponseB" tp:action="Final_CPA_BT_ReqBA"
2920   tp:packageId="NegoInit_FinalMessage">
2921                            <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
2922   tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2923   tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2924                            <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2925   tp:businessTransactionActivity="CPA Final BTA" tp:requestOrResponseAction="Final_CPA_BT_ReqBA"/>
2926                            <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2927                        </tp:ThisPartyActionBinding>
2928                        <tp:OtherPartyActionBinding>NegoInit_FinalResponseA</tp:OtherPartyActionBinding>
2929                    </tp:CanSend>
2930                    <!-- This send is for sending the Receipt Acknowledgment -->
2931                    <tp:CanSend>
2932                        <tp:ThisPartyActionBinding tp:id="NegoResp_ABID3" tp:action="ReceiptAcknowledgement"
2933   tp:packageId="NegoInit_ReceiptAcknowledgmentPackage">
2934                            <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="true"
2935   tp:isNonRepudiationReceiptRequired="true" tp:isConfidential="transient" tp:isAuthenticated="persistent"
2936   tp:isTamperProof="persistent" tp:isAuthorizationRequired="true"/>
2937                            <tp:ChannelId>asyncChannelB1</tp:ChannelId>
2938                        </tp:ThisPartyActionBinding>
2939                        <tp:OtherPartyActionBinding>NegoInit_ABID3</tp:OtherPartyActionBinding>
2940                    </tp:CanSend>
2941                    <!-- This receive is for receiving the Final message in the collaboration. This would be the double signed
2942   CPA document or acceptance or reject of the CPA  in the final Response document-->
2943                    <tp:CanReceive>
2944                        <tp:ThisPartyActionBinding tp:id="NegoResp_FinalResponseMessageB"
2945   tp:action="Final_CPA_BT_RespBA" tp:packageId="NegoInit_FinalMessage">
2946                            <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
2947   tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2948   tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2949                            <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2950   tp:businessTransactionActivity="CPA Final BTA" tp:requestOrResponseAction="Final_CPA_BT_RespBA"/>
2951                            <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2952                        </tp:ThisPartyActionBinding>
2953                        <tp:OtherPartyActionBinding>NegoInit_FinalResponseMessageA</tp:OtherPartyActionBinding>
2954                    </tp:CanReceive>
```

Automated Negotiation Specification   **Page 78 of 97**

```
2955                    <!-- This receive is for receiving theoffer in the first place -->
2956                    <tp:CanReceive>
2957                        <tp:ThisPartyActionBinding tp:id="NegoResp_ABID1" tp:action="CPA_Offer_BT_ReqBA"
2958    tp:packageId="NegoInit_OfferRequestPackage">
2959                            <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
2960    tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2961    tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2962                            <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2963    tp:businessTransactionActivity="CPA Offer BTA" tp:requestOrResponseAction="CPA_Offer_BT_ReqBA"/>
2964                            <tp:ChannelId>asyncChannelB1</tp:ChannelId>
2965                        </tp:ThisPartyActionBinding>
2966                        <tp:OtherPartyActionBinding>NegoInit_ABID1</tp:OtherPartyActionBinding>
2967                    </tp:CanReceive>
2968                    <!-- This Receive is for receiving the Receipt Acknowledgment -->
2969                    <tp:CanReceive>
2970                        <tp:ThisPartyActionBinding tp:id="NegoResp_ABID2" tp:action="ReceiptAcknowledgment"
2971    tp:packageId="NegoInit_ReceiptAcknowledgmentPackage">
2972                            <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="true"
2973    tp:isNonRepudiationReceiptRequired="true" tp:isConfidential="transient" tp:isAuthenticated="persistent"
2974    tp:isTamperProof="persistent" tp:isAuthorizationRequired="true"/>
2975                            <tp:ChannelId>asyncChannelB1</tp:ChannelId>
2976                        </tp:ThisPartyActionBinding>
2977                        <tp:OtherPartyActionBinding>NegoInit_ABID2</tp:OtherPartyActionBinding>
2978                    </tp:CanReceive>
2979                    <!-- This Receive is for receiving the Exception -->
2980                    <tp:CanReceive>
2981                        <tp:ThisPartyActionBinding tp:id="NegoResp_ABID4" tp:action="Exception"
2982    tp:packageId="NegoInit_ExceptionPackage">
2983                            <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="true"
2984    tp:isNonRepudiationReceiptRequired="true" tp:isConfidential="transient" tp:isAuthenticated="persistent"
2985    tp:isTamperProof="persistent" tp:isAuthorizationRequired="true"/>
2986                            <tp:ChannelId>asyncChannelB1</tp:ChannelId>
2987                        </tp:ThisPartyActionBinding>
2988                        <tp:OtherPartyActionBinding>NegoInit_ABID4</tp:OtherPartyActionBinding>
2989                    </tp:CanReceive>
2990                </tp:ServiceBinding>
2991            </tp:CollaborationRole>
2992            <!-- This role is for Negotiation Responder company performing the role of Negotiation Counter offer initiator -->
2993            <tp:CollaborationRole>
2994                <tp:ProcessSpecification tp:version="2.0" tp:name="CPPA-Negotiation" xlink:type="simple"
2995    xlink:href="http://www.oasis-open.org/committees/ebxml-cppa-negot/CPA_Negotiation_BPSS.xml"
2996    tp:uuid="bpid:ebXML:CPPA-Negotiation"/>
2997                <tp:Role tp:name="CPA Negotiation Counter Offer Initiator" xlink:type="simple" xlink:href="http://www.oasis-
2998    open.org/committees/ebxml-cppa-negot/CPA_Negotiation_BPSS.xm#CPA Negotiation Counter Offer Initiator"/>
2999                <tp:ServiceBinding>
3000                    <tp:Service>bpid:ebXML:CPPA-Negotiation</tp:Service>
3001                    <!-- This send is for sending the Negotiation Counter Offer.This happens in "CPA Counter Offer 1 BTA" -->
3002                    <tp:CanSend>
3003                        <tp:ThisPartyActionBinding tp:id="NegoResp_ABID12" tp:action="CPA_Counter_Offer_BT_ReqBA"
3004    tp:packageId="NegoInit_CounterOfferRequestPackage">
3005                            <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
3006    tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
3007    tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
3008                            <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
3009    tp:businessTransactionActivity="CPA Counter Offer 1 BTA"
3010    tp:requestOrResponseAction="CPA_Counter_Offer_BT_ReqBA">
3011                                <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
3012                            </tp:ActionContext>
3013                            <tp:ChannelId>asyncChannelB1</tp:ChannelId>
3014                        </tp:ThisPartyActionBinding>
3015                        <tp:OtherPartyActionBinding>NegoInit_ABID12</tp:OtherPartyActionBinding>
3016                    </tp:CanSend>
3017                    <!-- This send is for sending the Negotiation Counter Offer response.This happens in "CPA Counter Offer 2
3018    BTA" -->
3019                    <tp:CanSend>
3020                        <tp:ThisPartyActionBinding tp:id="NegoResp_ABID10" tp:action="CPA_Counter_Offer_BT_ReqBA"
```

Automated Negotiation Specification   **Page 79 of 97**

```
3021    tp:packageId="NegoInit_CounterOfferResponsePackage">
3022                            <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
3023    tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
3024    tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
3025                            <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
3026    tp:businessTransactionActivity="CPA Counter Offer 2 BTA"
3027    tp:requestOrResponseAction="CPA_Counter_Offer_BT_RespBA">
3028                                <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
3029                            </tp:ActionContext>
3030                            <tp:ChannelId>asyncChannelB1</tp:ChannelId>
3031                        </tp:ThisPartyActionBinding>
3032                        <tp:OtherPartyActionBinding>NegoInit_ABID10</tp:OtherPartyActionBinding>
3033                    </tp:CanSend>
3034                    <!-- This send is for sending the Receipt Acknowledgment -->
3035                    <tp:CanSend>
3036                        <tp:ThisPartyActionBinding tp:id="NegoResp_ABID7" tp:action="ReceiptAcknowledgement"
3037    tp:packageId="NegoInit_ReceiptAcknowledgmentPackage">
3038                            <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="true"
3039    tp:isNonRepudiationReceiptRequired="true" tp:isConfidential="transient" tp:isAuthenticated="persistent"
3040    tp:isTamperProof="persistent" tp:isAuthorizationRequired="true"/>
3041                            <tp:ChannelId>asyncChannelB1</tp:ChannelId>
3042                        </tp:ThisPartyActionBinding>
3043                        <tp:OtherPartyActionBinding>NegoInit_ABID7</tp:OtherPartyActionBinding>
3044                    </tp:CanSend>
3045                    <!-- This send is for sending the Final CPA message in CPA_Final_BTA_init_Initiator"-->
3046                    <tp:CanSend>
3047                        <tp:ThisPartyActionBinding tp:id="NegoCOResp_FinalMessageB1" tp:action="Final_CPA_BT_ReqBA"
3048    tp:packageId="NegoInit_FinalMessage">
3049                            <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
3050    tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
3051    tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
3052                            <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
3053    tp:businessTransactionActivity="CPA_Final_BTA_init_Initiator" tp:requestOrResponseAction="Final_CPA_BT_ReqBA">
3054                                <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
3055                            </tp:ActionContext>
3056                            <tp:ChannelId>asyncChannelA1</tp:ChannelId>
3057                        </tp:ThisPartyActionBinding>
3058                        <tp:OtherPartyActionBinding>NegoCOR_FinalMessageA1</tp:OtherPartyActionBinding>
3059                    </tp:CanSend>
3060                    <!-- This send is for sending the response to the Final CPA message in CPA_Final_BTA_init_Responder"-->
3061                    <tp:CanSend>
3062                        <tp:ThisPartyActionBinding tp:id="NegoCOResp_FinalMessageResponseB2"
3063    tp:action="Final_CPA_BT_RespBA" tp:packageId="NegoInit_FinalMessage">
3064                            <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
3065    tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
3066    tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
3067                            <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
3068    tp:businessTransactionActivity="CPA_Final_BTA_init_Responder"
3069    tp:requestOrResponseAction="Final_CPA_BT_RespBA">
3070                                <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
3071                            </tp:ActionContext>
3072                            <tp:ChannelId>asyncChannelA1</tp:ChannelId>
3073                        </tp:ThisPartyActionBinding>
3074                        <tp:OtherPartyActionBinding>NegoCOR_FinalMessageResponseA2</tp:OtherPartyActionBinding>
3075                    </tp:CanSend>
3076                    <!-- This receive is for receiving the response forNegotiation Counter Offer, could be accept, reject or again
3077    send a counter offer This happens in "CPA Counter Offer 1 BTA"-->
3078                    <tp:CanReceive>
3079                        <tp:ThisPartyActionBinding tp:id="NegoResp_ABID6" tp:action="CPA_Counter_Offer_BT_RespBA"
3080    tp:packageId="NegoInit_CounterOfferResponsePackage">
3081                            <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
3082    tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
3083    tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
3084                            <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
3085    tp:businessTransactionActivity="CPA Counter Offer 1 BTA"
3086    tp:requestOrResponseAction="CPA_Counter_Offer_BT_RespBA">
```

Automated Negotiation Specification   **Page 80 of 97**

```
3087                         <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
3088                       </tp:ActionContext>
3089                       <tp:ChannelId>asyncChannelB1</tp:ChannelId>
3090                   </tp:ThisPartyActionBinding>
3091                   <tp:OtherPartyActionBinding>NegoInit_ABID6</tp:OtherPartyActionBinding>
3092               </tp:CanReceive>
3093               <!-- This receive is for receiving Negotiation Counter Offer.This happens in "CPA Counter Offer 2 BTA"-->
3094               <tp:CanReceive>
3095                   <tp:ThisPartyActionBinding tp:id="NegoResp_ABID5" tp:action="CPA_Counter_Offer_BT_ReqBA"
3096       tp:packageId="NegoInit_CounterOfferRequestPackage">
3097                       <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
3098       tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
3099       tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
3100                       <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
3101       tp:businessTransactionActivity="CPA Counter Offer 2 BTA"
3102       tp:requestOrResponseAction="CPA_Counter_Offer_BT_ReqBA">
3103                         <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
3104                       </tp:ActionContext>
3105                       <tp:ChannelId>asyncChannelB1</tp:ChannelId>
3106                   </tp:ThisPartyActionBinding>
3107                   <tp:OtherPartyActionBinding>NegoInit_ABID5</tp:OtherPartyActionBinding>
3108               </tp:CanReceive>
3109               <!-- This receive is for receiving the Final CPA message in CPA_Final_BTA_init_Responder"-->
3110               <tp:CanReceive>
3111                   <tp:ThisPartyActionBinding tp:id="NegoCOResp_FinalMessageB" tp:action="Final_CPA_BT_ReqBA"
3112       tp:packageId="NegoInit_FinalMessage">
3113                       <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
3114       tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
3115       tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
3116                       <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
3117       tp:businessTransactionActivity="CPA_Final_BTA_init_Responder" tp:requestOrResponseAction="Final_CPA_BT_ReqBA">
3118                         <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
3119                       </tp:ActionContext>
3120                       <tp:ChannelId>asyncChannelA1</tp:ChannelId>
3121                   </tp:ThisPartyActionBinding>
3122                   <tp:OtherPartyActionBinding>NegoCOR_FinalMessageA</tp:OtherPartyActionBinding>
3123               </tp:CanReceive>
3124               <!-- This receive is for receiving the response to the Final CPA message in CPA_Final_BTA_init_Initiator"-->
3125               <tp:CanReceive>
3126                   <tp:ThisPartyActionBinding tp:id="NegoCOResp_FinalMessageResponseB"
3127       tp:action="Final_CPA_BT_RespBA" tp:packageId="NegoInit_FinalMessage">
3128                       <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
3129       tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
3130       tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
3131                       <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
3132       tp:businessTransactionActivity="CPA_Final_BTA_init_Initiator" tp:requestOrResponseAction="Final_CPA_BT_RespBA">
3133                         <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
3134                       </tp:ActionContext>
3135                       <tp:ChannelId>asyncChannelA1</tp:ChannelId>
3136                   </tp:ThisPartyActionBinding>
3137                   <tp:OtherPartyActionBinding>NegoCOR_FinalMessageResponseA</tp:OtherPartyActionBinding>
3138               </tp:CanReceive>
3139               <!-- This Receive is for receiving the Receipt Acknowledgment -->
3140               <tp:CanReceive>
3141                   <tp:ThisPartyActionBinding tp:id="NegoResp_ABID13" tp:action="ReceiptAcknowledgment"
3142       tp:packageId="NegoInit_ReceiptAcknowledgmentPackage">
3143                       <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="true"
3144       tp:isNonRepudiationReceiptRequired="true" tp:isConfidential="transient" tp:isAuthenticated="persistent"
3145       tp:isTamperProof="persistent" tp:isAuthorizationRequired="true"/>
3146                       <tp:ChannelId>asyncChannelB1</tp:ChannelId>
3147                   </tp:ThisPartyActionBinding>
3148                   <tp:OtherPartyActionBinding>NegoInit_ABID13</tp:OtherPartyActionBinding>
3149               </tp:CanReceive>
3150               <!-- This Receive is for receiving the Exception -->
3151               <tp:CanReceive>
3152                   <tp:ThisPartyActionBinding tp:id="NegoResp_ABID8" tp:action="Exception"
```

Automated Negotiation Specification   **Page 81 of 97**

```
3153    tp:packageId="NegoInit_ExceptionPackage">
3154                        <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="true"
3155    tp:isNonRepudiationReceiptRequired="true" tp:isConfidential="transient" tp:isAuthenticated="persistent"
3156    tp:isTamperProof="persistent" tp:isAuthorizationRequired="true"/>
3157                        <tp:ChannelId>asyncChannelB1</tp:ChannelId>
3158                    </tp:ThisPartyActionBinding>
3159                    <tp:OtherPartyActionBinding>NegoInit_ABID8</tp:OtherPartyActionBinding>
3160                </tp:CanReceive>
3161            </tp:ServiceBinding>
3162        </tp:CollaborationRole>
3163        <!-- Certificates used by the "Negotiation Initiator" company -->
3164        <tp:Certificate tp:certId="NegoResp_AppCert">
3165            <ds:KeyInfo>
3166                <ds:KeyName>NegoResp_AppCert_Key</ds:KeyName>
3167            </ds:KeyInfo>
3168        </tp:Certificate>
3169        <tp:SecurityDetails tp:securityId="NegoResp_MessageSecurity">
3170            <tp:TrustAnchors>
3171                <tp:AnchorCertificateRef tp:certId="NegoResp_AppCert"/>
3172            </tp:TrustAnchors>
3173        </tp:SecurityDetails>
3174        <tp:DeliveryChannel tp:channelId="asyncChannelB1" tp:transportId="transportB1"
3175    tp:docExchangeId="docExchangeB1">
3176            <tp:MessagingCharacteristics tp:syncReplyMode="none" tp:ackRequested="always"
3177    tp:ackSignatureRequested="always" tp:duplicateElimination="always"/>
3178        </tp:DeliveryChannel>
3179        <tp:Transport tp:transportId="transportB1">
3180            <tp:TransportSender>
3181                <tp:TransportProtocol tp:version="1.1">HTTP</tp:TransportProtocol>
3182                <tp:AccessAuthentication>basic</tp:AccessAuthentication>
3183            </tp:TransportSender>
3184            <tp:TransportReceiver>
3185                <tp:TransportProtocol tp:version="1.1">HTTP</tp:TransportProtocol>
3186                <tp:AccessAuthentication>basic</tp:AccessAuthentication>
3187                <tp:Endpoint tp:uri="https://www.NegoResp.com/servlets/ebxmlhandler/async" tp:type="allPurpose"/>
3188            </tp:TransportReceiver>
3189        </tp:Transport>
3190        <tp:DocExchange tp:docExchangeId="docExchangeB1">
3191            <tp:ebXMLSenderBinding tp:version="2.0"/>
3192            <tp:ebXMLReceiverBinding tp:version="2.0"/>
3193        </tp:DocExchange>
3194    </tp:PartyInfo>
3195    <!-- SimplePart corresponding to the SOAP Envelope -->
3196    <tp:SimplePart tp:id="NegoInit_MsgHdr" tp:mimetype="text/xml">
3197        <tp:NamespaceSupported tp:location="http://www.oasis-open.org/committees/ebxml-msg/schema/msg-header-
3198    2_0.xsd" tp:version="2.0">
3199    http://www.oasis-open.org/committees/ebxml-msg/schema/msg-header-2_0.xsd
3200    </tp:NamespaceSupported>
3201    </tp:SimplePart>
3202    <tp:SimplePart tp:id="NegoResp_MsgHdr" tp:mimetype="text/xml">
3203        <tp:NamespaceSupported tp:location="http://www.oasis-open.org/committees/ebxml-msg/schema/msg-header-
3204    2_0.xsd" tp:version="2.0">
3205    http://www.oasis-open.org/committees/ebxml-msg/schema/msg-header-2_0.xsd
3206    </tp:NamespaceSupported>
3207    </tp:SimplePart>
3208    <!-- SimplePart corresponding to a Receipt Acknowledgment business signal -->
3209    <tp:SimplePart tp:id="NegoInit_ReceiptAcknowledgment" tp:mimetype="application/xml">
3210        <tp:NamespaceSupported tp:location="http://www.ebxml.org/bpss/ReceiptAcknowledgment.xsd"
3211    tp:version="2.0">http://www.ebxml.org/bpss/ReceiptAcknowledgment.xsd
3212    </tp:NamespaceSupported>
3213    </tp:SimplePart>
3214    <tp:SimplePart tp:id="NegoResp_ReceiptAcknowledgment" tp:mimetype="application/xml">
3215        <tp:NamespaceSupported tp:location="http://www.ebxml.org/bpss/ReceiptAcknowledgment.xsd" tp:version="2.0">
3216    http://www.ebxml.org/bpss/ReceiptAcknowledgment.xsd
3217    </tp:NamespaceSupported>
3218    </tp:SimplePart>
```

Automated Negotiation Specification   **Page 82 of 97**

```
3219        <!-- SimplePart corresponding to an Exception business signal -->
3220        <tp:SimplePart tp:id="NegoInit_Exception" tp:mimetype="application/xml">
3221            <tp:NamespaceSupported tp:location="http://www.oasis-open.org/committees/ebxml-msg/schema/msg-header-
3222    2_0.xsd" tp:version="2.0">
3223        http://www.ebxml.org/bpss/Exception.xsd
3224      </tp:NamespaceSupported>
3225        </tp:SimplePart>
3226        <tp:SimplePart tp:id="NegoResp_Exception" tp:mimetype="application/xml">
3227            <tp:NamespaceSupported tp:location="http://www.oasis-open.org/committees/ebxml-msg/schema/msg-header-
3228    2_0.xsd" tp:version="2.0">
3229        http://www.ebxml.org/bpss/Exception.xsd
3230      </tp:NamespaceSupported>
3231        </tp:SimplePart>
3232        <!-- SimplePart corresponding to a negotiation offer request action -->
3233        <tp:SimplePart tp:id="NegoInit_OfferRequest" tp:mimetype="application/xml">
3234            <tp:NamespaceSupported tp:location="http://www.ebxml.org/schemas/NegotiationOffer.xsd" tp:version="1.0">
3235        http://www.ebxml.org/schemas/NegotiationOffer.xsd
3236      </tp:NamespaceSupported>
3237        </tp:SimplePart>
3238        <!-- SimplePart corresponding to a Negotiation offer response action (accept) -->
3239        <tp:SimplePart tp:id="NegoInit_OfferAccept" tp:mimetype="application/xml">
3240            <tp:NamespaceSupported tp:location="http://www.ebxml.org/schemas/OfferAccept.xsd" tp:version="1.0">
3241        http://www.ebxml.org/schemas/OfferAccept.xsd
3242      </tp:NamespaceSupported>
3243        </tp:SimplePart>
3244        <!-- SimplePart corresponding to a Negotiation offer response action (reject) -->
3245        <tp:SimplePart tp:id="NegoInit_OfferReject" tp:mimetype="application/xml">
3246            <tp:NamespaceSupported tp:location="http://www.ebxml.org/schemas/OfferReject.xsd" tp:version="1.0">
3247        http://www.ebxml.org/schemas/OfferReject.xsd
3248      </tp:NamespaceSupported>
3249        </tp:SimplePart>
3250        <!-- SimplePart corresponding to a Negotiation offer response action (counter pending) -->
3251        <tp:SimplePart tp:id="NegoInit_OfferCounterPending" tp:mimetype="application/xml">
3252            <tp:NamespaceSupported tp:location="http://www.ebxml.org/schemas/OfferCounterPending.xsd" tp:version="1.0">
3253        http://www.ebxml.org/schemas/OfferCounterPending.xsd
3254      </tp:NamespaceSupported>
3255        </tp:SimplePart>
3256        <!-- SimplePart corresponding to a Negotiation Counter  offer request action  -->
3257        <tp:SimplePart tp:id="NegoInit_CounterOfferRequest" tp:mimetype="application/xml">
3258            <tp:NamespaceSupported tp:location="http://www.ebxml.org/schemas/CounterOffer.xsd" tp:version="1.0">
3259        http://www.ebxml.org/schemas/CounterOfferRequest.xsd
3260      </tp:NamespaceSupported>
3261        </tp:SimplePart>
3262        <!-- SimplePart corresponding to a Negotiation Final document being sent in the negotiation process  -->
3263        <tp:SimplePart tp:id="NegoInit_FinalMessage" tp:mimetype="application/xml">
3264            <tp:NamespaceSupported tp:location="http://www.ebxml.org/schemas/FinalMessage.xsd" tp:version="1.0">
3265        http://www.ebxml.org/schemas/FinalMessage.xsd
3266      </tp:NamespaceSupported>
3267        </tp:SimplePart>
3268        <!-- SimplePart corresponding to a Negotiation Counter  offer request action  -->
3269        <tp:SimplePart tp:id="NegoInit_FinalMessageResponse" tp:mimetype="application/xml">
3270            <tp:NamespaceSupported tp:location="http://www.ebxml.org/schemas/FinalMessageResponse.xsd"
3271    tp:version="1.0">
3272        http://www.ebxml.org/schemas/FinalMessageResponse.xsd
3273      </tp:NamespaceSupported>
3274        </tp:SimplePart>
3275        <!-- An ebXML message with a SOAP Envelope only -->
3276        <tp:Packaging tp:id="NegoInit_MshSignalPackage">
3277            <tp:ProcessingCapabilities tp:parse="true" tp:generate="true"/>
3278            <tp:CompositeList>
3279                <tp:Composite tp:id="NegoInit_MshSignal" tp:mimetype="multipart/related" tp:mimeparameters="type=text/xml">
3280                    <tp:Constituent tp:idref="NegoInit_MsgHdr"/>
3281                </tp:Composite>
3282            </tp:CompositeList>
3283        </tp:Packaging>
3284        <!-- An ebXML message with a SOAP Envelope plus a Offer action payload -->
```

Automated Negotiation Specification   **Page 83 of 97**

```
3285        <tp:Packaging tp:id="NegoInit_OfferRequestPackage">
3286            <tp:ProcessingCapabilities tp:parse="true" tp:generate="true"/>
3287            <tp:CompositeList>
3288                <tp:Composite tp:id="NegoInit_OfferRequestMsgId" tp:mimetype="multipart/related"
3289    tp:mimeparameters="type=text/xml">
3290                    <tp:Constituent tp:idref="NegoInit_MsgHdr"/>
3291                    <tp:Constituent tp:idref="NegoInit_OfferRequest"/>
3292                </tp:Composite>
3293            </tp:CompositeList>
3294        </tp:Packaging>
3295        <!-- An ebXML message with a SOAP Envelope plus a offer response action payload -->
3296        <tp:Packaging tp:id="NegoInit_OfferResponsePackage">
3297            <tp:ProcessingCapabilities tp:parse="true" tp:generate="true"/>
3298            <tp:CompositeList>
3299                <tp:Composite tp:id="NegoInit_OfferResponseAcceptMsgId" tp:mimetype="multipart/related"
3300    tp:mimeparameters="type=text/xml">
3301                    <tp:Constituent tp:idref="NegoInit_MsgHdr"/>
3302                    <tp:Constituent tp:idref="NegoInit_OfferAccept"/>
3303                </tp:Composite>
3304            </tp:CompositeList>
3305            <tp:CompositeList>
3306                <tp:Composite tp:id="NegoInit_OfferResponseRejectMsgId" tp:mimetype="multipart/related"
3307    tp:mimeparameters="type=text/xml">
3308                    <tp:Constituent tp:idref="NegoInit_MsgHdr"/>
3309                    <tp:Constituent tp:idref="NegoInit_OfferReject"/>
3310                </tp:Composite>
3311            </tp:CompositeList>
3312            <tp:CompositeList>
3313                <tp:Composite tp:id="NegoInit_OfferResponsePendingMsgId" tp:mimetype="multipart/related"
3314    tp:mimeparameters="type=text/xml">
3315                    <tp:Constituent tp:idref="NegoInit_MsgHdr"/>
3316                    <tp:Constituent tp:idref="NegoInit_OfferCounterPending"/>
3317                </tp:Composite>
3318            </tp:CompositeList>
3319        </tp:Packaging>
3320        <!-- An ebXML message with a SOAP Envelope plus a counter offer request action payload -->
3321        <tp:Packaging tp:id="NegoInit_CounterOfferRequestPackage">
3322            <tp:ProcessingCapabilities tp:parse="true" tp:generate="true"/>
3323            <tp:CompositeList>
3324                <tp:Composite tp:id="NegoInit_CounterOfferRequestMsgId" tp:mimetype="multipart/related"
3325    tp:mimeparameters="type=text/xml">
3326                    <tp:Constituent tp:idref="NegoInit_MsgHdr"/>
3327                    <tp:Constituent tp:idref="NegoInit_CounterOfferRequest"/>
3328                </tp:Composite>
3329            </tp:CompositeList>
3330        </tp:Packaging>
3331        <!-- An ebXML message with a SOAP Envelope plus a counter offer response action payload -->
3332        <tp:Packaging tp:id="NegoInit_CounterOfferResponsePackage">
3333            <tp:ProcessingCapabilities tp:parse="true" tp:generate="true"/>
3334            <tp:CompositeList>
3335                <tp:Composite tp:id="NegoInit_CounterOfferResponseAcceptMsgId" tp:mimetype="multipart/related"
3336    tp:mimeparameters="type=text/xml">
3337                    <tp:Constituent tp:idref="NegoInit_MsgHdr"/>
3338                    <tp:Constituent tp:idref="NegoInit_OfferAccept"/>
3339                </tp:Composite>
3340            </tp:CompositeList>
3341            <tp:CompositeList>
3342                <tp:Composite tp:id="NegoInit_CounterOfferResponseRejectMsgId" tp:mimetype="multipart/related"
3343    tp:mimeparameters="type=text/xml">
3344                    <tp:Constituent tp:idref="NegoInit_MsgHdr"/>
3345                    <tp:Constituent tp:idref="NegoInit_OfferReject"/>
3346                </tp:Composite>
3347            </tp:CompositeList>
3348            <tp:CompositeList>
3349                <tp:Composite tp:id="NegoInit_CounterOfferResponsePendingMsgId" tp:mimetype="multipart/related"
3350    tp:mimeparameters="type=text/xml">
```

Automated Negotiation Specification   **Page 84 of 97**

```
3351              <tp:Constituent tp:idref="NegoInit_MsgHdr"/>
3352              <tp:Constituent tp:idref="NegoInit_OfferCounterPending"/>
3353          </tp:Composite>
3354        </tp:CompositeList>
3355      </tp:Packaging>
3356      <!-- An ebXML message with a SOAP Envelope plus a Receipt Acknowledgment payload -->
3357      <tp:Packaging tp:id="NegoInit_ReceiptAcknowledgmentPackage">
3358        <tp:ProcessingCapabilities tp:parse="true" tp:generate="true"/>
3359        <tp:CompositeList>
3360          <tp:Composite tp:id="NegoInit_ReceiptAcknowledgmentMsg" tp:mimetype="multipart/related"
3361    tp:mimeparameters="type=text/xml">
3362              <tp:Constituent tp:idref="NegoInit_MsgHdr"/>
3363              <tp:Constituent tp:idref="NegoInit_ReceiptAcknowledgment"/>
3364          </tp:Composite>
3365        </tp:CompositeList>
3366      </tp:Packaging>
3367      <tp:Packaging tp:id="NegoResp_ReceiptAcknowledgmentPackage">
3368        <tp:ProcessingCapabilities tp:parse="true" tp:generate="true"/>
3369        <tp:CompositeList>
3370          <tp:Composite tp:id="NegoResp_ReceiptAcknowledgmentMsg" tp:mimetype="multipart/related"
3371    tp:mimeparameters="type=text/xml">
3372              <tp:Constituent tp:idref="NegoResp_MsgHdr"/>
3373              <tp:Constituent tp:idref="NegoResp_ReceiptAcknowledgment"/>
3374          </tp:Composite>
3375        </tp:CompositeList>
3376      </tp:Packaging>
3377      <!-- An ebXML message with a SOAP Envelope plus an Exception payload -->
3378      <tp:Packaging tp:id="NegoInit_ExceptionPackage">
3379        <tp:ProcessingCapabilities tp:parse="true" tp:generate="true"/>
3380        <tp:CompositeList>
3381          <tp:Composite tp:id="NegoInit_ExceptionMsg" tp:mimetype="multipart/related"
3382    tp:mimeparameters="type=text/xml">
3383              <tp:Constituent tp:idref="NegoInit_MsgHdr"/>
3384              <tp:Constituent tp:idref="NegoInit_Exception"/>
3385          </tp:Composite>
3386        </tp:CompositeList>
3387      </tp:Packaging>
3388      <tp:Packaging tp:id="NegoResp_ExceptionPackage">
3389        <tp:ProcessingCapabilities tp:parse="true" tp:generate="true"/>
3390        <tp:CompositeList>
3391          <tp:Composite tp:id="NegoResp_ExceptionMsg" tp:mimetype="multipart/related"
3392    tp:mimeparameters="type=text/xml">
3393              <tp:Constituent tp:idref="NegoResp_MsgHdr"/>
3394              <tp:Constituent tp:idref="NegoResp_Exception"/>
3395          </tp:Composite>
3396        </tp:CompositeList>
3397      </tp:Packaging>
3398      <tp:Comment xml:lang="en-US">CPPA negotiation between NegoInit.com and NegoResp.com</tp:Comment>
3399    </tp:CollaborationProtocolAgreement>
```

Appendix D BPSS Instance Document for Automated Negotiation (Normative)

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

```
<?xml version="1.0" encoding="UTF-8"?>
<ProcessSpecification xmlns="http://www.ebxml.org/BusinessProcess" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xsi:schemaLocation="http://www.ebxml.org/BusinessProcess ebBPSS1.04.xsd" name="CPPA-Negotiation"
uuid="bpid:ebXML:CPPA-Negotiation" version="R02.00">
    <Documentation>This business process describes CPPA negotiation process</Documentation>
    <!--CPA Offer Document-->
    <BusinessDocument name="CPA Offer Doc" nameID="CPA_Offer_Doc"/>
    <!--CPA Acceptt Offer Document-->
    <BusinessDocument name="CPA Accept Offer Doc" nameID="CPA_Accept_Offer_Doc"/>
    <!--CPA Counter Pending Offer Document-->
    <BusinessDocument name="CPA Counter Pending Offer Doc" nameID="CPA_Counter_Pending_Offer_Doc"/>
    <!--CPA Counter Offer Document-->
    <BusinessDocument name="CPA Counter Offer Doc" nameID="CPA_Counter_Offer_Doc"/>
    <!--CPA Reject Offer Document-->
    <BusinessDocument name="CPA Reject Offer Doc" nameID="CPA_Reject_Offer_Doc"/>
    <!--Changed 09/16 CPA Document. This will probably come from the CPA specification-->
    <BusinessDocument name="CPA Final Doc" nameID="CPA_Final_Doc"/>
    <!--Changed 09/16 . Response to final CPA Document. This will probably come from the CPA specification
         This is used when the CPA is not signed just to show acceptance or denial of final CPA-->
    <BusinessDocument name="CPA Final Response DOC" nameID="CPA_Final_Response_Doc"/>
    <!--Changed 09/16 . Response to final CPA Document which is signed and agreed to create a double singed CPA.
Receiving party will create a certificate over the signed CPA and sent that. This will probably come from the CPA
specification-->
    <BusinessDocument name="CPA Final Response DOC Signed" nameID="CPA_Final_Response_Doc_Signed"/>
    <!-- Changed 09/16. Business Transaction for sending the CPA. This CPA is sent by the party finally accepting the offer-
-->
    <BusinessTransaction name="CPA Final BT" nameID="CPA_Final_BT">
        <RequestingBusinessActivity name="Final_CPA_BT_ReqBA" nameID="Final_CPA_BT_ReqBA"
isAuthorizationRequired="false" isIntelligibleCheckRequired="false" isNonRepudiationReceiptRequired="false"
isNonRepudiationRequired="false">
            <DocumentEnvelope businessDocument="CPA Final Doc" businessDocumentIDRef="CPA_Final_Doc"
isAuthenticated="none" isConfidential="none" isTamperProof="none"/>
        </RequestingBusinessActivity>
        <RespondingBusinessActivity name="Final_CPA_BT_RespBA" nameID="Final_CPA_BT_RespBA"
isAuthorizationRequired="false" isIntelligibleCheckRequired="false" isNonRepudiationRequired="false">
            <DocumentEnvelope businessDocument="CPA Final Response Doc"
businessDocumentIDRef="CPA_Final_Response_Doc" isAuthenticated="none" isConfidential="none"
isPositiveResponse="true" isTamperProof="none"/>
            <DocumentEnvelope businessDocument="CPA Final Response Doc Signed"
businessDocumentIDRef="CPA_Final_Response_Doc_Signed" isAuthenticated="none" isConfidential="none"
isPositiveResponse="true" isTamperProof="none"/>
        </RespondingBusinessActivity>
    </BusinessTransaction>
    <!-- Business Transaction for the original negotiation cppa -->
    <BusinessTransaction name="CPA Offer BT" nameID="CPA_Offer_BT">
        <RequestingBusinessActivity name="CPA_Offer_BT_ReqBA" nameID="CPA_Offer_BT_ReqBA"
isAuthorizationRequired="false" isIntelligibleCheckRequired="false" isNonRepudiationReceiptRequired="false"
isNonRepudiationRequired="false">
            <DocumentEnvelope businessDocument="CPA Offer Doc" businessDocumentIDRef="CPA_Offer_Doc"
isAuthenticated="none" isConfidential="none" isTamperProof="none"/>
        </RequestingBusinessActivity>
        <RespondingBusinessActivity name="CPA_Offer_BT_RespBA" nameID="CPA_Offer_BT_RespBA"
isAuthorizationRequired="false" isIntelligibleCheckRequired="false" isNonRepudiationRequired="false">
            <DocumentEnvelope businessDocument="CPA Accept Offer Doc"
businessDocumentIDRef="CPA_Accept_Offer_Doc" isAuthenticated="none" isConfidential="none"
```

```
3460   isPositiveResponse="true" isTamperProof="none"/>
3461              <DocumentEnvelope businessDocument="CPA Reject Offer Doc"
3462   businessDocumentIDRef="CPA_Reject_Offer_Doc" isAuthenticated="none" isConfidential="none"
3463   isPositiveResponse="false" isTamperProof="none"/>
3464              <DocumentEnvelope businessDocument="CPA Counter Pending Offer Doc"
3465   businessDocumentIDRef="CPA_Counter_Pending_Offer_Doc" isAuthenticated="none" isConfidential="none"
3466   isPositiveResponse="true" isTamperProof="none"/>
3467          </RespondingBusinessActivity>
3468      </BusinessTransaction>
3469      <!-- Business Transaction for sending the counter offer -->
3470      <BusinessTransaction name="CPA Counter Offer BT" nameID="CPA_Counter_Offer_BT">
3471          <RequestingBusinessActivity name="CPA_Counter_Offer_BT_ReqBA" nameID="CPA_Counter_Offer_BT_ReqBA"
3472   isAuthorizationRequired="false" isIntelligibleCheckRequired="false" isNonRepudiationReceiptRequired="false"
3473   isNonRepudiationRequired="false">
3474              <DocumentEnvelope businessDocument="CPA Counter Offer Doc"
3475   businessDocumentIDRef="CPA_Counter_Offer_Doc" isAuthenticated="none" isConfidential="none"
3476   isTamperProof="none"/>
3477          </RequestingBusinessActivity>
3478          <RespondingBusinessActivity name="CPA_Counter_Offer_BT_RespBA"
3479   nameID="CPA_Counter_Offer_BT_RespBA" isAuthorizationRequired="false" isIntelligibleCheckRequired="false"
3480   isNonRepudiationRequired="false">
3481              <DocumentEnvelope businessDocument="CPA Accept Offer Doc"
3482   businessDocumentIDRef="CPA_Accept_Offer_Doc" isAuthenticated="none" isConfidential="none"
3483   isPositiveResponse="true" isTamperProof="none"/>
3484              <DocumentEnvelope businessDocument="CPA Reject Offer Doc"
3485   businessDocumentIDRef="CPA_Reject_Offer_Doc" isAuthenticated="none" isConfidential="none"
3486   isPositiveResponse="false" isTamperProof="none"/>
3487              <DocumentEnvelope businessDocument="CPA Counter Pending Offer Doc"
3488   businessDocumentIDRef="CPA_Counter_Pending_Offer_Doc" isAuthenticated="none" isConfidential="none"
3489   isPositiveResponse="true" isTamperProof="none"/>
3490          </RespondingBusinessActivity>
3491      </BusinessTransaction>
3492      <!-- Main collaboration for negotiation business process -->
3493      <BinaryCollaboration name="CPA Negotiation BC" nameID="CPA_Negotiation_BC"
3494   initiatingRole="CPA_Negotiation_Initiator_Role">
3495          <!-- Role for initiator for negotiation process -->
3496          <Role name="CPA Negotiation Initiator" nameID="CPA_Negotiation_Initiator_Role"/>
3497          <!-- Role for initial responder of business collaboration -->
3498          <Role name="CPA Negotiation Responder" nameID="CPA_Negotiation_Responder_Role"/>
3499          <Start toBusinessState="CPA_Offer_BTA"/>
3500          <BusinessTransactionActivity name="CPA Offer BTA" nameID="CPA_Offer_BTA" businessTransaction="CPA Offer
3501   BT" businessTransactionIDRef="CPA_Offer_BT" fromRole="CPA Negotiation Initiator"
3502   fromRoleIDRef="CPA_Negotiation_Initiator_Role" toRole="CPA Negotiation Responder"
3503   toRoleIDRef="CPA_Negotiation_Responder_Role" isLegallyBinding="false" isConcurrent="false"/>
3504          <CollaborationActivity name="CPA Counter Offer CA" binaryCollaboration="CPA Negotiation Counter Offer BC"
3505   binaryCollaborationIDRef="CPA_Negotiation_CounterOfferBC" fromRole="CPA Negotiation Counter Offer Initiator"
3506   fromRoleIDRef="CPA_Negotiation_CounterOfferInitiator_Role" toRole="CPA Negotiation Counter Offer Responder"
3507   toRoleIDRef="CPA_Negotiation_CounterOfferResponder_Role" preCondition="Initiating Role for this activity corresponds to
3508   Responding Role in CPA Offer BTA"/>
3509          <BusinessTransactionActivity name="CPA Final BTA" nameID="CPA_Final_BTA" businessTransaction="CPA Final
3510   BT" businessTransactionIDRef="CPA_Final_BT" fromRole="CPA Negotiation Responder"
3511   fromRoleIDRef="CPA_Negotiation_Responder_Role" toRole="CPA Negotiation Initiator"
3512   toRoleIDRef="CPA_Negotiation_Initiator_Role" isLegallyBinding="false" isConcurrent="false"/>
3513          <!-- If final CPA BTA goes through fine, then overall collaboration is marked success -->
3514          <Success fromBusinessState="CPA Final BTA" conditionGuard="Success"/>
3515          <!-- If inner collaboration goes through fine, then overall collaboration is marked success. Inner collaboration
3516             Would have gone through the transaction that ends up with either the final CPA (Signed if needed) -->
3517          <Success fromBusinessState="CPA Counter Offer CA" conditionGuard="Success"/>
3518          <!-- If Reject offer document is sent for offer bta collaboration is marked as failure-->
3519          <Failure fromBusinessState="CPA Offer BTA" conditionGuard="BusinessFailure">
3520              <ConditionExpression expressionLanguage="DocumentEnvelopeLanguage" expression="CPA Reject Offer
3521   Doc"/>
3522          </Failure>
3523          <!-- If Final CPA BTA fails for some reason, then collaboration is marked as failure -->
3524          <Failure fromBusinessState="CPA Final BTA" conditionGuard="Failure"/>
3525          <Failure fromBusinessState="CPA Counter Offer CA" conditionGuard="Failure"/>
```

Automated Negotiation Specification   **Page 87 of 97**

```
3526        <!-- Transition to Final CPA offer  binary Transaction if the responder for main transaction accepts the initial offer -->
3527        <Transition fromBusinessState="CPA Offer BTA" toBusinessState="CPA Final BTA">
3528            <ConditionExpression expressionLanguage="DocumentEnvelopeLanguage" expression="CPA Accept Offer
3529    Doc"/>
3530        </Transition>
3531        <!-- Transition to counter offer  binary collaboration if the responder for main transaction returns a counter offer
3532    pending message -->
3533        <Transition fromBusinessState="CPA Offer BTA" toBusinessState="CPA Counter Offer CA">
3534            <ConditionExpression expressionLanguage="DocumentEnvelopeLanguage" expression="CPA Counter Pending
3535    Offer Doc"/>
3536        </Transition>
3537      </BinaryCollaboration>
3538      <BinaryCollaboration name="CPA Negotiation Counter Offer BC" nameID="CPA_Negotiation_CounterOfferBC"
3539    initiatingRole="CPA_Negotiation_CounterOfferInitiator_Role">
3540        <Role name="CPA Negotiation Counter Offer Initiator" nameID="CPA_Negotiation_CounterOfferInitiator_Role"/>
3541        <Role name="CPA Negotiation Counter Offer Responder"
3542    nameID="CPA_Negotiation_CounterOfferResponder_Role"/>
3543        <!-- This collaboration starts with the negotiation process responder sending the counter offer -->
3544        <Start toBusinessState="CPA Counter Offer 1 BTA">
3545        <!-- This transaction activity is for negotiation process responder sending the counter offer -->
3546        <BusinessTransactionActivity name="CPA Counter Offer 1 BTA" nameID="CPA_Counter_Offer_1_BTA"
3547    businessTransaction="CPA Counter Offer BT" businessTransactionIDRef="CPA_Counter_Offer_BT" fromRole="CPA
3548    Negotiation Counter Offer Initiator" fromRoleIDRef="CPA_Negotiation_CounterOfferInitiator_Role" toRole="CPA Negotiation
3549    Counter Offer Non Initiator" toRoleIDRef="CPA_Negotiation_CounterOfferResponder_Role" isLegallyBinding="false"
3550    isConcurrent="false" postCondition="Parties reverse roles they play"/>
3551        <!-- This transaction acitvity is for negotiation process initiator sending the counter offer -->
3552        <BusinessTransactionActivity name="CPA Counter Offer 2 BTA" nameID="CPA_Counter_Offer_2_BTA"
3553    businessTransaction="CPA Counter Offer BT" businessTransactionIDRef="CPA_Counter_Offer_BT" fromRole="CPA
3554    Negotiation Counter Offer Responder" fromRoleIDRef="CPA_Negotiation_CounterOfferResponder_Role" toRole="CPA
3555    Negotiation Counter Offer Initiator" toRoleIDRef="CPA_Negotiation_CounterOfferInitiator_Role" isLegallyBinding="false"
3556    isConcurrent="false" postCondition="Parties reverse roles they play"/>
3557        <BusinessTransactionActivity name="CPA Final BTA Init Initiator" nameID="CPA_Final_BTA_init_Initiator"
3558    businessTransaction="CPA Final BT" businessTransactionIDRef="CPA_Final_BT" fromRole="CPA Negotiation Counter
3559    Offer Initiator" fromRoleIDRef="CPA_Negotiation_CounterOfferInitiator_Role" toRole="CPA Negotiation Counter Offer
3560    Responder" toRoleIDRef="CPA_Negotiation_CounterOfferResponder_Role" isLegallyBinding="false" isConcurrent="false"/>
3561        <BusinessTransactionActivity name="CPA Final BTA Init Responder" nameID="CPA_Final_BTA_init_Responder"
3562    businessTransaction="CPA Final BT" businessTransactionIDRef="CPA_Final_BT" fromRole="CPA Negotiation Counter
3563    Offer Responder" fromRoleIDRef="CPA_Negotiation_CounterOfferResponder_Role" toRole="CPA Negotiation Counter
3564    Offer Initiator" toRoleIDRef="CPA_Negotiation_CounterOfferInitiator_Role" isLegallyBinding="false" isConcurrent="false"/>
3565        <!-- Inner collaboration succeeds if the final BTA which involves sending final CPA succeeds -->
3566        <Success fromBusinessState="CPA Final BTA Init Initiator" conditionGuard="Success"/>
3567        <!-- Inner collaboration succeeds if the final BTA which involves sending final CPA succeeds. This is
3568            the same as above but the difference is this initiated by a different party -->
3569        <Success fromBusinessState="CPA Final BTA Init Responder" conditionGuard="Success"/>
3570        <!-- Inner collaboration fails if the final BTA which involves sending final CPA fails -->
3571        <Failure fromBusinessState="CPA Final BTA Init Initiator" conditionGuard="Failure"/>
3572        <!-- Inner collaboration fails if the final BTA which involves sending final CPA fails. This is
3573            the same as above but the difference is this initiated by a different party -->
3574        <Failure fromBusinessState="CPA Final BTA Init Responder" conditionGuard="Failure"/>
3575        <Failure fromBusinessState="CPA Counter Offer 1 BTA" conditionGuard="BusinessFailure">
3576            <ConditionExpression expressionLanguage="DocumentEnvelopeLanguage" expression="CPA Reject Offer
3577    Doc"/>
3578        </Failure>
3579        <Failure fromBusinessState="CPA Counter Offer 2 BTA" conditionGuard="BusinessFailure">
3580            <ConditionExpression expressionLanguage="DocumentEnvelopeLanguage" expression="CPA Reject Offer
3581    Doc"/>
3582        </Failure>
3583        <!-- If the negotiation process responder (initiator in this innercollaboration) sends  an acceptace offer, negotiation
3584    process responder sends the final CPA -->
3585        <Transition fromBusinessState="CPA Counter Offer 2 BTA" toBusinessState="CPA Final BTA Init Initiator">
3586            <ConditionExpression expressionLanguage="DocumentEnvelopeLanguage" expression="CPA Accept Offer
3587    Doc"/>
3588        </Transition>
3589        <!-- If the negotiation process initiator (responder in this inner collaboration) sends an acceptace offer, negotiation
3590    process initiator sends the final CPA -->
3591        <Transition fromBusinessState="CPA Counter Offer 1 BTA" toBusinessState="CPA Final BTA Init Responder">
```

Automated Negotiation Specification   **Page 88 of 97**

```
3592          <ConditionExpression expressionLanguage="DocumentEnvelopeLanguage" expression="CPA Accept Offer
3593   Doc"/>
3594          </Transition>
3595          <!-- If the negotiation process responder sends counter offer and negotiation process initiator sends a counter offer,
3596   negotiation process initiator sends the counter offer next time -->
3597          <Transition fromBusinessState="CPA Counter Offer 1 BTA" toBusinessState="CPA Counter Offer 2 BTA">
3598              <ConditionExpression expressionLanguage="DocumentEnvelopeLanguage" expression="CPA Counter Pending
3599   Offer Doc"/>
3600          </Transition>
3601          <!-- If the negotiation process initiator sends a counter offer and negotiation process responds sends a counter offer,
3602   negotiation process responder sends the counter offer next time, hence the transition back to original BTA-->
3603          <Transition fromBusinessState="CPA Counter Offer 2 BTA" toBusinessState="CPA Counter Offer 1 BTA">
3604              <ConditionExpression expressionLanguage="DocumentEnvelopeLanguage" expression="CPA Counter Pending
3605   Offer Doc"/>
3606          </Transition>
3607      </BinaryCollaboration>
3608   </ProcessSpecification>
```

Negotiation.spec.20Jan03.doc   1/24/2003 6:30 PM

Appendix E Instance Documents for Business Signals

3609

3610 The XML Schemas of the business signals are defined in [ebBPSS].

Acceptance Acknowledgment

3611

3612 The instance document for the AcceptanceAcknowledgment business signal is available as a text
3613 file at:

Exception

3614

3615 The instance document for the Exception business signal is available as a text file at:

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

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

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!-- edited with XML Spy v4.4 U (http://www.xmlspy.com) by neelakantan kartha (Sterling Commerce) -->
<NegotiationDescriptor xmlns="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpp-cpa-negot-2_0.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://www.oasis-
open.org/committees/ebxml-cppa/schema/cpp-cpa-2_0.xsd
             NDD1.xsd" xmlns:xsd="http://www.w3.org/2001/XMLSchema" documentLocation="C:\Documents and
Settings\nkartha\My Documents\ebxml\negotiation\cpa-example-2_0a.xml">
    <!--The value of cpaid might be negotiable, since a party might require that the cpaid conform to a particular format.
However, automatic negotiation on the values is difficult. For instance, how does a party convey to the other party the kinds
of cpaids that it deems o.k? Without this information being conveyed somehow, it wil be difficult to come to agreement
automatically. Fortunately, this issue pertains to the negotiation algorithm and not the NDD. For version 1, we can stipulate
that the cpaid must be URI -->
    <NegotiableInformationItem xpath="/CollaborationProtocolAgreement/@cpaid">
        <Value/>
    </NegotiableInformationItem>
    <!-- Versions might be negotiable, since one party might have a product that conforms to an earlier version of the spec.
By an OrderedValue, it is implied that there is a preference to the earliervalues or latervalues, as given b the attribute
preferenceOrder-->
    <NegotiableInformationItem xpath="/CollaborationProtocolAgreement/@version">
        <OrderedValue preference="EarlierPreferred">
            <Value> 1.0 </Value>
            <Value> 2_0.a</Value>
        </OrderedValue>
    </NegotiableInformationItem>
    <!--The value can be one of the following three: proposed, agreed and signed. However, I do not think that this attribute
is negotiated-rather one party might set the value of this attribute to agreed, once it is satisfied that a satisfactory CPA has
been reached. Again, how and when this attrribute is set seems part of the negotiation algorithm.
Also, the possible values of this attribute are part of the CPPA schema-hence there is no need to repeat them here -->
    <NegotiableInformationItem xpath="/CollaborationProtocolAgreement/Status/@value">
        <Value/>
    </NegotiableInformationItem>
    <!--Here, I am taking the (simplistic) assumption that each party specfies the earliest time for starting and the latest time
for ending the Start element (that species the  Starting Date and Time for the CPA). No preference function is given. See the
next entry  for an example of how one would encode a piecewise linear preference funciton.        -->
    <NegotiableInformationItem xpath="/CollaborationProtocolAgreement/Start">
        <ValueWithPreferenceMeasure>
            <EndPoints>
                <EarliestStart> 1998-04-07T18:39:09Z </EarliestStart>
                <LatestEnd>2002-11-31T13:20:00.000-05:00 </LatestEnd>
            </EndPoints>
        </ValueWithPreferenceMeasure>
    </NegotiableInformationItem>
    <!--Nothing new here, when compared to the previous NegotiableInformationItem except that this gives an example of a
piecewise linear preference function. I suspect that this might be over engineering at this point -->
    <NegotiableInformationItem xpath="/CollaborationProtocolAgreement/End">
        <ValueWithPreferenceMeasure>
            <EndPoints>
                <EarliestStart> 1998-04-07T18:39:09Z </EarliestStart>
                <LatestEnd>2002-11-31T13:20:00.000-05:00 </LatestEnd>
            </EndPoints>
            <PreferenceFunction>
                <PiecewiseLinearPiece>
                    <x1>1998-04-07T18:39:09Z </x1>
                    <y1>15</y1>
                    <x1>2000-11-31T13:20:00.000-05:00 </x1>
```

Automated Negotiation Specification   **Page 91 of 97**

```
3676                    <y1>30</y1>
3677                </PiecewiseLinearPiece>
3678                <PiecewiseLinearPiece>
3679                    <x1>2000-11-31T13:20:00.000-05:00  </x1>
3680                    <y1>30</y1>
3681                    <x1>2002-11-31T13:20:00.000-05:00 </x1>
3682                    <y1>300</y1>
3683                </PiecewiseLinearPiece>
3684            </PreferenceFunction>
3685         </ValueWithPreferenceMeasure>
3686     </NegotiableInformationItem>
3687     <!--Note the ConversationConstraints is an element that might be present or absent, and hence a party might negotiate
3688     the presence or absence of this element. There are four cases to consider A party (a) insists that an element must be
3689     present (b) insists that an element is absent (c) is ok with the element being present or absent, but has a preference for one
3690     or the other (d) is o.k with the element being present or absent, and is completely agnostic.-->
3691     <NegotiableInformationItem xpath="/CollaborationProtocolAgreement/ConversationConstraints">
3692            <PresentOrNot value="MustBePresent"/>
3693     </NegotiableInformationItem>
3694     <!-- Note that invocationLimit is an attribute of ConversationConstraints that may or may not be present. So first of all,
3695     the presence or absence of this attribute may be negotiable. Then, the value of this attribute may also be negotiable -->
3696     <NegotiableInformationItem xpath="/CollaborationProtocolAgreement/ConversationConstraints/@invocationLimit">
3697            <IntegerValues>
3698                <RangeInfo preferenceOrder="SmallerPreferred">
3699                    <EndPoints>
3700                        <SmallestValue>1</SmallestValue>
3701                        <LatestValue>5</LatestValue>
3702                    </EndPoints>
3703                </RangeInfo>
3704            </IntegerValues>
3705     </NegotiableInformationItem>
3706     <!--concurrentConversations is similar to invocationLimit. I am including this just for the sake of illustrating another use of
3707     the schema -->
3708     <NegotiableInformationItem
3709 xpath="/CollaborationProtocolAgreement/ConversationConstraints/@concurrentConversations">
3710            <IntegerValues>
3711                <PresentOrNot value="MustBePresent"/>
3712                <RangeInfo>
3713                    <EndPoints>
3714                        <SmallestValue>2</SmallestValue>
3715                        <LatestValue>8</LatestValue>
3716                    </EndPoints>
3717                    <PreferenceFunction>
3718                        <FunctionDefinedByEquation> x**2-2*x+3</FunctionDefinedByEquation>
3719                    </PreferenceFunction>
3720                </RangeInfo>
3721            </IntegerValues>
3722     </NegotiableInformationItem>
3723     <!--The partyInfo element raises a number of interesting issues. In the CPA, there can be exactly two partyInfo
3724     elements, hence there is no negotiation on these once a CPA has been formed. Since this NDD refers to
3725     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
3726     the document referred to by this NDD were a CPP. A CPP can have multiple PartyInfo elements and one among these must
3727     be chosen to form the CPA. Thus the issue here is to associate a preference order between several elements at the same
3728     level. A simple way of doing this in a CPP is as follows: -->
3729     <NegotiableInformationItem xpath="/CollaborationProtocolProfile/PartyInfo[2]">
3730            <Preference value="1"/>
3731     </NegotiableInformationItem>
3732     <NegotiableInformationItem xpath="/CollaborationProtocolProfile/PartyInfo[1]">
3733            <Preference value="3"/>
3734     </NegotiableInformationItem>
3735     <NegotiableInformationItem xpath="/CollaborationProtocolProfile/PartyInfo[3]">
3736            <Preference value="2"/>
3737     </NegotiableInformationItem>
3738     <!--This is included to provide an example where the cardinality of an element may be negotiable. It might be the case
3739     that the number of PartyId elements within a partyInfo element is negotiable (because, say of limitations the underlying
3740     system has of handling a large number of partyIds) -->
3741     <NegotiableInformationItem xpath="/CollaborationProtocolAgreement/PartyInfo/PartyId">
```

Automated Negotiation Specification   **Page 92 of 97**

```
3742          <Cardinality>
3743            <RangeInfo>
3744              <EndPoints>
3745                <SmallestValue>1</SmallestValue>
3746                <LatestValue>5</LatestValue>
3747              </EndPoints>
3748            </RangeInfo>
3749          </Cardinality>
3750        </NegotiableInformationItem>
3751        <!--Example of a boolean value-->
3752        <NegotiableInformationItem
3753   xpath="/CollaborationProtocolAgreement/PartyInfo/CollaborationRole/ServiceBinding/Service/CanSend/ThisPartyActionBind
3754   ing/BusinessTransactionCharacteristics/@isNonRepudiationRequired">
3755          <BooleanValue preference="TruePreferred">
3756            <PresentOrNot value="MustBePresent"/>
3757          </BooleanValue>
3758        </NegotiableInformationItem>
3759        <!--Example of negotiating a duration-->
3760        <NegotiableInformationItem
3761   xpath="/CollaborationProtocolAgreement/PartyInfo/CollaborationRole/ServiceBinding/Service/CanSend/ThisPartyActionBind
3762   ing/BusinessTransactionCharacteristics/@timeToAcknowledgeReceipt">
3763          <DurationWithPreference preferenceOrder="SmallerPreferred">
3764            <MinimumDuration> PT5M </MinimumDuration>
3765            <MaximumDuration>PT6M</MaximumDuration>
3766          </DurationWithPreference>
3767        </NegotiableInformationItem>
3768        <!--This is how would express that one element of an enumeration must be present. Note that the possible values of the
3769   enumeration is defined in the cpp-cpa schema and need not be repeated here-->
3770        <NegotiableInformationItem
3771   xpath="/CollaborationProtocolAgreement/PartyInfo/DeliveryChannel/MessagingCharactersitcs/@syncReplyMode">
3772          <OrderedValue>
3773            <PresentOrNot value="MustBePresent"/>
3774            <Value> signalsOnly</Value>
3775          </OrderedValue>
3776        </NegotiableInformationItem>
3777   </NegotiationDescriptor>
3778   <!--Notes
3779   0. The top element of an NDD document is named NegotiationDescriptor.  The NegotiationDescriptor element contains
3780      NegotiationInformationItem elements for each item that is negotiable. (
3781   1. The documentLocation attribute of NegotiationDescriptor element is a uri that  points to the document for which this >
3782      NDD document pertains to. In particular, the xpath attribute of a NegotiablInformationItem element is an xpath of
3783      this document. The documentLocation attribute is a required attribute.
3784
3785
3786
3787
3788   Non-Negotiable elements and Attributes
3789   ================================
3790   1.  CollaborationProtocolAgreement
3791   2.  CollaborationProtocolAgreement/@schemaLocation
3792   3.  CollaborationProtocolAgreement/Status
3793   4.  CollaborationProtocolAgreement/PartyInfo/@partyName (Since this is set by each party, it is difficult to see how this
3794   would be negotiable. If it is, it would be similar to /CollaborationProtocolAgreement/@cpaid)
3795   5.  CollaborationProtocolAgreement/PartyInfo/@defaultMshChannelId and
3796   CollaborationProtocolAgreement/PartyInfo/@defaultMsPackageId  (Again, if these are negotiable, it would be as a result of
3797   the negotiation algoithm recognizing that the default values are not reasonable. Again, only the value can be negotiated, as
3798   in /CollaborationProtocolAgreement/@cpaid)
3799   6.  CollaborationProtocolAgreement/PartyInfo/PartyRef/@xlink:type(always  simple)
3800   7.  CollaborationProtocolAgreement/PartyInfo/CollaborationRole/ProcessSpecification
3801   8.  CollaborationProtocolAgreement/PartyInfo/CollaborationRole/ProcessSpecification@name
3802   9.  CollaborationProtocolAgreement/PartyInfo/CollaborationRole/ProcessSpecification@xlink:type (always simple)
3803   10.
3804   CollaborationProtocolAgreement/PartyInfo/CollaborationRole/ProcessSpecification/ds:Reference/ds:Trasforms/ds:Transform
3805   /@ds:Algorithm (fixed by the spec)
3806
3807   Elements and attributes similar to others in the sample
```

Automated Negotiation Specification   **Page 93 of 97**

3808 ===
3809 0. CollaborationProtocolAgreement/PartyInfo/PartyId/@type (similar to /CollaborationProtocolAgreement/@version", with an
3810 enumeration that enumerates the range of understood naming systems)
3811 1. CollaborationProtocolAgreement/PartyInfo/PartyRef (similar to CollaborationProtocolAgreement/PartyInfo/PartyId)
3812 2. CollaborationProtocolAgreement/PartyInfo/PartyRef/@xlink:href (similar to /CollaborationProtocolAgreement/@cpaid)
3813 3. CollaborationProtocolAgreement/PartyInfo/PartyRef/@type (similar to CollaborationProtocolAgreement/@version)
3814 4. CollaborationProtocolAgreement/PartyInfo/PartyRef/@schemaLocation
3815 (similar to /CollaborationProtocolAgreement/@cpaid)
3816
3817 5. CollaborationProtocolAgreement/PartyInfo/CollaborationRole (similar to negotiating the cardinality of
3818 /CollaborationProtocolAgreement/PartyInfo/PartyId)
3819
3820 6. CollaborationProtocolAgreement/PartyInfo/CollaborationRole/ProcessSpecification@version (similar to
3821 /CollaborationProtocolAgreement/@version)
3822
3823 7. CollaborationProtocolAgreement/PartyInfo/CollaborationRole/ProcessSpecification@xlink:href and uuid
3824 ((similar to /CollaborationProtocolAgreement/@cpaid)
3825
3826
3827 !-->

Negotiation.spec.20Jan03.doc   1/24/2003 6:30 PM

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

Example of Offer Message Instance Document

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

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

<tp:NegotiationMessage xmlns:tp="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpa-negot-1_0.xsd"
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cppa="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpp-cpa-2_0.xsd"
 xsi:schemaLocation="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpa-negot-1_0.xsd
negotiationMsg.xsd"
 businessMsgId="busMsg001" binding="false" negotiationDialogId="negotDialog001" offerId="offer001"
messageType="Offer">
   <tp:NCPA uri="http://www.companya.com/ncpa/myncpa.xml"/>
   <tp:CPATemplateId id="uri:companyA-and-companyB-CPA1" cppa:version="1.0"/>
   <cppa:SecurityDetails cppa:securityId="ID">
       <cppa:SecurityPolicy></cppa:SecurityPolicy>
   </cppa:SecurityDetails>
   <tp:InitiatingParty>
       <cppa:PartyId cppa:type="urn:oasis:names:tc:ebxml-cppa:partyid-type:duns">123456789</cppa:PartyId>
       <CPPId id="companya-cpp123456789" cppa:version="1.0"/>
   </tp:InitiatingParty>
   <tp:RespondingParty>
       <cppa:PartyId cppa:type="urn:oasis:names:tc:ebxml-cppa:partyid-type:duns">987654321</cppa:PartyId>
       <CPPId id="companyb-cpp987654321" cppa:version="1.0"/>
   </tp:RespondingParty>
   <tp:BPSSBusinessDocumentName name="CPA_Offer_Doc"></tp:BPSSBusinessDocumentName>
   <ExpirationDate>2002-12-20T00:00:00Z</ExpirationDate>
   <tp:BusinessDocuments>
     <CPATemplateDoc>
       <NDD>
           <Uri>http://www.companya.com/proposedncpa/ncpa.xml</Uri>
       </NDD>
       <CPATemplate>
           <Uri>http://www.companya.com/proposedcpa/companya-companyb-cpa1234.xml</Uri>
       </CPATemplate>
     </CPATemplateDoc>
   </tp:BusinessDocuments>
   <tp:NegotiationContent>
       <tp:AcceptedItem xpath="/"/>
   </tp:NegotiationContent>
</tp:NegotiationMessage>
```

Example of Counter-Offer Message Instance Document

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

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

<tp:NegotiationMessage xmlns:tp="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpa-negot-1_0.xsd"
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cppa="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpp-cpa-2_0.xsd"
```

```
3881   xsi:schemaLocation="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpa-negot-1_0.xsd
3882  negotiationMsg.xsd"
3883   businessMsgId="busMsg002" binding="false"  inresponseTo="busMsg001" negotiationDialogId="negotDialog001"
3884  offerId="offer001" messageType="CounterOffer">
3885    <tp:NCPA uri="http://www.companya.com/ncpa/myncpa.xml"/>
3886    <tp:CPATemplateId id="uri:companyA-and-companyB-CPA1" cppa:version="1.0"/>
3887    <cppa:SecurityDetails cppa:securityId="ID">
3888      <cppa:SecurityPolicy></cppa:SecurityPolicy>
3889    </cppa:SecurityDetails>
3890    <tp:InitiatingParty>
3891      <cppa:PartyId cppa:type="urn:oasis:names:tc:ebxml-cppa:partyid-type:duns">123456789</cppa:PartyId>
3892      <CPPId id="companya-cpp123456789" cppa:version="1.0"/>
3893    </tp:InitiatingParty>
3894    <tp:RespondingParty>
3895      <cppa:PartyId cppa:type="urn:oasis:names:tc:ebxml-cppa:partyid-type:duns">987654321</cppa:PartyId>
3896      <CPPId id="companyb-cpp987654321" cppa:version="1.0"/>
3897    </tp:RespondingParty>
3898    <tp:BPSSBusinessDocumentName name="CPA_Counter_Offer_Doc"></tp:BPSSBusinessDocumentName>
3899    <ExpirationDate>2002-12-20T00:00:00Z</ExpirationDate>
3900    <tp:BusinessDocuments>
3901      <CPATemplateDoc>
3902        <NDD>
3903          <Uri>http://www.companyb.com/proposedncpa/ncpa.xml</Uri>
3904        </NDD>
3905        <CPATemplate>
3906          <Uri>http://www.companyb.com/proposedcpa/companya-companyb-cpa1234.xml</Uri>
3907        </CPATemplate>
3908      </CPATemplateDoc>
3909    </tp:BusinessDocuments>
3910    <tp:NegotiationContent>
3911      <tp:AcceptedItem xpath="/CollaborationProtocolAgreement/PartyInfo[0]"/>
3912      <tp:AcceptedItem xpath="/CollaborationProtocolAgreement/PartyInfo[1]/Certificate"/>
3913      <tp:AcceptedItem xpath="/CollaborationProtocolAgreement/PartyInfo[1]/SecurityDetails"/>
3914      <tp:AcceptedItem xpath="/CollaborationProtocolAgreement/PartyInfo[1]/DeliveryChannel"/>
3915      <tp:AcceptedItem xpath="/CollaborationProtocolAgreement/PartyInfo[1]/Transport"/>
3916      <tp:AcceptedItem xpath="/CollaborationProtocolAgreement/PartyInfo[1]/DocExchange"/>
3917      <tp:UpdatedItem
3918  xpath="/CollaborationProtocolAggreement/PartyInfo[1]/CollaborationRole/ServiceBinding/Cansend[0]/ThisPartyAction
3919  Binding/BusinessTransactionCharacteristics@isNonRepudiationRequired"
3920      originalValue="true"
3921      proposedValue="false"/>
3922    </tp:NegotiationContent>
3923  </tp:NegotiationMessage>
```

Appendix H Glossary of Terms

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

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

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

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

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

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

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

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

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

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

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