

# Automated Negotiation of Collaboration- Protocol Agreements Specification Version 0.01

## OASIS ebXML Collaboration Protocol Profile and Agreement Technical Committee

Date TBD

### **Status of this Document**

This document specifies an ebXML SPECIFICATION for the eBusiness community.

Distribution of this document is unlimited.

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

#### ***This version:***

URL TBD

#### ***Errata for this version:***

URL TBD

#### ***Previous version:***

URL TBD

## 30 **1 Automated Negotiation Sub Team Members**

31 Martin Sachs, 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

## 2 Table of Contents

45	Status of this Document.....	1
46	1 Automated Negotiation Sub Team Members .....	2
47	2 Table of Contents.....	3
48	3 Introduction .....	5
49	3.1 Summary of Contents of Document .....	5
50	3.2 Definition and Scope of this Specification .....	5
51	3.3 Document Conventions .....	5
52	3.4 Versioning of the Specification, Schema, and Related Documents .....	6
53	3.5 Definitions .....	6
54	3.6 Audience.....	6
55	3.7 Assumptions .....	6
56	3.8 Related Documents .....	7
57	3.9 Acknowledgments .....	7
58	4 Design Objectives.....	8
59	5 System Overview.....	10
60	5.1 Main Components of CPA Negotiation.....	10
61	5.2 Overview of CPA Negotiation.....	12
62	5.3 CPP and NDD Formation and Editing.....	14
63	5.4 Discovery of CPPs and CPA Templates.....	14
64	5.5 Negotiation through an Intermediary.....	14
65	6 CPA Template .....	15
66	6.1 Advantages of Starting Negotiation with a CPA Template .....	15
67	6.2 CPA Template composition.....	15
68	6.3 Error conditions during CPA Template Composition.....	15
69	7 CPP and CPA Template Content.....	16
70	7.1 Validation of CPP and CPA Template.....	16
71	7.2 Preference Order.....	16
72	7.3 CPA Period of Validity.....	16
73	8 Negotiation CPA (NCPA) .....	18
74	8.1 Document Exchange.....	18
75	8.2 Transport.....	19
76	8.3 Packaging .....	19
77	8.4 Security of Negotiation Process.....	19
78	8.5 Explanation of NCPA Example.....	19
79	9 Pre-Conditions for Negotiation.....	20
80	10 Negotiability of CPA Elements and Attributes.....	21
81	10.1 Enumerations.....	21
82	10.2 CollaborationRole element and its child elements.....	21
83	10.3 Elements or Attributes whose Cardinality Includes Zero .....	22
84	10.4 Values .....	22
85	10.5 Transport Endpoints.....	22
86	10.6 Security.....	22
87	10.6.1 Trust Anchors and Related Matters .....	22
88	10.7 Discussion of Various Elements and Attributes .....	24
89	11 Negotiation Descriptor Document .....	27
90	11.1 Use of NDD.....	27
91	11.2 General Principles of Contents of NDD .....	27
92	11.3 Composition of an NDD for a CPA Template.....	28
93	11.4 Need for Human Input.....	29
94	11.5 Extensibility of CPA.....	29
95	11.6 Explanation of Contents of NDD.....	29
96	12 Negotiation Messages.....	30

97	12.1 CPA Offer/Counter Offer .....	30
98	12.1.1 Message Content.....	30
99	12.1.2 CPA ID, Negotiation Dialog ID, Unique Business Message ID, and InResponseTo .....	31
100	12.1.3 BPSS BusinessDocument Name.....	32
101	12.1.4 Offer and Counter Offer .....	32
102	12.2 CPA Offer rejected .....	32
103	12.3 CPA Offer accepted.....	33
104	12.4 CPA Offer counter pending .....	33
105	12.5 Reasons for Decline.....	33
106	12.6 Status .....	33
107	12.7 Assumptions .....	34
108	13 Negotiation Protocol.....	35
109	13.1 General Principles of Negotiation Protocol .....	35
110	13.2 CPA Identifier.....	35
111	13.3 Negotiation-Dialogue Identifier.....	35
112	13.4 Offer Identifier.....	35
113	13.5 Negotiation Status.....	36
114	13.6 ebXML Conversation .....	36
115	13.7 Negotiation CPA.....	36
116	13.8 Initial Offer .....	37
117	13.9 Simultaneous Initial Offers .....	37
118	13.10 Offer and Counter Offer .....	37
119	13.10.1 Responses to Offer and Counter Offer.....	38
120	13.10.2 Offer-Counter Offer Acceptance Time.....	39
121	13.11 Negotiation Ordering Dependencies.....	39
122	13.12 Conclusion of Negotiation.....	39
123	13.13 Signing the CPA .....	40
124	13.14 Reasons for Rejection during Negotiation.....	41
125	13.15 BPSS Instance for Automated Negotiation.....	41
126	13.15.1 Offer-Counter-Offer Choreography.....	42
127	13.15.2 Final CPA exchange .....	43
128	13.15.3 Negotiation Business Signals.....	44
129	13.15.4 State Diagrams.....	44
130	14 Negotiation Algorithms .....	47
131	14.1 CPPs and NDDs.....	47
132	15 References .....	48
133	16 Conformance .....	49
134	16.1 NDD and Negotiation Messages.....	49
135	16.2 NCPA Instance Document.....	49
136	16.3 Negotiation BPSS Instance Document .....	50
137	16.4 Negotiation Business Signals.....	50
138	17 Disclaimer.....	51
139	18 Contact Information.....	52
140	Appendix A XML Schema for Negotiation Descriptor Document (Normative).....	54
141	Appendix B XML Schema for Negotiation Messages (Normative).....	55
142	Appendix C Negotiation CPA Example (Non-Normative) .....	56
143	Appendix D BPSS Instance Document for Automated Negotiation (Normative).....	69
144	Appendix E Instance Documents for Business Signals .....	73
145	E.1 Acceptance Acknowledgment.....	73
146	E.2 Exception .....	73
147	Appendix F Example of NDD Instance Document (Non-Normative).....	74
148	Appendix G Examples of Negotiation-Message Instance Documents (Non-Normative).....	75
149	Appendix H Glossary of Terms .....	76
150		

## 3 Introduction

### 3.1 Summary of Contents of Document

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

This document is organized as follows:

- Section 3 introduces the specification and discusses various procedural matters
- Section 4 summarizes the design objectives.
- Section 5 is a system-level overview.
- Section 7 discusses content of CPPs and CPA templates with respect to negotiation.
- Section 6 discusses the CPA template.
- Section 8 gives the rules for constructing a Negotiation CPA, the CPA that governs the negotiation process.
- Section 9 discusses conditions that must be met before negotiation can begin.
- Section 10 discusses negotiability of elements and attributes in the CPA.
- Section 11 defines and discusses the Negotiation Descriptor Document (NDD) that is used to describe offers and counter offers.
- Section 12 defines the contents of the negotiation messages.
- Section 13 defines the negotiation protocol including the ebXML Business Process Specification Schema[ebBPSS] instance document that MAY be used to describe the negotiation transactions and their choreography.
- Section 14 discusses negotiation algorithms.
- The appendices include XML Schemas for the NDD and negotiation messages, the BPSS negotiation instance document, examples of an NDD instance document and negotiation message instance documents, non-normative aspects of CPA composition, and a glossary of terms.

### 3.2 Definition and Scope of this Specification

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

This specification defines the rules for automated negotiation of CPAs. It defines the negotiation protocol and the contents of the documents that are part of the negotiation protocol.

### 3.3 Document Conventions

Terms in *Italics* are defined in Appendix H or in the glossary of the CPPA specification[ebCPP].

Terms listed in ***Bold Italics*** represent the element and/or attribute content of the XML *CPP*, *CPA*, or related definitions.

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

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

References to external documents are represented with BLOCK text enclosed in brackets, e.g. [RFC2396]. The references are listed in Section 15.

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

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

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

### 3.4 Versioning of the Specification, Schema, and Related Documents

***TO BE DECIDED.***

### 3.5 Definitions

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

### 3.6 Audience

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

### 3.7 Assumptions

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

### 3.8 Related Documents

Related documents include ebXML specifications on the following topics:

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

See Section 15 for the complete list of references.

### 3.9 Acknowledgments

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

## 4 Design Objectives

This specification defines the protocol, messages, and documents associated with automatically negotiating the contents of a *CPA*. It does NOT define negotiation algorithms in detail. The negotiation algorithm is part of the private process at each *Party* and may embody private or proprietary strategies. This specification does define the rules that ensure interoperability between two *Parties*' negotiation algorithms.

Following are the objectives for the design of this specification.

- The design is based on negotiating the contents of a CPA starting with a CPA template (draft CPA) that one prospective trading partner sends to the other as an initial offer. A CPA template contains elements and attributes that need to be negotiated with a prospective trading partner. A Party can publish a CPA template in a registry or can create one from its CPP and the prospective trading partner's CPP.
- The specification defines the negotiation protocol transactions and choreography by means of an ebXML Business Process Specification Schema[ebBPSS] instance document. 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).
- The negotiation protocol is governed by a Negotiation CPA (NCPA). The NCPA is a standard ebXML CPA that defines a minimal set of function that all Parties can be expected to support without Parties having to negotiate the NCPA before negotiating the CPA for their Business collaboration.
- Avoid requiring changes to the CPPA and BPSS specifications, at least for version 1 of the negotiation spec.
- Use deterministic algorithms
- The negotiation process should converge rapidly.
  - ◆ The process should either succeed or fail.
  - ◆ The process should invoke human intervention on failure
  - ◆ The design should avoid deadlock such as iterative loops that don't advance the state of the negotiation. An example is reiteration over the same offer or counter offer that was previously rejected by either or both parties.
    - The specification should state rules that avoid such iterative loops even if it is decided that automatic detection of loops is out of scope for version 1.
- It must be absolutely clear at any point in the negotiation which party (i.e., only one party) has the initiative to send the next request (counter offer).
  - ◆ The design should avoid race conditions in which both parties simultaneously send an a counter offer. The choreography should make this an error condition.

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

- The design should minimize the amount of state that has to be saved.
- Offer rejection semantics should be strong; rejection should not be a tactical maneuver.
- When more than one result works, the protocol should rank them and find the fairest



- 290 solution. ***ISN'T THIS REALLY A STATEMENT ABOUT THE NEGOTIATION***  
291 ***ALGORITHM, WHICH IS MOSTLY OUT OF SCOPE?***  
292 • The negotiation protocol should be described by a separate state diagram for each party (not  
293 of the process as a whole) since that is how it will be implemented.  
294

## 5 System Overview

The *CPA* negotiation process begins when one *Party* makes an initial offer to a second *Party*. The initial offer consists of a *CPA* template and a *Negotiation Descriptor Document (NDD)* that describes what is negotiable in the *CPA* template.

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

### 5.1 Main Components of CPA Negotiation

Figure 1 illustrates the main components of *CPA* negotiation. The following are shown:

- *NCPA*: The Negotiation *CPA* controls the negotiation process.
- Negotiation BPSS instance document: An ebXML Business Process Specification Schema[ebBPSS] instance document that MAY be used to define the negotiation collaborative protocol. This BPSS instance document is referenced from an *NCPA*.
- *CPP*: Parties A and B publish their *CPPs* in an ebXML Registry[ebRS] or otherwise exchange them when they discover each other.
- *CPA* template: A *CPA* in which some items remained to be filled in by one or the other *Party*, or negotiated between them.
- *NDD*: The Negotiation Descriptor Document, a document associated with a *CPP* or a *CPA* template that defines what is negotiable, ranges of numeric values, etc. The *NDD* is used in the negotiation protocol.
- Negotiation Messages: The messages used to exchange offer and counter-offer information between negotiating *Parties*.
- Negotiation Protocol: The collaborative protocol that produces a negotiated *CPA*. Although shown as a single box in this figure, the negotiation protocol is executed between the two *Parties* or between each *Party* and an intermediary.

***THE MEANS OF ARRIVING AT AN NCPA, AS DESCRIBED IN THE FOLLOWING PARAGRAPH, NEEDS MORE THOUGHT AND AGREEMENT BY THE SUBTEAM.***

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

Also in the registry are one or more standardized *NCPA* templates, which are complete *CPAs* except for party identification and endpoint addresses. Associated with each *Party's CPP* is an

*NCPA*, based on one of the standard *NCPA* templates, containing that *Party's* *Party* identification information and transport endpoint address.

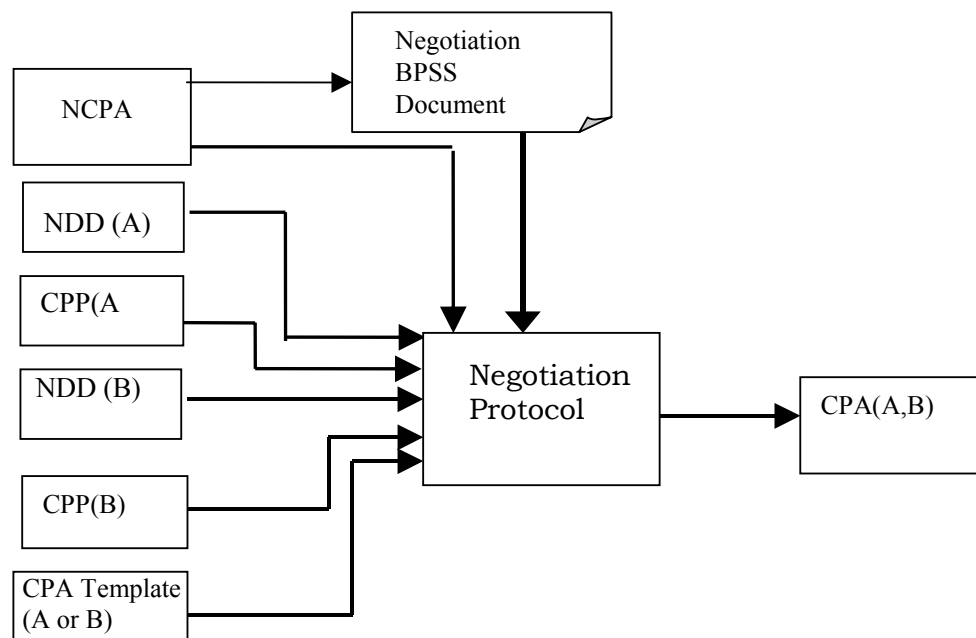
When *Party B* discovers *Party A* as a potential trading partner, *Party B* composes a *CPA* template from its own *CPP* and *Party A's* *CPP*. If *Party A* published an *NDD* along with its *CPP*, *Party B* MAY use the information in *Party A's* *NDD* along with its own *NDD* in composing the *NDD*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

*Party B* also creates an *NCPA* by inserting its identification information and endpoint address into *Party A's* *NCPA*. *Party B* then sends the draft *CPA*, the *NCPA* that it created, and an initial offer (*NDD*) to *Party A* using the information in the *NCPA*.

**THE PROCEDURE FOR ESTABLISHING AN NCPA BETWEEN TWO PARTIES NEEDS MORE CONSIDERATION. IT PROBABLY HAS TO BE IN PLACE BEFORE PARTY A SENDS PARTY B AN INITIAL OFFER.**

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

The two *Parties* can then perform the negotiation protocol, exchanging counter offers until they create an agreed *CPA*. They are then ready to do electronic business.



**Figure 1, Components of CPA Negotiation**

## 5.2 Overview of CPA Negotiation

Figure 2 is a high-level view of the negotiation process. Following are some details of the negotiation process illustrated in Figure 2.

- Initial inputs:
  - ◆ *CPPs* and the associated *NDDs* of two prospective partners or a *CPA* template and *NDD* that one partner provides to a prospective partner.
    - For the case of the *CPA* template and *NDD*, the *CPA* template might be generated by one of the *Parties*, might be a copy of a *CPA* used by someone else that is almost exactly what is needed, or might be supplied by a third-party negotiation service.
  - ◆ Proposed Process-Specification document (BPSS instance document)
    - The partners can negotiate about which BPSS instance document to use based on the name of the BPSS instance document (i.e. syntactic negotiation) but not over the details within a given BPSS instance document (semantic negotiation).
- One Party prepares a CPA template and an NDD that describes what is negotiable and submits the CPA template and NDD to the other Party as an initial offer.
- The two Parties then exchange counter offers until they arrive at a mutually acceptable CPA. Offer and counter-offer information is in negotiation messages exchanged using negotiation business transactions defined in the NCPA and BPSS instance document.
- Result of negotiation:
  - ◆ A successful result is a *CPA* that is ready to sign and use, possibly subject to human approval.
  - ◆ An unsuccessful result means that agreement was not possible on some items in the *CPA*. Possibly, further human interaction could resolve the incompatibilities.
- Concluding negotiation
  - ◆ The *Party* that received the last counter offer builds the complete *CPA* and sends it to the other *Party*. If the *Party* that received the initial offer accepted it without sending a counter offer, that *Party* fills in details such as its Party ID and transport endpoint address and then sends it to the other *Party*.
    - If it was agreed that the *CPA* is to be signed, the *Party* that sends the final *CPA* signs it before sending it.
  - ◆ The other *Party* verifies the contents of the completed *CPA* including, perhaps validation of the first *Party's* signature. If these tests are successful, that *Party* signs the new *CPA* (if signing was agreed to) and returns it to the first *Party*.
  - ◆ The two *Parties* now deploy the new *CPA* and begin doing business.

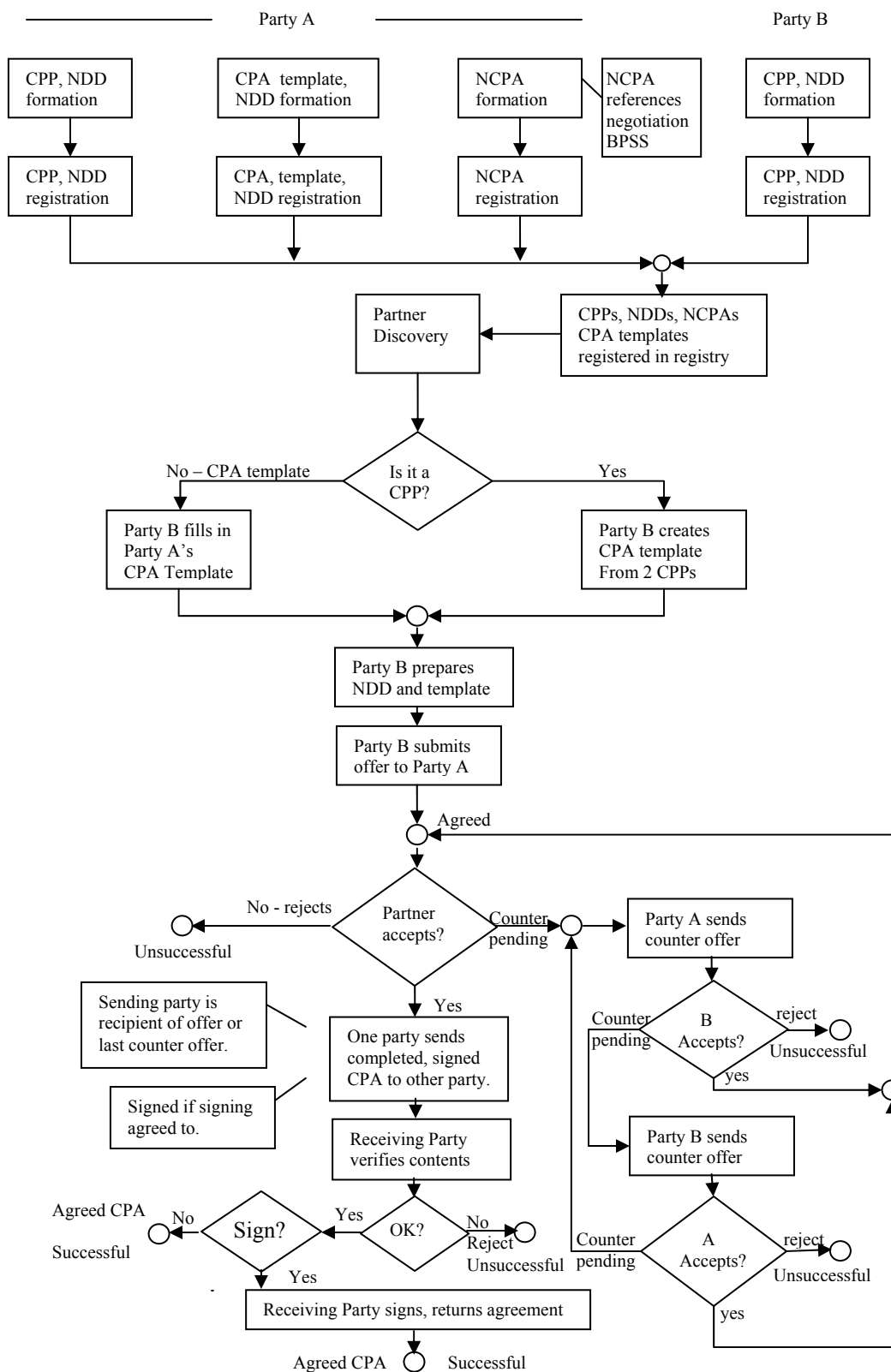


Figure 2, Negotiation Process

### 5.3 CPP and NDD Formation and Editing

These are pre-discovery steps that are out of scope for the negotiation specification, they are included here in the interest of completeness. Following are the elements of *CPP* and *NDD* formation.

- CPP Template
  - ◆ Supplied with software installation (configured options)
  - ◆ Edited to reflect preferences
- NDD formation.
  - ◆ Although *NDD* formation is out of scope, the *NDD* schema is a key component of this specification.
- Tool for custom CPP formation
- Tool for CPA and CPA template formation.
- Tool for NDD formation
- Service(s) for supplying CPPs or CPA templates
  - ◆ UDDI advertised, SOAP, ebXML, simple HTTP GET, and so on.
- ebXML Registry submission (publication)

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

### 5.4 Discovery of CPPs and CPA Templates

The discovery process is out of scope for the negotiation specification; it is included here in the interest of completeness. Following are some points concerning the discovery process.

- The minimum requirement is to be able to perform an HTTP GET of a CPP from a URL obtained by means outside the scope of this specification.
- UDDI ebXML Registry bootstrap. This permits CPPs to be advertised in either UDDI or the ebXML Registry.
- Search and retrieval in ebXML registry or similar registry.
- Well-known address. (ADDRESS OF WHAT?)
- Should/can a registry have any further role(s), perhaps as value-added services?
  - ◆ Notification of *CPP* expirations?
  - ◆ Accept filled-out *CPA* templates?

### 5.5 Negotiation through an Intermediary

Negotiation through an intermediary is out of scope for this version of the specification if it requires a 3-party negotiation *CPA*. It may be possible to use an intermediary if the interactions between each Party and the intermediary are defined by a separate *Negotiation CPA* and a suitable BPSS instance document.

## 6 CPA Template

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

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

### 6.1 Advantages of Starting Negotiation with a CPA Template

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

The result is that the non-controversial aspects of the agreement are recorded in the *CPA* template before negotiation starts. This simplifies the negotiating process by removing from consideration all subjects that were resolved during the composition process. The negotiation process operates on a smaller set of items and will converge rapidly. Eliminating the non-controversial items from the negotiation process simplifies the ontology by focusing attention on what is meant by the items to be negotiated. In addition, the process of composing the new *NDD* will uncover any incompatibilities between the *Parties* before the start of the negotiation process. The two *Parties* can either resolve those incompatibilities by human to human contact or conclude that no resolution is possible, without having first to go through a fruitless negotiation process.

### 6.2 CPA Template composition

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

### 6.3 Error conditions during CPA Template Composition

***THIS SECTION WILL INCLUDE A DISCUSSION OF ERROR CONDITIONS THAT CAN BE DETECTED DURING THE CPA TEMPLATE COMPOSITION PROCESS. ALTERNATIVELY, SHOULD WE RELY ON THE CPA-COMPOSITION APPENDIX OF [EBCPP] FOR THIS INFORMATION?***

## 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

***RULES ARE NEEDED TO COVER SPECIFIC CASES. FOR EXAMPLE A RULE IS NEEDED TO COVER THE CASE WHERE PARTY 1 ALLOWS ELEMENTS A AND B WITH A PREFERENCE FOR A WHILE PARTY 2 ALLOWS ELEMENTS A AND B WITH A PREFERENCE FOR B.***

### 7.3 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 expire. All of these times are negotiable but it will simplify matters if the times in the *CPA* template are mutually consistent. It should be understood that the ***Start*** and ***End*** elements do not appear in the



523 *CPPs*; they must be added when the *CPA* template is composed from the *CPPs*.

## 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, i.e., the CollaborationRole, ProcessSpecification, and Role elements.

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 ackRequested, ackSignatureRequested, and duplicateElimination (they are used only with reliable messaging). For the actor attribute, specify either of the permitted values; this attribute is ignored when ackRequested = “never”.

***THE VALUE OF THE SYNCREPLYMODE ATTRIBUTE SHOULD BE SPECIFIED IN THIS NEGOTIATION SPECIFICATION. IT SHOULD NOT HAVE TO BE NEGOTIATED.***

***THE FOLLOWING IS AN ALTERNATIVE THAT WOULD REQUIRE DEFINING A NEW BINDING IN THE CPPA SPECIFICATION.***

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

## 8.2 Transport

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

## 8.3 Packaging

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

## 8.4 Security of Negotiation Process

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

- If both Parties have the same trust model, negotiation can proceed in a secure fashion.
  - ♦ An initial negotiation of trust anchors and other security matters might be needed. Consider exchanging this information dynamically, using *Message* exchanges.. The might be slower, but simpler, than putting it in the *NCPA*. This might involve human intervention to evaluate and accept the proposed trust model and then to configure the systems to use it for negotiation.
- One Party might have to add a new trust anchor proposed by the other Party.
- The signing certificate need not be the same as the others.
- Certificate validity.
- Are self-signed certificates permitted?
- For the initial version of the specification, omit document-exchange certificates.
- Signing of negotiation Messages has to be covered either in the *NCPA* or in the initial security negotiation mentioned above.

## 8.5 Explanation of NCPA Example

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

***TO BE SUPPLIED.***

## 9 Pre-Conditions for Negotiation

This section discusses conditions that must be met before negotiation. If these conditions are not met, a successful outcome is unlikely. The discussions relate to *CPPs* or a *CPA* template as appropriate

The two partners must agree on what negotiation process to follow, i.e. what *NCPA* to use for negotiation. (The *NCPA* identifies the negotiation BPSS instance to be used.)

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

- There MUST be at least one transport protocol in common.
- There MUST be a minimum level of compatibility between at least one DocumentExchange element in each CPP (DETAILS TO BE DETERMINED).
- There MUST be at least one certificate authority (CA) in common between two CPPs. The CAs are identified in the certificates referred to by AnchorCertificateReference elements.
- THIS LIST WILL BE EXPANDED.

See Section 7 for related information.

## 10 Negotiability of CPA Elements and Attributes

***THIS SECTION IS BASED ON THE WORK BEING DONE WITH THE CPA ELEMENT AND ATTRIBUTE NEGOTIABILITY SPREADSHEET.***

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

### 10.1 Enumerations

There are several cases of enumerations:

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

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

The *CPPA* schema itself is input to the negotiation process. Therefore, enumerations that are defined in full in the *CPPA* schema don't necessarily have to be defined in full in the *NDD* schema.

### 10.2 CollaborationRole element and its child elements

The normal case is that the two *CPPs* are being composed into a *CPA* template specify the same process-specification document; each one specifies a different role (e.g. "buyer" and "seller"). The following considerations apply to roles:

- If both *CPPs* specify the same role (e.g. both specify "buyer"), the situation cannot be resolved automatically. Human contact is needed and one *CPP* must be changed to specify the other role.
- If both *CPPs* specify both roles (i.e. two *CollaborationRole* elements with opposite roles), this cannot be resolved automatically. Human contact is needed and the two *Parties* must agree on which *Party* plays which role.
- If *CPP A* specifies one role and *CPP B* specifies both roles, chose the role in *CPP B* which is opposite to the role specified in *CPP A*.
- If both *CPPs* specify more than one process-specification document (*BPSS* instance document) but there is only one in common to the two *Parties*, use that one.
- If both *CPPs* specify more than one process-specification document that is in common to both of them, human contact is needed to decide whether all the common ones are to be used in the collaboration or which one is to be used.

From the viewpoint of *CPA* composition and negotiation, the best practice is to include only one process specification document in each *CPP*.

While the choice of process-specification document is negotiable, the contents are not negotiable. Those BPSS attributes that can be overridden by corresponding attributes in the *CPA* are negotiable as attributes in the *CPA*.

***IT IS TO BE DETERMINED WHETHER THE CHOICE OF PROCESS-SPECIFICATION DOCUMENT IS NEGOTIABLE IN VERSION 1 OF THIS SPECIFICATION.***

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

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

### **10.4 Values**

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

### **10.5 Transport Endpoints**

Transport endpoints are not really negotiable since any *Party* can define whatever endpoints it chooses. There may be issues of matching endpoint characteristics. One example is the endpoint type. *Parties* may need to negotiate what endpoint types are used.

***IT WAS NOT CLEAR TO THE SUBTEAM HOW MUCH USE WILL BE MADE OF ENDPOINT TYPES OTHER THAN “ALL PURPOSE”. FOR ITEMS WHOSE WIDE USE IS NOT CERTAIN, IT MAY BE BETTER NOT TO DESIGN IN DETAIL IN THE FIRST VERSION. INSTEAD, WE COULD INCLUDE A NON-NORMATIVE NOTE ON WHATEVER WE UNDERSTAND ABOUT EACH SUCH ITEM AND LEAVE IT FOR FUTURE VERSIONS TO CONSIDER THE NEED TO NEGOTIATE IT.***

### **10.6 Security**

***THESE POINTS NEED FURTHER DISCUSSION AND DECISIONS.***

- Negotiation on certificates might require human contact.
- A Party’s unwillingness to handle the proposed trust model is a reason for failure of the negotiation.

#### **10.6.1 Trust Anchors and Related Matters**

This section discusses the kinds of negotiation that might take place for aligning *SecurityDetails* and *TrustAnchors* with various *CertificateRefs*.

There are 3 major levels for alignments in public-key infrastructure (PKI). ***ALIGNMENTS OF OTHER SECURITY CREDENTIALS ALSO NEED TO BE DISCUSSED HERE.***

1. Transport-level security
2. Messaging-level security
3. Application-level security

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

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

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

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

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

***OTHER DETAILS ABOUT ALGORITHMS OR STRENGTHS NEED TO BE ADDED.***

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

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

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

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

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

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

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

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

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

## 10.7 Discussion of Various Elements and Attributes

### ***RULES FOR ADDITIONAL ELEMENTS AND ATTRIBUTES PROBABLY HAVE TO BE ADDED.***

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

***Start*** and ***End*** elements: The value of the ***Start*** element must precede the value of the ***End*** element and the times stated in the ***Start*** and ***End*** elements must not be outside the certificate validity periods. The rules in this paragraph are not really negotiation points but they do bear on the composition process and the validity of the *CPA*.

***defaultMSHChannelId:*** Since a delivery channel contains both *Parties'* properties, the two *Parties* have to agree on both *Parties'* default delivery channels. ***MORE DISCUSSION IS NEEDED ON THIS SUBJECT.***

***defaultMSHPackageId:*** ***A USE CASE IS NEEDED.***

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

***PartyRef:*** ***THE TYPE ATTRIBUTE OF PARTYREF NEEDS A USE CASE FOR NEGOTIABILITY.*** One possible reason to negotiate is that a *Party* may not be able to



understand the other *Party's* **PartyRef** document. For example, the geographical contexts might not match. While negotiating the contents of the **PartyRef** document is out of scope for this specification, negotiating the contents might lead to negotiating the schema (type), which is in scope.

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

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

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

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

**Role**: The two *Parties* have to have opposite roles in a collaboration. This MUST be validated. **THERE IS NO KNOWN USE CASE FOR NEGOTIATING IT.**

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

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

**PackageId** might be negotiable.

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

**CollaborationActivity**: This allows a **Party** to specify a complete path inside the BPSS instance document. It is not usually changed.

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

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

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

832 ***Signing the CPA***: Negotiation of signing is accomplished by negotiating presence of the *CPA*  
833 ***Signature*** element and its child *ds:Signature* elements. See Section 13.13 for details.

## 11 Negotiation Descriptor Document

The *Negotiation Descriptor Document (NDD)* describes what is negotiable in the accompanying *CPP* or *CPA* template. It SHALL describe only the negotiable elements and attributes and SHALL omit those elements and attributes that are not negotiable or for which there is nothing to negotiate.

The *NDD* identifies the *CPP* or *CPA* template. The *CPP* or *CPA* template does not identify the *NDD* since a party may have many different *NDDs* associated with the same *CPP* or *CPA* template. These could be for different negotiation processes, different categories of partner, etc.

### 11.1 Use of NDD

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

NOTE: This means that an item which is initially not negotiable cannot be made negotiable during the negotiation process.

### 11.2 General Principles of Contents of NDD

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

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

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

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

With each XPATH expression, the negotiability of the element or attribute is defined by child

elements. These child elements represent the negotiability characteristics of the element or attribute identified by the XPATH statement. Examples are:

- Cardinality (range of permitted cardinalities)
- For a numeric value, minimum, maximum, and negotiation step size
- For choices, XPATH statements, ID attribute values, qnames, element values, etc. which identify the specific choices within the document being negotiated. Examples in the CPA are certificates, delivery channels, transport protocols, and signature algorithms.

The following rules define what is negotiable at the point referenced by an XPATH expression:

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

If a non-leaf element is referenced, its child elements SHALL NOT be referenced since that could introduce contradictions. In other words, a non-leaf element cannot be negotiable independent of its descendants.

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

Formally, the negotiation defined in this specification begins when one *Party* presents an initial offer, consisting of a CPA template and its NDD, to another *Party*. However, the following RECOMMENDATIONS on constructing an *NDD* for a *CPA* template might facilitate the negotiation process.

It is RECOMMENDED that when *Party A* composes a *CPA* template and NDD for an initial offer to *Party B*, *Party A* take into account the requirements expressed in the *NDD* that goes with *Party B's CPP* (if available). The new *NDD* would be a composite of the two sets of requirements that is acceptable to both parties as a starting point in negotiation. The composite is not like an inclusive OR; it is more like an intersection.

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

*Party A* SHOULD NOT include items in the new *NDD* that were not in *Party B's* original *NDD* because *Party B* did not intend to negotiate on the items that it did not put in its original *NDD*. For those items that were not in *Party B's* initial *NDD*, *Party A* MUST either accept what is in

Party B's *CPP* or recognize that there is an irreconcilable conflict.

Note that it is not mandatory for *Party A* to take *Party B's NDD* into account in composing the *NDD* for the *CPA* template since incompatibilities will anyway be removed during the exchange of counter offers. However, taking *Party B's NDD* into account will speed up convergence (or recognition of fatal incompatibilities) and reduce the possibilities of unnecessary rejects during negotiation. In other words, composing a *CPA* template and combined *NDD* before starting negotiation simplifies the negotiation process by:

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

#### **11.4 Need for Human Input**

***NEGOTIATION OF SOME ITEMS MAY REQUIRE HUMAN INPUT, ESPECIALLY IN VERSION 1. THIS SHOULD BE INDICATED IN THE NDD FOR THOSE ITEMS. WE HAVE TO DEFINE HOW TO INDICATE THAT HUMAN INPUT IS NEEDED.***

#### **11.5 Extensibility of CPA**

Extensibility is provided in the *CPPA* schema. Extensions are negotiable. It can be done by indicating what foreign namespaces a party accepts.

#### **11.6 Explanation of Contents of NDD**

This section discusses the schema and instance example of the *NDD*. See Appendix A for the schema and Appendix F for an example of an instance document.

***THE CONTENTS OF THIS SECTION WILL BE DERIVED FROM THE RESULTS OF THE WORK ON THE ELEMENT AND ATTRIBUTE SPREADSHEET. SPECIFICALLY, A SET OF NEGOTIABILITY PATTERNS WILL BE IDENTIFIED THAT WILL BE USED TO DESIGN THE NDD SCHEMA.***

## 12 Negotiation Messages

A negotiation *Message* includes the details of 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/or the *CPA* or their URLs.

See Appendix B for the complete negotiation-*message* XML Schema. See Appendix G for examples of negotiation-message instance documents.

***THE FOLLOWING MATERIAL IS JEAN ZHENG'S SEPT. 6, 2002 OVERVIEW. IT WILL BE REPLACED BY A COMPLETE ELEMENT BY ELEMENT (DEPTH FIRST) EXPLANATION IN THE STYLE OF THE CPPA SPECIFICATION.***

***AN INITIAL VERSION OF THE SCHEMA WAS DISTRIBUTED ON THE NEGOT LIST SERVER 10/16/02.***

### 12.1 CPA Offer/Counter Offer

#### 12.1.1 Message Content

##### Business Information:

- Reference of proposed NCPA url
- Security Information
  - o SecurityPolicy/Trust Anchor/CertificateInformation
- Initiator's CPP ID
- Potential Partner's CPP ID
- Reference of potential Partner's NDD ID
- Initiating Role: My Party ID
- Responding Role: Potential Partner's Party ID
- CPAID and version(or CPATemplate ID and version)
- Unique Business Message ID – uniquely identify the current Business Message within the scope of one negotiation dialog.
- negotiationDialogId – uniquely identify an ongoing dialog that connects multiple Offer/Counter-Offer transactions
- OfferID – uniquely identify each instance of Offer or Counter-Offer
- InResponseTo – Unique Business Message ID of the previous Offer or Counter-Offer this business message is responding. Can be Null for the initiating Offer of the dialog
- Binding (Yes/No)
- Status: "Offer", "Counter Offer", "Single-Party-Signed", "Signed"
- BPSS BusinessDocument Name
- Expiration Date

##### Business Documents:

- Actual NDD Attachment and/or partner-accessable-url of my NDD
  - o Either a complete URL that both party can access or the actual document is expected.

- Proposed CPA
  - o Either a complete URL that both party can access or the actual document is expected.
- OR
- CPATemplate
  - o Either a complete URL that both party can access or the actual document is expected.
- Negotiation Content (For an initial offer, this element can be empty or it can describe the changes made by the offering *Party* to the information in the other *Party's* *CPP* or *CPA* template when forming the *CPA* template of the initial offer.)
  - o Accepted Items<sup>1</sup>(1...n) (required. These are the items that have been accepted by the sending party during all exchange prior to this message and within the same negotiation dialog. ***(SEE BOLD ITALIC NOTE DIRECTLY BELOW)***
    - Xpath of item
  - o Deleted Items (0...n) (optional element only)
    - Xpath of item
  - o Updated Items (0...n)
    - Xpath of item
    - Original Value of item
    - Proposed Value of item
  - o Inserted Items (0...n) (optional item only)
    - Xpath of item
    - Proposed Value of item

***THE SPECIFICATION MUST STATE WHETHER THE INFORMATION UNDER "NEGOTIATION CONTENT" CONTAINS:***

***- THE RESULT OF PROCESSING THE COUNTER OFFER TO WHICH THIS MESSAGE IS THE RESPONSE,***

***- ALL ITEMS ACCEPTED BY THE SENDING PARTY SINCE THE START OF THE NEGOTIATION DIALOGUE (INCLUDING THE ONES BEING ACCEPTED BY THIS MESSAGE),***

***- ALL ITEMS ACCEPTED BY BOTH PARTIES SINCE THE START OF THE NEGOTIATION DIALOGUE (INCLUDING THE ONES BEING ACCEPTED BY THIS MESSAGE.***

Any of the items listed in Negotiation Content can be either a leaf node or non-leaf node. Non-leaf node will indicates the entire subtree is subject to the corresponding change action. If both Non-leaf node AND its children leaf node are present in Negotiation Content, then the Negotiation Content SHOULD be considered invalid.

**12.1.2 CPA ID, Negotiation Dialog ID, Unique Business Message ID, and InResponseTo**  
CPA ID and its version shall remain the same throughout any negotiation dialog.

Negotiation Dialog ID is used to identify a particular negotiation dialog thread. Negotiation Dialog ID can be CPA ID. The value of CPAID could be used as the value for Negotiation

<sup>1</sup> Item can be either an "Element" or an "Attribute".

Dialog ID.

Unique Business Message ID, an unique id that identify the current business message within the scope of one negotiation thread.

InResponseTo, list the Unique Business Message ID of the last incoming Offer or Counter Offer this current business message is responding to.

### **12.1.3 BPSS BusinessDocument Name**

The name (e.g. "CPA Offer Doc") of each BusinessDocument that participate in BPSS Negotiation process will be placed inside each message. The valid enumeration of this field include:

- CPA Offer Doc
- CPA Accept Offer Doc
- CPA Counter Pending Offer Doc
- CPA Counter Offer Doc
- CPA Reject Offer Doc

***NEED TO ADD ADDITIONAL VALUES. SEE BPSS INSTANCE DOCUMENT.***

### **12.1.4 Offer and Counter Offer**

In the two-party scenario, if Party A initiates the dialog by sending Part B an Offer, Party B sends back a Counter Offer, in order to counter this Counter Offer, Party A sends another "Counter Offer" to Party B. In other words, only the initiating offer is "Offer", the rest of negotiation will be conducted by exchanging "Counter Offers".

Based on Hima's CPPA Negotiation BPSS example, Offer differs from subsequent Counter Offers. Offer will always contain the complete initial CPA document and NDD document, or a CPA Template.

We can use two different schemas, one for the initial Offering, and one for the subsequent Offer and Counter Offer. This implies the very first Offer Transaction is different from the subsequent business transactions.

Last but not the least, throughout this negotiation dialog, each Party can terminate the negotiation by sending "CPA offer Rejected" in responding to an incoming Offer/Counter Offer. Human to human contact is encouraged after "CPA offer Rejected" is sent but before a brand new dialog is initiated.

## **12.2 CPA Offer rejected**

### **Business Information:**

- CPAID and version
- negotiationDialogId
- Initiating Role
- Responding Role
- Binding (Yes/No)



- 1078 - Status: "Rejected"
- 1079 - Reason for Rejection

1080

1081 Possible scenario

- 1082 - Party A sends Party B an Offer(or Proposal if we are going to revise the current CPA negotiation model)
- 1083
- 1084 - Party B didn't send any message back before the Offer/Proposal expired

### 1085 **12.3 CPA Offer accepted**

1086 Business Information:

- 1087 - CPAID
- 1088 - negotiationDialogId
- 1089 - Initiating Role
- 1090 - Responding Role
- 1091 - Binding (Yes/No)
- 1092 - Status: "Accepted"
- 1093 - Expiration Date

1094 Payload:

- 1095 - Signed CPA document

1096

### 1097 **12.4 CPA Offer counter pending**

1098 Business Information:

- 1099 - CPAID and version
- 1100 - negotiationDialogId
- 1101 - Initiating Role
- 1102 - Responding Role
- 1103 - Binding (Yes/No)
- 1104 - Status: "Counter Pending"

1105

### 1106 **12.5 Reasons for Decline**

- 1107 - Expired CPP
- 1108 - Unable to fulfill Security Requirements
- 1109 - Proposed Security Policy is inadequate
- 1110 - Out of sequence counter offer
- 1111 - Negotiation failed to converge

1112

### 1113 **12.6 Status**

- 1114 - Offer
- 1115 - Rejected
- 1116 - Accepted
- 1117 - Counter Pending
- 1118 - Counter Offer
- 1119 - Signed
- 1120 - Expired

1121

1122 **12.7 Assumptions**

- 1123     - In counter-proposals, elements that are not listed as “add/delete/update” are accepted as is.

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

### 13.2 CPA Identifier

When a *Party* creates a *CPA* template, that *Party* shall assign a valid value to the *cpald* attribute in the *CPA* template. See Section 10.7 regarding negotiability of the *cpald* attribute.

### 13.3 Negotiation-Dialogue Identifier

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

If it should be necessary to suspend the negotiation for an extended period, the *Negotiation-Dialogue Identifier* SHALL be used to obtain the state information necessary to resume the negotiation.

The value of the *cpald* 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 *cpald* attribute. The reason is to ensure that if a negotiation fails and the same *CPA* template, with the same value of the *cpald* attribute is used in a second negotiation attempt, uniqueness of the *Negotiation Dialog Identifier* is preserved.

NOTE: Although it is not expected that *Negotiation Dialogues* 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 Identifier* from 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 has a unique *Offer Identifier*. 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 Dialogue*. The *Offer Identifier* is qualified by the *Party Id* of the issuer and the *Negotiation-Dialogue Identifier*.

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

### 13.5 Negotiation Status

The ***Status*** element in the *CPA* records the state of the composition/negotiation process. The states of its ***value*** attribute progress as follows:

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

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

### 13.6 ebXML Conversation

A single *Negotiation Dialogue* (executed without being suspended and resumed) corresponds to a single ebXML *Conversation*.

If this specification (either version 1 or a subsequent version) provides the capability of suspending a negotiation and resuming it at a later time, the *Conversation* ends when the negotiation is suspended. When the negotiation is resumed at a later time, a new *Conversation* is started. Suspending and resuming a negotiation requires that the applications persist all the state information needed for resuming the negotiation later. The *Party* that issues the message which causes the negotiation to resume MUST include the *Negotiation-Dialogue Identifier* in the message.

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

For use with *Message* services, such as “vanilla SOAP”, which have no *Conversation* construct, the *Negotiation-Dialogue Identifier* serves the purpose of a *Conversation* identifier at the application level

### 13.7 Negotiation CPA

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

### 13.8 Initial Offer

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

- If *Party* B discovered the *CPP* of *Party* A (a potential trading partner), *Party* B composes a *CPA* template from its *CPP* and *Party* A's *CPP*. *Party* B then prepares an *NDD* that describes what is negotiable in the *CPA* template. If *Party* A had also published an *NDD*, *Party* B SHOULD take that *NDD* into account in preparing the *NDD* for the initial offer.
- If *Party* B discovered the *CPA* template and *NDD* of *Party* A, *Party* B modifies the *CPA* template to include information about itself, makes other modifications to negotiable items in the *CPA* template that are indicated in the *Party* A's *NDD*, and prepares a new *NDD* to go with the modified *CPA* template.

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

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

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

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

### 13.9 Simultaneous Initial Offers

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

### 13.10 Offer and Counter Offer

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

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

A counter offer SHALL NOT propose a wholesale change of subject matter. For example a counter offer SHALL NOT propose changes in the roles of the participants.

***IT IS TO BE DETERMINED WHETHER A COUNTER OFFER CAN PROPOSE A BPSS INSTANCE OTHER THAN THE ONE THAT WAS IN THE INITIAL OFFER. IF PROPOSING A DIFFERENT BPSS INSTANCE IS NOT PERMITTED IN VERSION 1, THEN THE SPEC SHOULD STATE THAT A PARTY THAT WISHES TO PROPOSE A DIFFERENT BPSS INSTANCE SHALL REJECT THE RECEIVED OFFER OR COUNTER OFFER AND CAN THEN ISSUE ITS OWN INITIAL OFFER INCLUDING***

**THE DESIRED BPSS INSTANCE.**

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

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

If a counter offer contains only indications of acceptance of items, the *Party* that sent it is indicating acceptance of the *CPA* as modified by the prior steps in the negotiation. A *Party* SHALL NOT send a counter offer that has no new proposals if any items in the *NDD* remain to be negotiated.

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

**13.10.1 Responses to Offer and Counter Offer**

Following are the responses to an offer or counter offer.

**THE FOLLOWING LIST MAY NEED TO BE EXTENDED.**

- Accept with no changes (an agreed CPA has been achieved)
  - ◆ Accept
  - ◆ Accept and deploy (dynamic eCommerce)
- Counter offer pending: The counter offer might consist of
  - ◆ Deleted elements and attributes.
  - ◆ Added elements and attributes.
  - ◆ Re-ordered elements using an [XPath]-based list of changes with status of required or preferred.
  - ◆ Changed values of elements and attributes.
  - ◆ Identification of items that were accepted in the previous offer or counter offer.
- Rejection: with reason(s) for rejection. See Section 13.14 for additional information. Rejection is final. It ends the Negotiation Dialogue 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).**

### 13.10.2 Offer-Counter Offer Acceptance Time

A maximum time (interval) for acceptance is associated with each offer or counter offer. If the sender of the offer or counter offer does not receive a response during this interval, the offer or counter offer is considered withdrawn. The sender can then send a new offer or counter offer, or re-send the original one. The acceptance interval is a business-level timeout; processing it is independent of any document-exchange or transport-level message-loss recovery rules.

The initiator of the counter offer is not REQUIRED to send anything to the other *Party* at the when the acceptance interval expires. The initiator SHALL record the current *Negotiation Dialogue* as expired, and when the other responder finally sends a reply, if it is anything other than a reject, the initiator SHALL reply with a reject.

Note that in this scenario, neither the initiator nor the responder SHALL terminate the *Negotiation Dialog* until the reject message has been sent by the initiator.

***DO WE NEED TO PRESCRIBE SOMETHING FOR THE CASE WHERE THE INITIATOR DOES NOT RECEIVE THE REJECT MESSAGE FROM THE RESPONDER IN A REASONABLE TIME?***

***CONSIDER WHETHER ONE OF THE BPSS TIMING ATTRIBUTES CAN BE USED TO REPRESENT THE ACCEPTANCE INTERVAL.***

### 13.11 Negotiation Ordering Dependencies

***SUBJECT TO FURTHER DISCUSSION, THE QUESTION OF NEGOTIATION ORDERING DEPENDENCES IS DEFERRED TO BEYOND VERSION 1.***

### 13.12 Conclusion of Negotiation

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

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

When agreement has been reached following exchanges of counter offers, the *Party* that received and accepted the final counter offer SHALL send the completed *CPA* (or its URL) to the other *Party* for approval. The receiving *Party* SHALL respond, indicating either approval or rejection. If signing was agreed to, the sending *Party* SHALL sign the *CPA* before sending it. The receiving *Party* SHALL check that the new *CPA* conforms to its understanding of what it should be. The receiving *Party* can also validate the first *Party*'s signature. If the receiving *Party*

approves the *CPA*, the receiving *Party* SHALL sign the *CPA* over the first *Party*'s signature and return it to the first *Party*. Otherwise the receiving *Party* SHALL respond indicating rejection.

The *Party* that received the completed *CPA* shall respond in one of the following ways:

- *Message* indicating that a completed *CPA* was received (***BusinessDocument name*** = "*CPA Final Response Doc*")
  - ◆ A separate indicator in the *Message* distinguishes between accept and reject.
- *Message* that sends a completed *CPA* signed by the sender (***BusinessDocument name*** = "*CPA Final Response Doc Signed*").
  - ◆ Used when signing was agreed to and the received *CPA* was signed by the sending *Party*.

Following are some reasons for rejecting the received *CPA*:

- The final *CPA* does not agree with the recipient's understanding of what should be in it (some kind of state-tracking mismatch).
- The signature on the final *CPA* cannot be validated.
- The final *CPA* was not signed although signing was agreed to.

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

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

Acceptance and rejection of the double-signed *CPA* are indicated by business signals. See Section 13.15.2 for details.

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

### 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 should be understood 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.

### 13.14 Reasons for Rejection during Negotiation

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. These are elaborated in Section 6.2.

The rejection message SHALL include reason, contact name, phone, and/or URL for further information.

Following are some reasons for rejection:

***THE FOLLOWING LIST MUST INCLUDE EVERYTHING DEFINED IN THE NEGOTIATION MESSAGES.***

- CPA contents. Examples:
  - ◆ Signature on *CPA* template failed validation.
  - ◆ Signature on agreed *CPA* failed validation
    - *CPA* is not signed until it is agreed to.
  - ◆ proposed security too weak
  - ◆ proposed *Packaging* not supported
  - ◆ unable to support signals requested (process specification document)
- Business relationship
  - ◆ *CPA* unsupported without existing business relationship.
- Negotiation process
  - ◆ 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".

- ◆ Proposed *CPA* previously received and not accepted.
- The current offer's validity interval has expired.
- CPA format problems
  - ◆ Examples: parsing error, data invalid
- Internal System Error

### 13.15 BPSS Instance for Automated Negotiation

The choreography of the negotiation protocol MAY be defined by an instance document of the ebXML Business Process Specification Schema[ebBPSS]. The BPSS instance document for

automated negotiation is in Appendix D.

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

- If the response to the offer is “accept offer”, the choreography transitions to the final *CPA* exchange (see below).
- If the response to the offer is “reject offer”, the choreography immediately concludes.
- If the response to the offer is “counter offer pending”, the choreography then goes into an alternation of counter offer and response between the two *Parties* which continues until:
  - ◆ A response of “accept offer” causes the choreography to transition to the final *CPA* exchange.
  - ◆ A response of “reject offer” concludes the choreography.

Several business document names are defined directly under the ***ProcessSpecification*** element and referenced in various places as described below.

The BPSS instance document defines initiator and responder role names for each binary collaboration, collaboration activity, and binary transaction activity. For simplicity in the explanation below, in most cases, the terms “initiator” and “responder” are used. For each stage of the choreography, the NCPA associates role names with actual *Parties* in the ***Action*** elements under the ***CollaborationRole*** elements.

### 13.15.1 Offer-Counter-Offer Choreography

A counter offer is a requesting document in a new *Business Transaction*, not a response to an offer. To issue a counter offer, the recipient of an offer SHALL reply “counter pending offer” and then issue the counter offer as a new *Business Transaction*. This avoids a race condition with respect to which *Party* sends the next message. It also avoids any need to for the two *Parties* to switch roles.

The choreography begins with the “CPA\_Offer\_BT” ***BusinessTransactionActivity*** element under the “CPA\_Negotiation\_BC” ***BinaryCollaboration*** element. A *CPA* offer message is sent from the “CPA\_Negotiation\_Initiator\_Role” *Party* to the “CPA\_Negotiation\_Responder-Role” *Party* by means of the “CPA\_Offer\_BT” ***BusinessTransaction***. The “CPA\_Offer\_ReqBA” ***RequestingBusinessActivity*** sends the “CPA\_Offer\_Doc” message from the initiator *Party* to the responder *Party*. The “CPA\_Offer\_BT\_RespBA” ***RespondingBusinessActivity*** then sends the response message from the responder *Party* to the initiator *Party*. This message is then evaluated as defined by the ***Success***, ***Failure***, and ***Transition*** elements under the “CPA\_Negotiation\_BC” binary collaboration. These are the elements whose ***fromBusinessState*** attribute has the value “CPA OfferBTA”. The value of the ***expression*** attribute in each of these elements is the name of the response message, as follows:

- ***Success*** element: If the response message is “CPA Accept Offer Doc”, the proposed *CPA* has been accepted by the responder *Party* and the choreography transitions to the final *CPA* exchange.
- ***Failure*** element: If the response message is “CPA Reject Offer Doc”. The proposed *CPA* has been unconditionally rejected by the responder *Party* and the choreography concludes.
- ***Transition*** element: If the response message is “CPA Counter Pending Offer Doc”, the

responder *Party* will send a counter offer as the next business transaction. The **toBusinessState** attribute of the **Transition** element identifies “CPA Counter Offer CA” (the name of the **CollaborationActivity** element) as the next state in the choreography.

If the response message to the “CPA\_Offer\_Doc” message was “CPA Counter Pending Offer Doc”, the transition described above takes place and takes the choreography to state “CPA Counter Offer CA”, i.e. to the **CollaborationActivity** element named “CPA Counter Offer CA”. This **CollaborationActivity** element references the “CPA\_Negotiation\_CounterOfferBC” **BinaryCollaboration** element.

The initial request message is under the “CPA\_Counter\_Offer\_1\_BTA” **BusinessTransactionActivity** element, which is the “from” state for the following. The party which received the original CPA offer is now the initiator in this **BusinessTransactionActivity**. The message is sent from the “CPA\_Negotiation\_CounterOfferInitiator\_Role” role to the “CPA\_Negotiation\_CounterOfferResponder\_Role” role by means of the “CPA\_Counter\_Offer\_BT” **BusinessTransaction**. The message, “CPA\_Counter\_Offer\_Doc”, is identified in the “CPA\_Counter\_Offer\_ReqBA” **RequestingBusinessActivity**. The response message is sent by means of the “CPA\_Counter\_Offer\_BT\_RespBA” **RespondingBusinessActivity**. One of three response messages can be sent, as discussed below. The response message is then evaluated as defined by the **Success**, **Failure**, and **Transition** elements under the “CPA\_Negotiation\_BC” **BinaryCollaboration**.

- **Success** element: If the response message is “CPA Accept Offer Doc”, the proposed *CPA* has been accepted by the responder *Party* and the choreography transitions to the final *CPA* exchange.
- **Failure** element: If the response message is “CPA Reject Offer Doc”, the proposed *CPA* has been unconditionally rejected by the responder *Party* and the choreography concludes.
- **Transition** element: If the response message is “CPA Counter Pending Offer Doc”, the responder *Party* will send a counter offer as the next business transaction. The **toBusinessState** attribute of the **Transition** element identifies the “CPA Counter Offer 2 BTA” **BusinessTransactionActivity** as the next state in the choreography.

If the above transition takes place, it means that the *Party* that was the responder now becomes the initiator to supply a counter offer to the counter offer. The “CPA Counter Offer 2 BTA” **BusinessTransactionActivity** is now performed in the same manner as the the “CPA\_Counter\_Offer\_1\_BTA” **BusinessTransactionActivity**, described above.

The choreography then iterates between the “CPA\_Counter\_Offer\_1\_BTA” **BusinessTransactionActivity** and the “CPA Counter Offer 2 BTA” **BusinessTransactionActivity** until a success or failure is achieved. Success causes the choreography to transition to the final CPA exchange. Failure ends the choreography.

### 13.15.2 Final CPA exchange

When either the initial offer or a counter offer is accepted in full, the choreography transitions to the “CPA Final BT” business transaction. The purpose of this transaction is for the *Party* that accepted the offer or counter offer to send the completed *CPA* to the other *Party*.

If the initial offer was accepted, the next business state is the “CPA Final BTA Init Responder” business transaction activity, which references the “CPA Final BT” business transaction. The initiator *Party* for the “Req BA Final\_CPA\_BT\_ReqBA” requesting business activity sends the “CPA Final Doc” message, containing the CPA or its URL, to the other (responder) *Party*. The responder *Party* checks the CPA and performs the responding business activity conveying one of:

- The “CPA Final Response Doc” *Message* to acknowledge receipt of an acceptable CPA or rejection. Acceptance and rejection are indicated by values of the status indicator in the negotiation *Message*.
- THE ABOVE SENTENCE NEEDS TO BE REVISED TO USE THE CORRECT NAME OF THE STATUS ELEMENT OR ATTRIBUTE WHEN THE MESSAGE SCHEMA IS COMPLETED.**
- The “CPA Final Response Doc Signed” *Message*, to acknowledge receipt of an acceptable signed CPA and return that CPA with the responder *Party*’s signature over the initiator *Party*’s signature.

The *Party* that receives the final (double signed) CPA SHOULD test it for possible error conditions as described in Section 13.12. The *Party* that received the double-signed CPA shall reply with the AcceptanceAcknowledgment business signal if the CPA is acceptable or with the Exception business signal if the CPA is not acceptable. See Section 13.15.3 regarding the business signals.

If a counter offer was accepted in full, the choreography transitions to the “CPA Final BTA Initiator Initiator” business transaction activity, which references the “CPA Final BT” business transaction and proceeds as for acceptance of an initial offer.

### 13.15.3 Negotiation Business Signals

The negotiation protocol uses the AcceptanceAcknowledgment and Exception business signals as discussed in Section 13.15.2. These signals are instance documents of the business signals defined in [ebBPSS]. Examples of these instance documents are in Appendix E .

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

### 13.15.4 State Diagrams

The choreography is illustrated by the state diagram shown in Figure 3 and Figure 4.

**THE STATE DIAGRAMS NEED SOME CORRECTIONS:**

**- IN Figure 3, THE DOCUMENT NAMES IN THE CENTER DO NOT ALWAYS AGREE WITH THE NAMES IN THE INSTANCE DOCUMENT.**

**- IN Figure 4:**

- ♦ **UNLIKE Figure 3, THE DOCUMENT NAMES ARE NOT USED.**
- ♦ **THE RETURN OF THE DOUBLE-SIGNED CPA IS NOT SHOWN. PRESUMABLY IT IS ANOTHER OUTPUT FROM THE "RECEIVED FINAL CPA" STATE.**



**Figure 3, State Diagram for Initial Offer and Counter Offers**

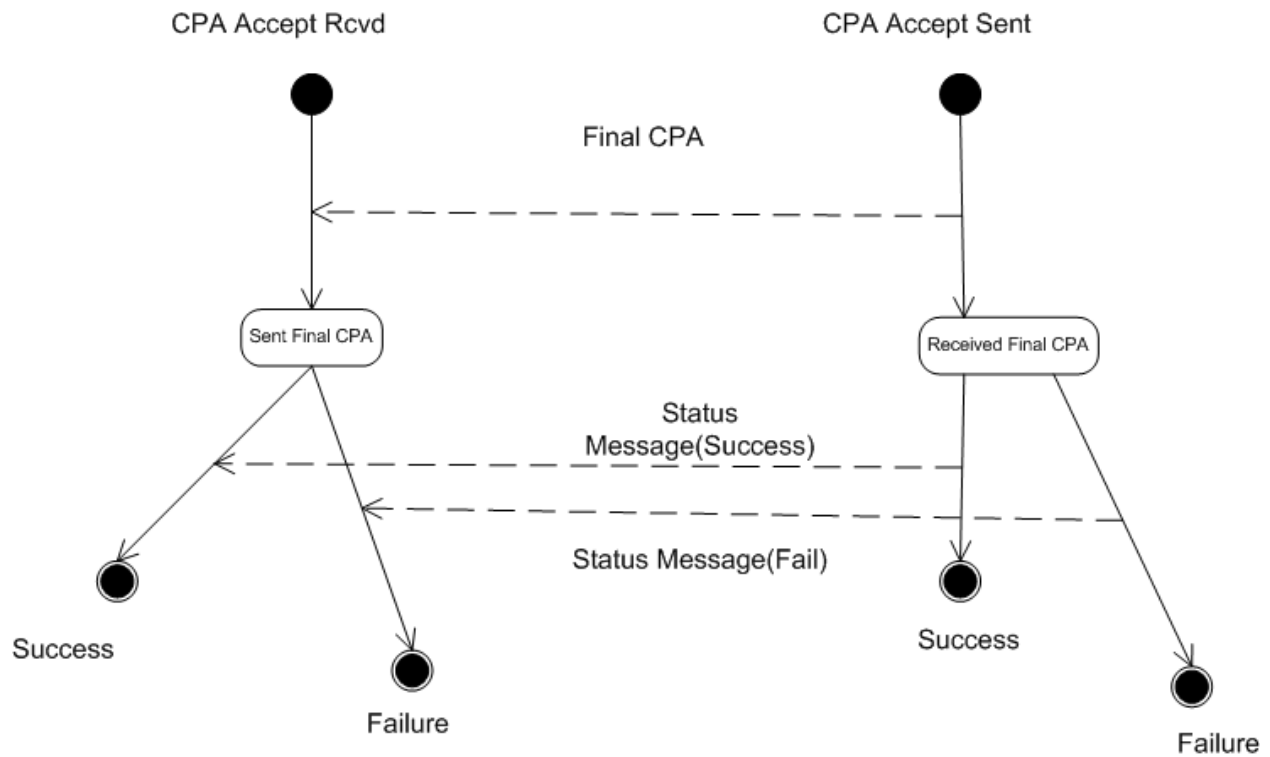


Figure 4, State Diagram for Final Transaction

## 14 Negotiation Algorithms

The negotiation algorithm is an application (business process). It is embodied in the private process at each *Party*. Note that the BPSS instance 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.

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.

***WE NEED TO DECIDE WHAT GOES IN THIS SECTION. THE ABOVE MATERIAL IS INTERESTING BUT THE SECTION IS OF NO USE UNLESS IT DOES PRESCRIBE ASPECTS OF THE NEGOTIATION ALGORITHMS THAT ENSURE INTEROPERABILITY.***

### 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*

- [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>
- [XMLDSIG] XML Signature Syntax and Processing, Worldwide Web Consortium,  
<http://www.w3.org/TR/xmlsig-core/>
- [XMLENC] XML Encryption Syntax and Processing, Worldwide Web Consortium,  
<http://www.w3.org/TR/2002/CR-xmlenc-core-20020304/>
- [XPath] XML Path Language (XPath) Version 1.0,  
<http://www.w3.org/TR/xpath>



## 16 Conformance

***THIS IS A MINOR VARIATION ON THE STATEMENT IN THE CPPA V2 SPECIFICATION. IT NEEDS TO BE REVIEWED AND PERHAPS FURTHER CHANGES.***

In order to conform to this specification, an implementation:

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

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

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

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

### 16.1 NDD and Negotiation Messages

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

#### ***URLS TO BE SUPPLIED***

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

### 16.2 NCPA Instance Document

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

[http://www.oasis-open.org/committees/ebxml-cppa/schema/cpp-cpa-2\\_0.xsd](http://www.oasis-open.org/committees/ebxml-cppa/schema/cpp-cpa-2_0.xsd)

by using two or more validating XML Schema parsers that conform to the W3C XML Schema

specifications[XMLSCHEMA-1, XMLSCHEMA-2].

### **16.3 Negotiation BPSS Instance Document**

An implementation of a tool or service that creates or maintains negotiation BPSS instance documents SHALL be determined to be conformant by validation of the BPSS instance documents, created or modified by said tool or service, against the XML Schema[XMLSCHEMA-1] definition of the BPSS in available from

#### ***URL TO BE SUPPLIED.***

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

### **16.4 Negotiation Business Signals**

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

#### ***URL TO BE SUPPLIED.***

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

**1699 17 Disclaimer**

1700 The views and specification expressed in this document are those of the authors and are not  
1701 necessarily those of their employers. The authors and their employers specifically disclaim  
1702 responsibility for any problems arising from correct or incorrect implementation or use of this  
1703 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>

## Notices

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

Portions of this document are copyright © 2002 OASIS.

**Copyright (C) The Organization for the Advancement of Structured Information Standards [OASIS] 2002. All Rights Reserved.**

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to OASIS, except as needed for the purpose of developing OASIS specifications, in which case the procedures for copyrights defined in the OASIS Intellectual Property Rights document must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by OASIS or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and OASIS DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

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

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

OASIS has been notified of intellectual property rights claimed in regard to some or all of the 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:

1749 **Appendix B XML Schema for Negotiation Messages**  
1750 **(Normative)**

1751 The XML Schema for the negotiation messages is available in text form at:  
1752

## 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 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="CPA-Negotiation" xlink:type="simple"
xlink:href="http://www.oasis-open.org/committees/ebxml-cppa-negot/CPA_Negotiation_BPSS.xml"
tp:uuid="bpid:ebXML:CPA-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.xml#CPA Negotiation Initiator"/>
      <tp:ServiceBinding>
        <tp:Service>bpid:ebXML:CPA-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">
```



```

1813         <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
1814 tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
1815 tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
1816         <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
1817 tp:businessTransactionActivity="CPA Final BTA" tp:requestOrResponseAction="Final_CPA_BT_RespBA"/>
1818         <tp:ChannelId>asyncChannelA1</tp:ChannelId>
1819     </tp:ThisPartyActionBinding>
1820     <tp:OtherPartyActionBinding>NegoResp_FinalResponseMessageB</tp:OtherPartyActionBinding>
1821 </tp:CanSend>
1822 <!-- This receive is for receiving the response for Negotiation Offer, could be acceptance, reject or counter
1823 offer-->
1824     <tp:CanReceive>
1825         <tp:ThisPartyActionBinding tp:id="Negolnit_ABID9" tp:action="CPA_Offer_BT_RespBA"
1826 tp:packageId="Negolnit_OfferResponsePackage">
1827             <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
1828 tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
1829 tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
1830             <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
1831 tp:businessTransactionActivity="CPA Offer BTA" tp:requestOrResponseAction="CPA_Offer_BT_RespBA"/>
1832             <tp:ChannelId>asyncChannelA1</tp:ChannelId>
1833         </tp:ThisPartyActionBinding>
1834         <tp:OtherPartyActionBinding>NegoResp_ABID9</tp:OtherPartyActionBinding>
1835     </tp:CanReceive>
1836     <!-- This receive is for receiving the Final Response document in the final BTA -->
1837     <tp:CanReceive>
1838         <tp:ThisPartyActionBinding tp:id="Negolnit_FinalResponseA" tp:action="Final_CPA_BT_ReqBA"
1839 tp:packageId="Negolnit_FinalMessage">
1840             <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
1841 tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
1842 tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
1843             <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
1844 tp:businessTransactionActivity="CPA Final BTA" tp:requestOrResponseAction="Final_CPA_BT_ReqBA"/>
1845             <tp:ChannelId>asyncChannelA1</tp:ChannelId>
1846         </tp:ThisPartyActionBinding>
1847         <tp:OtherPartyActionBinding>NegoResp_FinalResponseB</tp:OtherPartyActionBinding>
1848     </tp:CanReceive>
1849     <!-- This Receive is for receiving the Receipt Acknowledgment -->
1850     <tp:CanReceive>
1851         <tp:ThisPartyActionBinding tp:id="Negolnit_ABID3" tp:action="ReceiptAcknowledgment"
1852 tp:packageId="Negolnit_ReceiptAcknowledgmentPackage">
1853             <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="true"
1854 tp:isNonRepudiationReceiptRequired="true" tp:isConfidential="transient" tp:isAuthenticated="persistent"
1855 tp:isTamperProof="persistent" tp:isAuthorizationRequired="true"/>
1856             <tp:ChannelId>asyncChannelA1</tp:ChannelId>
1857         </tp:ThisPartyActionBinding>
1858         <tp:OtherPartyActionBinding>NegoResp_ABID3</tp:OtherPartyActionBinding>
1859     </tp:CanReceive>
1860     <!-- This Receive is for receiving the Exception -->
1861     <tp:CanReceive>
1862         <tp:ThisPartyActionBinding tp:id="Negolnit_ABID4" tp:action="Exception"
1863 tp:packageId="Negolnit_ExceptionPackage">
1864             <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="true"
1865 tp:isNonRepudiationReceiptRequired="true" tp:isConfidential="transient" tp:isAuthenticated="persistent"
1866 tp:isTamperProof="persistent" tp:isAuthorizationRequired="true"/>
1867             <tp:ChannelId>asyncChannelA1</tp:ChannelId>
1868         </tp:ThisPartyActionBinding>
1869         <tp:OtherPartyActionBinding>NegoResp_ABID4</tp:OtherPartyActionBinding>
1870     </tp:CanReceive>
1871 </tp:ServiceBinding>
1872 </tp:CollaborationRole>
1873 <!-- This role is for Negotiation Initiator company performing the role of Negotiation Counter offer responder -->
1874 <tp:CollaborationRole>
1875     <tp:ProcessSpecification tp:version="2.0" tp:name="CPPA-Negotiation" xlink:type="simple"
1876 xlink:href="http://www.oasis-open.org/committees/ebxml-cppa-negot/CPA_Negotiation_BPSS.xml"
1877 tp:uuid="bpid:ebXML:CPPA-Negotiation"/>
1878     <tp:Role tp:name="CPA Negotiation Counter Offer Responder" xlink:type="simple" xlink:href="http://www.oasis-

```

```

1879 open.org/committees/ebxml-cppa-negot/CPA_Negotiation_BPSS.xml#CPA Negotiation Counter Offer Responder"/>
1880     <tp:ServiceBinding>
1881         <tp:Service>bpid:ebXML:CPPA-Negotiation</tp:Service>
1882         <!-- This send is for sending the Negotiation Counter Offer in "CPA Counter Offer 2 BTA"-->
1883         <tp:CanSend>
1884             <tp:ThisPartyActionBinding tp:id="Negolnit_ABID5" tp:action="CPA_Counter_Offer_BT_ReqBA"
1885 tp:packageId="Negolnit_CounterOfferRequestPackage">
1886                 <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
1887 tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
1888 tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
1889                 <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
1890 tp:businessTransactionActivity="CPA Counter Offer 2 BTA"
1891 tp:requestOrResponseAction="CPA_Counter_Offer_BT_ReqBA">
1892                     <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
1893                 </tp:ActionContext>
1894                 <tp:ChannelId>asyncChannelA1</tp:ChannelId>
1895             </tp:ThisPartyActionBinding>
1896             <tp:OtherPartyActionBinding>NegoResp_ABID5</tp:OtherPartyActionBinding>
1897         </tp:CanSend>
1898         <!-- This send is for sending the Negotiation Counter Offer Response in "CPA Counter Offer 1 BTA"-->
1899         <tp:CanSend>
1900             <tp:ThisPartyActionBinding tp:id="Negolnit_ABID6" tp:action="CPA_Counter_Offer_BT_ReqBA"
1901 tp:packageId="Negolnit_CounterOfferResponsePackage">
1902                 <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
1903 tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
1904 tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
1905                 <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
1906 tp:businessTransactionActivity="CPA Counter Offer 1 BTA"
1907 tp:requestOrResponseAction="CPA_Counter_Offer_BT_RespBA">
1908                     <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
1909                 </tp:ActionContext>
1910                 <tp:ChannelId>asyncChannelA1</tp:ChannelId>
1911             </tp:ThisPartyActionBinding>
1912             <tp:OtherPartyActionBinding>NegoResp_ABID6</tp:OtherPartyActionBinding>
1913         </tp:CanSend>
1914         <!-- This send is for sending the Final CPA Response message in CPA_Final_BTA_init_Responder"-->
1915         <tp:CanSend>
1916             <tp:ThisPartyActionBinding tp:id="NegoCOR_FinalMessageA" tp:action="Final_CPA_BT_ReqBA"
1917 tp:packageId="Negolnit_FinalMessage">
1918                 <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
1919 tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
1920 tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
1921                 <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
1922 tp:businessTransactionActivity="CPA_Final_BTA_init_Responder" tp:requestOrResponseAction="Final_CPA_BT_ReqBA">
1923                     <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
1924                 </tp:ActionContext>
1925                 <tp:ChannelId>asyncChannelA1</tp:ChannelId>
1926             </tp:ThisPartyActionBinding>
1927             <tp:OtherPartyActionBinding>NegoCOResp_FinalMessageB</tp:OtherPartyActionBinding>
1928         </tp:CanSend>
1929         <!-- This send is for sending the response to the Final CPA Response message in
1930 CPA_Final_BTA_init_Initiator"-->
1931         <tp:CanSend>
1932             <tp:ThisPartyActionBinding tp:id="NegoCOR_FinalMessageResponseA"
1933 tp:action="Final_CPA_BT_RespBA" tp:packageId="Negolnit_FinalMessage">
1934                 <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
1935 tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
1936 tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
1937                 <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
1938 tp:businessTransactionActivity="CPA_Final_BTA_init_Initiator" tp:requestOrResponseAction="Final_CPA_BT_RespBA">
1939                     <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
1940                 </tp:ActionContext>
1941                 <tp:ChannelId>asyncChannelA1</tp:ChannelId>
1942             </tp:ThisPartyActionBinding>
1943             <tp:OtherPartyActionBinding>NegoCOResp_FinalMessageResponseB</tp:OtherPartyActionBinding>
1944         </tp:CanSend>

```

```

1945      <!-- This send is for sending the Receipt Acknowledgment -->
1946      <tp:CanSend>
1947          <tp:ThisPartyActionBinding tp:id="Negolnit_ABID13" tp:action="ReceiptAcknowledgement"
1948 tp:packageId="Negolnit_ReceiptAcknowledgmentPackage">
1949              <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="true"
1950 tp:isNonRepudiationReceiptRequired="true" tp:isConfidential="transient" tp:isAuthenticated="persistent"
1951 tp:isTamperProof="persistent" tp:isAuthorizationRequired="true"/>
1952              <tp:ChannelId>asyncChannelA1</tp:ChannelId>
1953          </tp:ThisPartyActionBinding>
1954          <tp:OtherPartyActionBinding>NegoResp_ABID13</tp:OtherPartyActionBinding>
1955      </tp:CanSend>
1956      <!-- This receive is for receiving the Final CPA message in CPA_Final_BTA_init_Initiator-->
1957      <tp:CanReceive>
1958          <tp:ThisPartyActionBinding tp:id="NegoCOR_FinalMessageA1" tp:action="Final_CPA_BT_ReqBA"
1959 tp:packageId="Negolnit_FinalMessage">
1960              <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
1961 tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
1962 tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
1963              <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
1964 tp:businessTransactionActivity="CPA_Final_BTA_init_Initiator" tp:requestOrResponseAction="Final_CPA_BT_ReqBA">
1965                  <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
1966              </tp:ActionContext>
1967              <tp:ChannelId>asyncChannelA1</tp:ChannelId>
1968          </tp:ThisPartyActionBinding>
1969          <tp:OtherPartyActionBinding>NegoCOResp_FinalMessageB1</tp:OtherPartyActionBinding>
1970      </tp:CanReceive>
1971      <!-- This receive is for receiving the response to the Final CPA message in
1972 CPA_Final_BTA_init_Responder-->
1973      <tp:CanReceive>
1974          <tp:ThisPartyActionBinding tp:id="NegoCOR_FinalMessageResponseA2"
1975 tp:action="Final_CPA_BT_RespBA" tp:packageId="Negolnit_FinalMessage">
1976              <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
1977 tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
1978 tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
1979              <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
1980 tp:businessTransactionActivity="CPA_Final_BTA_init_Responder"
1981 tp:requestOrResponseAction="Final_CPA_BT_RespBA">
1982                  <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
1983              </tp:ActionContext>
1984              <tp:ChannelId>asyncChannelA1</tp:ChannelId>
1985          </tp:ThisPartyActionBinding>
1986          <tp:OtherPartyActionBinding>NegoCOResp_FinalMessageResponseB2</tp:OtherPartyActionBinding>
1987      </tp:CanReceive>
1988      <!-- This receive is for receiving the response forNegotiation Counter Offer, could be accept, reject or again
1989 send a counter offer This happens in "CPA Counter Offer 2 BTA"-->
1990      <tp:CanReceive>
1991          <tp:ThisPartyActionBinding tp:id="Negolnit_ABID10" tp:action="CPA_Counter_Offer_BT_RespBA"
1992 tp:packageId="Negolnit_CounterOfferResponsePackage">
1993              <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
1994 tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
1995 tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
1996              <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
1997 tp:businessTransactionActivity="CPA Counter Offer 2 BTA"
1998 tp:requestOrResponseAction="CPA_Counter_Offer_BT_RespBA">
1999                  <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
2000              </tp:ActionContext>
2001              <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2002          </tp:ThisPartyActionBinding>
2003          <tp:OtherPartyActionBinding>NegoResp_ABID10</tp:OtherPartyActionBinding>
2004      </tp:CanReceive>
2005      <!-- This receive is for receiving the Negotiation Counter Offer. This happens in "CPA Counter Offer 1 BTA"--
2006 >
2007      <tp:CanReceive>
2008          <tp:ThisPartyActionBinding tp:id="Negolnit_ABID12" tp:action="CPA_Counter_Offer_BT_RespBA"
2009 tp:packageId="Negolnit_CounterOfferRequestPackage">
2010              <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"

```

```

2011 tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2012 tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2013 <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2014 tp:businessTransactionActivity="CPA Counter Offer 1 BTA"
2015 tp:requestOrResponseAction="CPA_Counter_Offer_BT_ReqBA">
2016 <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
2017 </tp:ActionContext>
2018 <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2019 </tp:ThisPartyActionBinding>
2020 <tp:OtherPartyActionBinding>NegoResp_ABID12</tp:OtherPartyActionBinding>
2021 </tp:CanReceive>
2022 <!-- This Receive is for receiving the Receipt Acknowledgment -->
2023 <tp:CanReceive>
2024 <tp:ThisPartyActionBinding tp:id="NegoInit_ABID7" tp:action="ReceiptAcknowledgment"
2025 tp:packageId="NegoInit_ReceiptAcknowledgmentPackage">
2026 <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="true"
2027 tp:isNonRepudiationReceiptRequired="true" tp:isConfidential="transient" tp:isAuthenticated="persistent"
2028 tp:isTamperProof="persistent" tp:isAuthorizationRequired="true"/>
2029 <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2030 </tp:ThisPartyActionBinding>
2031 <tp:OtherPartyActionBinding>NegoResp_ABID7</tp:OtherPartyActionBinding>
2032 </tp:CanReceive>
2033 <!-- This Receive is for receiving the Exception -->
2034 <tp:CanReceive>
2035 <tp:ThisPartyActionBinding tp:id="NegoInit_ABID8" tp:action="Exception"
2036 tp:packageId="NegoInit_ExceptionPackage">
2037 <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="true"
2038 tp:isNonRepudiationReceiptRequired="true" tp:isConfidential="transient" tp:isAuthenticated="persistent"
2039 tp:isTamperProof="persistent" tp:isAuthorizationRequired="true"/>
2040 <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2041 </tp:ThisPartyActionBinding>
2042 <tp:OtherPartyActionBinding>NegoResp_ABID8</tp:OtherPartyActionBinding>
2043 </tp:CanReceive>
2044 </tp:ServiceBinding>
2045 </tp:CollaborationRole>
2046 <!-- Certificates used by the "Negotiation Initiator" company -->
2047 <tp:Certificate tp:certId="NegoInit_AppCert">
2048 <ds:KeyInfo>
2049 <ds:KeyName>NegoInit_AppCert_Key</ds:KeyName>
2050 </ds:KeyInfo>
2051 </tp:Certificate>
2052 <tp:SecurityDetails tp:securityId="NegoInit_MessageSecurity">
2053 <tp:TrustAnchors>
2054 <tp:AnchorCertificateRef tp:certId="NegoInit_AppCert"/>
2055 </tp:TrustAnchors>
2056 </tp:SecurityDetails>
2057 <tp:DeliveryChannel tp:channelId="asyncChannelA1" tp:transportId="transportA1"
2058 tp:docExchangeId="docExchangeA1">
2059 <tp:MessagingCharacteristics tp:syncReplyMode="none" tp:ackRequested="always"
2060 tp:ackSignatureRequested="always" tp:duplicateElimination="always"/>
2061 </tp:DeliveryChannel>
2062 <tp:Transport tp:transportId="transportA1">
2063 <tp:TransportSender>
2064 <tp:TransportProtocol tp:version="1.1">HTTP</tp:TransportProtocol>
2065 <tp:AccessAuthentication>basic</tp:AccessAuthentication>
2066 </tp:TransportSender>
2067 <tp:TransportReceiver>
2068 <tp:TransportProtocol tp:version="1.1">HTTP</tp:TransportProtocol>
2069 <tp:AccessAuthentication>basic</tp:AccessAuthentication>
2070 <tp:Endpoint tp:uri="https://www.NegoInit.com/servlets/ebxmlhandler/async" tp:type="allPurpose"/>
2071 </tp:TransportReceiver>
2072 </tp:Transport>
2073 <tp:DocExchange tp:docExchangeId="docExchangeA1">
2074 <tp:ebXMLSenderBinding tp:version="2.0"/>
2075 <tp:ebXMLReceiverBinding tp:version="2.0"/>
2076 </tp:DocExchange>

```



```

2077     </tp:PartyInfo>
2078     <!-- Party info for Negotiation Responder -->
2079     <tp:PartyInfo tp:partyName="NegotiationResponder" tp:defaultMshChannelId="asyncChannelB1"
2080 tp:defaultMshPackageId="Negolnit_MshSignalPackage">
2081         <tp:PartyId tp:type="urn:oasis:names:tc:ebxml-cppa:partyid-type:duns">123456789</tp:PartyId>
2082         <tp:PartyRef xlink:href="http://NegoResp.com/about.html"/>
2083         <!-- This role is for Negotiation Responder performing the role of Negotiation Responder -->
2084         <tp:CollaborationRole>
2085             <tp:ProcessSpecification tp:version="2.0" tp:name="CPPA-Negotiation" xlink:type="simple"
2086 xlink:href="http://www.oasis-open.org/committees/ebxml-cppa-negot/CPA_Negotiation_BPSS.xml"
2087 tp:uuid="bpid:ebXML:CPPA-Negotiation"/>
2088             <tp:Role tp:name="CPA Negotiation Responder" xlink:type="simple" xlink:href="http://www.oasis-
2089 open.org/committees/ebxml-cppa-negot/CPA_Negotiation_BPSS.xml#CPA Negotiation Responder"/>
2090             <tp:ServiceBinding>
2091                 <tp:Service>bpid:ebXML:CPPA-Negotiation</tp:Service>
2092                 <!-- This send is for sending the Negotiation Offer Response, this could be accept, pending, response-->
2093                 <tp:CanSend>
2094                     <tp:ThisPartyActionBinding tp:id="NegoResp_ABID9" tp:action="CPA_Offer_BT_RespBA"
2095 tp:packageId="Negolnit_OfferResponsePackage">
2096                         <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
2097 tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2098 tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2099                         <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2100 tp:businessTransactionActivity="CPA Offer BTA" tp:requestOrResponseAction="CPA_Offer_BT_RespBA"/>
2101                         <tp:ChannelId>asyncChannelB1</tp:ChannelId>
2102                     </tp:ThisPartyActionBinding>
2103                     <tp:OtherPartyActionBinding>Negolnit_ABID9</tp:OtherPartyActionBinding>
2104                 </tp:CanSend>
2105                 <!-- This send is for sending the Final Response document in the final BTA -->
2106                 <tp:CanSend>
2107                     <tp:ThisPartyActionBinding tp:id="NegoResp_FinalResponseB" tp:action="Final_CPA_BT_ReqBA"
2108 tp:packageId="Negolnit_FinalMessage">
2109                         <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
2110 tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2111 tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2112                         <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2113 tp:businessTransactionActivity="CPA Final BTA" tp:requestOrResponseAction="Final_CPA_BT_ReqBA"/>
2114                         <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2115                     </tp:ThisPartyActionBinding>
2116                     <tp:OtherPartyActionBinding>Negolnit_FinalResponseA</tp:OtherPartyActionBinding>
2117                 </tp:CanSend>
2118                 <!-- This send is for sending the Receipt Acknowledgment -->
2119                 <tp:CanSend>
2120                     <tp:ThisPartyActionBinding tp:id="NegoResp_ABID3" tp:action="ReceiptAcknowledgement"
2121 tp:packageId="Negolnit_ReceiptAcknowledgmentPackage">
2122                         <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="true"
2123 tp:isNonRepudiationReceiptRequired="true" tp:isConfidential="transient" tp:isAuthenticated="persistent"
2124 tp:isTamperProof="persistent" tp:isAuthorizationRequired="true"/>
2125                         <tp:ChannelId>asyncChannelB1</tp:ChannelId>
2126                     </tp:ThisPartyActionBinding>
2127                     <tp:OtherPartyActionBinding>Negolnit_ABID3</tp:OtherPartyActionBinding>
2128                 </tp:CanSend>
2129                 <!-- This receive is for receiving the Final message in the collaboration. This would be the double signed
2130 CPA document or acceptance or reject of the CPA in the final Response document-->
2131                 <tp:CanReceive>
2132                     <tp:ThisPartyActionBinding tp:id="NegoResp_FinalResponseMessageB"
2133 tp:action="Final_CPA_BT_RespBA" tp:packageId="Negolnit_FinalMessage">
2134                         <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
2135 tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2136 tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2137                         <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2138 tp:businessTransactionActivity="CPA Final BTA" tp:requestOrResponseAction="Final_CPA_BT_RespBA"/>
2139                         <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2140                     </tp:ThisPartyActionBinding>
2141                     <tp:OtherPartyActionBinding>Negolnit_FinalResponseMessageA</tp:OtherPartyActionBinding>
2142                 </tp:CanReceive>

```

```

2143      <!-- This receive is for receiving the offer in the first place -->
2144      <tp:CanReceive>
2145          <tp:ThisPartyActionBinding tp:id="NegoResp_ABID1" tp:action="CPA_Offer_BT_ReqBA"
2146 tp:packageId="Negolnit_OfferRequestPackage">
2147              <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
2148 tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2149 tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2150              <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2151 tp:businessTransactionActivity="CPA Offer BTA" tp:requestOrResponseAction="CPA_Offer_BT_ReqBA"/>
2152              <tp:ChannelId>asyncChannelB1</tp:ChannelId>
2153          </tp:ThisPartyActionBinding>
2154          <tp:OtherPartyActionBinding>Negolnit_ABID1</tp:OtherPartyActionBinding>
2155      </tp:CanReceive>
2156      <!-- This Receive is for receiving the Receipt Acknowledgment -->
2157      <tp:CanReceive>
2158          <tp:ThisPartyActionBinding tp:id="NegoResp_ABID2" tp:action="ReceiptAcknowledgment"
2159 tp:packageId="Negolnit_ReceiptAcknowledgmentPackage">
2160              <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="true"
2161 tp:isNonRepudiationReceiptRequired="true" tp:isConfidential="transient" tp:isAuthenticated="persistent"
2162 tp:isTamperProof="persistent" tp:isAuthorizationRequired="true"/>
2163              <tp:ChannelId>asyncChannelB1</tp:ChannelId>
2164          </tp:ThisPartyActionBinding>
2165          <tp:OtherPartyActionBinding>Negolnit_ABID2</tp:OtherPartyActionBinding>
2166      </tp:CanReceive>
2167      <!-- This Receive is for receiving the Exception -->
2168      <tp:CanReceive>
2169          <tp:ThisPartyActionBinding tp:id="NegoResp_ABID4" tp:action="Exception"
2170 tp:packageId="Negolnit_ExceptionPackage">
2171              <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="true"
2172 tp:isNonRepudiationReceiptRequired="true" tp:isConfidential="transient" tp:isAuthenticated="persistent"
2173 tp:isTamperProof="persistent" tp:isAuthorizationRequired="true"/>
2174              <tp:ChannelId>asyncChannelB1</tp:ChannelId>
2175          </tp:ThisPartyActionBinding>
2176          <tp:OtherPartyActionBinding>Negolnit_ABID4</tp:OtherPartyActionBinding>
2177      </tp:CanReceive>
2178      </tp:ServiceBinding>
2179      </tp:CollaborationRole>
2180      <!-- This role is for Negotiation Responder company performing the role of Negotiation Counter offer initiator -->
2181      <tp:CollaborationRole>
2182          <tp:ProcessSpecification tp:version="2.0" tp:name="CPA-Negotiation" xlink:type="simple"
2183 xlink:href="http://www.oasis-open.org/committees/ebxml-cppa-negot/CPA_Negotiation_BPSS.xml"
2184 tp:uuid="bpid:ebXML:CPA-Negotiation"/>
2185          <tp:Role tp:name="CPA Negotiation Counter Offer Initiator" xlink:type="simple" xlink:href="http://www.oasis-
2186 open.org/committees/ebxml-cppa-negot/CPA_Negotiation_BPSS.xml#CPA Negotiation Counter Offer Initiator"/>
2187          <tp:ServiceBinding>
2188              <tp:Service>bpid:ebXML:CPA-Negotiation</tp:Service>
2189          <!-- This send is for sending the Negotiation Counter Offer. This happens in "CPA Counter Offer 1 BTA" -->
2190          <tp:CanSend>
2191              <tp:ThisPartyActionBinding tp:id="NegoResp_ABID12" tp:action="CPA_Counter_Offer_BT_ReqBA"
2192 tp:packageId="Negolnit_CounterOfferRequestPackage">
2193                  <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
2194 tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2195 tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2196                  <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2197 tp:businessTransactionActivity="CPA Counter Offer 1 BTA"
2198 tp:requestOrResponseAction="CPA_Counter_Offer_BT_ReqBA">
2199                      <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
2200                  </tp:ActionContext>
2201                  <tp:ChannelId>asyncChannelB1</tp:ChannelId>
2202              </tp:ThisPartyActionBinding>
2203              <tp:OtherPartyActionBinding>Negolnit_ABID12</tp:OtherPartyActionBinding>
2204          </tp:CanSend>
2205          <!-- This send is for sending the Negotiation Counter Offer response. This happens in "CPA Counter Offer 2
2206 BTA" -->
2207          <tp:CanSend>
2208              <tp:ThisPartyActionBinding tp:id="NegoResp_ABID10" tp:action="CPA_Counter_Offer_BT_ReqBA"

```

```

2209 tp:packageId="Negolnit_CounterOfferResponsePackage">
2210     <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
2211 tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2212 tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2213     <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2214 tp:businessTransactionActivity="CPA Counter Offer 2 BTA"
2215 tp:requestOrResponseAction="CPA_Counter_Offer_BT_RespBA">
2216         <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
2217     </tp:ActionContext>
2218     <tp:ChannelId>asyncChannelB1</tp:ChannelId>
2219 </tp:ThisPartyActionBinding>
2220 <tp:OtherPartyActionBinding>Negolnit_ABID10</tp:OtherPartyActionBinding>
2221 </tp:CanSend>
2222 <!-- This send is for sending the Receipt Acknowledgment -->
2223 <tp:CanSend>
2224     <tp:ThisPartyActionBinding tp:id="NegoResp_ABID7" tp:action="ReceiptAcknowledgement"
2225 tp:packageId="Negolnit_ReceiptAcknowledgmentPackage">
2226     <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="true"
2227 tp:isNonRepudiationReceiptRequired="true" tp:isConfidential="transient" tp:isAuthenticated="persistent"
2228 tp:isTamperProof="persistent" tp:isAuthorizationRequired="true"/>
2229     <tp:ChannelId>asyncChannelB1</tp:ChannelId>
2230 </tp:ThisPartyActionBinding>
2231 <tp:OtherPartyActionBinding>Negolnit_ABID7</tp:OtherPartyActionBinding>
2232 </tp:CanSend>
2233 <!-- This send is for sending the Final CPA message in CPA_Final_BTA_init_Initiator-->
2234 <tp:CanSend>
2235     <tp:ThisPartyActionBinding tp:id="NegoCOResp_FinalMessageB1" tp:action="Final_CPA_BT_ReqBA"
2236 tp:packageId="Negolnit_FinalMessage">
2237     <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
2238 tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2239 tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2240     <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2241 tp:businessTransactionActivity="CPA_Final_BTA_init_Initiator" tp:requestOrResponseAction="Final_CPA_BT_ReqBA">
2242         <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
2243     </tp:ActionContext>
2244     <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2245 </tp:ThisPartyActionBinding>
2246 <tp:OtherPartyActionBinding>NegoCOR_FinalMessageA1</tp:OtherPartyActionBinding>
2247 </tp:CanSend>
2248 <!-- This send is for sending the response to the Final CPA message in CPA_Final_BTA_init_Responder-->
2249 <tp:CanSend>
2250     <tp:ThisPartyActionBinding tp:id="NegoCOResp_FinalMessageResponseB2"
2251 tp:action="Final_CPA_BT_RespBA" tp:packageId="Negolnit_FinalMessage">
2252     <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
2253 tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2254 tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2255     <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2256 tp:businessTransactionActivity="CPA_Final_BTA_init_Responder"
2257 tp:requestOrResponseAction="Final_CPA_BT_RespBA">
2258         <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
2259     </tp:ActionContext>
2260     <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2261 </tp:ThisPartyActionBinding>
2262 <tp:OtherPartyActionBinding>NegoCOR_FinalMessageResponseA2</tp:OtherPartyActionBinding>
2263 </tp:CanSend>
2264 <!-- This receive is for receiving the response forNegotiation Counter Offer, could be accept, reject or again
2265 send a counter offer This happens in "CPA Counter Offer 1 BTA"-->
2266 <tp:CanReceive>
2267     <tp:ThisPartyActionBinding tp:id="NegoResp_ABID6" tp:action="CPA_Counter_Offer_BT_RespBA"
2268 tp:packageId="Negolnit_CounterOfferResponsePackage">
2269     <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
2270 tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2271 tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2272     <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2273 tp:businessTransactionActivity="CPA Counter Offer 1 BTA"
2274 tp:requestOrResponseAction="CPA_Counter_Offer_BT_RespBA">

```

```

2275         <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
2276     </tp:ActionContext>
2277     <tp:ChannelId>asyncChannelB1</tp:ChannelId>
2278 </tp:ThisPartyActionBinding>
2279 <tp:OtherPartyActionBinding>Negolnit_ABID6</tp:OtherPartyActionBinding>
2280 </tp:CanReceive>
2281 <!-- This receive is for receiving Negotiation Counter Offer.This happens in "CPA Counter Offer 2 BTA"-->
2282 <tp:CanReceive>
2283     <tp:ThisPartyActionBinding tp:id="NegoResp_ABID5" tp:action="CPA_Counter_Offer_BT_ReqBA"
2284 tp:packageId="Negolnit_CounterOfferRequestPackage">
2285         <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
2286 tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2287 tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2288         <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2289 tp:businessTransactionActivity="CPA Counter Offer 2 BTA"
2290 tp:requestOrResponseAction="CPA_Counter_Offer_BT_ReqBA">
2291             <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
2292         </tp:ActionContext>
2293         <tp:ChannelId>asyncChannelB1</tp:ChannelId>
2294 </tp:ThisPartyActionBinding>
2295 <tp:OtherPartyActionBinding>Negolnit_ABID5</tp:OtherPartyActionBinding>
2296 </tp:CanReceive>
2297 <!-- This receive is for receiving the Final CPA message in CPA_Final_BTA_init_Responder"-->
2298 <tp:CanReceive>
2299     <tp:ThisPartyActionBinding tp:id="NegoCOResp_FinalMessageB" tp:action="Final_CPA_BT_ReqBA"
2300 tp:packageId="Negolnit_FinalMessage">
2301         <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
2302 tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2303 tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2304         <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2305 tp:businessTransactionActivity="CPA_Final_BTA_init_Responder" tp:requestOrResponseAction="Final_CPA_BT_ReqBA">
2306             <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
2307         </tp:ActionContext>
2308         <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2309 </tp:ThisPartyActionBinding>
2310 <tp:OtherPartyActionBinding>NegoCOR_FinalMessageA</tp:OtherPartyActionBinding>
2311 </tp:CanReceive>
2312 <!-- This receive is for receiving the response to the Final CPA message in CPA_Final_BTA_init_Initiator"-->
2313 <tp:CanReceive>
2314     <tp:ThisPartyActionBinding tp:id="NegoCOResp_FinalMessageResponseB"
2315 tp:action="Final_CPA_BT_RespBA" tp:packageId="Negolnit_FinalMessage">
2316         <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="false"
2317 tp:isNonRepudiationReceiptRequired="false" tp:isConfidential="none" tp:isAuthenticated="none" tp:isTamperProof="none"
2318 tp:isAuthorizationRequired="false" tp:timeToAcknowledgeReceipt="PT2H" tp:timeToPerform="P1D"/>
2319         <tp:ActionContext tp:binaryCollaboration="CPA Negotiation BC"
2320 tp:businessTransactionActivity="CPA_Final_BTA_init_Initiator" tp:requestOrResponseAction="Final_CPA_BT_RespBA">
2321             <tp:CollaborationActivity tp:name="CPA Counter Offer CA"/>
2322         </tp:ActionContext>
2323         <tp:ChannelId>asyncChannelA1</tp:ChannelId>
2324 </tp:ThisPartyActionBinding>
2325 <tp:OtherPartyActionBinding>NegoCOR_FinalMessageResponseA</tp:OtherPartyActionBinding>
2326 </tp:CanReceive>
2327 <!-- This Receive is for receiving the Receipt Acknowledgment -->
2328 <tp:CanReceive>
2329     <tp:ThisPartyActionBinding tp:id="NegoResp_ABID13" tp:action="ReceiptAcknowledgment"
2330 tp:packageId="Negolnit_ReceiptAcknowledgmentPackage">
2331         <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="true"
2332 tp:isNonRepudiationReceiptRequired="true" tp:isConfidential="transient" tp:isAuthenticated="persistent"
2333 tp:isTamperProof="persistent" tp:isAuthorizationRequired="true"/>
2334         <tp:ChannelId>asyncChannelB1</tp:ChannelId>
2335 </tp:ThisPartyActionBinding>
2336 <tp:OtherPartyActionBinding>Negolnit_ABID13</tp:OtherPartyActionBinding>
2337 </tp:CanReceive>
2338 <!-- This Receive is for receiving the Exception -->
2339 <tp:CanReceive>
2340     <tp:ThisPartyActionBinding tp:id="NegoResp_ABID8" tp:action="Exception"

```



```

2341 tp:packageId="Negolnit_ExceptionPackage">
2342     <tp:BusinessTransactionCharacteristics tp:isNonRepudiationRequired="true"
2343 tp:isNonRepudiationReceiptRequired="true" tp:isConfidential="transient" tp:isAuthenticated="persistent"
2344 tp:isTamperProof="persistent" tp:isAuthorizationRequired="true"/>
2345     <tp:ChannelId>asyncChannelB1</tp:ChannelId>
2346     </tp:ThisPartyActionBinding>
2347     <tp:OtherPartyActionBinding>Negolnit_ABID8</tp:OtherPartyActionBinding>
2348     </tp:CanReceive>
2349     </tp:ServiceBinding>
2350 </tp:CollaborationRole>
2351 <!-- Certificates used by the "Negotiation Initiator" company -->
2352 <tp:Certificate tp:certId="NegoResp_AppCert">
2353     <ds:KeyInfo>
2354         <ds:KeyName>NegoResp_AppCert_Key</ds:KeyName>
2355     </ds:KeyInfo>
2356 </tp:Certificate>
2357 <tp:SecurityDetails tp:securityId="NegoResp_MessageSecurity">
2358     <tp:TrustAnchors>
2359         <tp:AnchorCertificateRef tp:certId="NegoResp_AppCert"/>
2360     </tp:TrustAnchors>
2361 </tp:SecurityDetails>
2362 <tp:DeliveryChannel tp:channelId="asyncChannelB1" tp:transportId="transportB1"
2363 tp:docExchangeId="docExchangeB1">
2364     <tp:MessagingCharacteristics tp:syncReplyMode="none" tp:ackRequested="always"
2365 tp:ackSignatureRequested="always" tp:duplicateElimination="always"/>
2366 </tp:DeliveryChannel>
2367 <tp:Transport tp:transportId="transportB1">
2368     <tp:TransportSender>
2369         <tp:TransportProtocol tp:version="1.1">HTTP</tp:TransportProtocol>
2370         <tp:AccessAuthentication>basic</tp:AccessAuthentication>
2371     </tp:TransportSender>
2372     <tp:TransportReceiver>
2373         <tp:TransportProtocol tp:version="1.1">HTTP</tp:TransportProtocol>
2374         <tp:AccessAuthentication>basic</tp:AccessAuthentication>
2375         <tp:Endpoint tp:uri="https://www.NegoResp.com/servlets/ebxmlhandler/async" tp:type="allPurpose"/>
2376     </tp:TransportReceiver>
2377 </tp:Transport>
2378 <tp:DocExchange tp:docExchangeId="docExchangeB1">
2379     <tp:ebXMLSenderBinding tp:version="2.0"/>
2380     <tp:ebXMLReceiverBinding tp:version="2.0"/>
2381 </tp:DocExchange>
2382 </tp:PartyInfo>
2383 <!-- SimplePart corresponding to the SOAP Envelope -->
2384 <tp:SimplePart tp:id="Negolnit_MsgHdr" tp:mimetype="text/xml">
2385     <tp:NamespaceSupported tp:location="http://www.oasis-open.org/committees/ebxml-msg/schema/msg-header-
2386 2_0.xsd" tp:version="2.0">
2387         http://www.oasis-open.org/committees/ebxml-msg/schema/msg-header-2_0.xsd
2388     </tp:NamespaceSupported>
2389 </tp:SimplePart>
2390 <tp:SimplePart tp:id="NegoResp_MsgHdr" tp:mimetype="text/xml">
2391     <tp:NamespaceSupported tp:location="http://www.oasis-open.org/committees/ebxml-msg/schema/msg-header-
2392 2_0.xsd" tp:version="2.0">
2393         http://www.oasis-open.org/committees/ebxml-msg/schema/msg-header-2_0.xsd
2394     </tp:NamespaceSupported>
2395 </tp:SimplePart>
2396 <!-- SimplePart corresponding to a Receipt Acknowledgment business signal -->
2397 <tp:SimplePart tp:id="Negolnit_ReceiptAcknowledgment" tp:mimetype="application/xml">
2398     <tp:NamespaceSupported tp:location="http://www.ebxml.org/bpss/ReceiptAcknowledgment.xsd"
2399 tp:version="2.0">http://www.ebxml.org/bpss/ReceiptAcknowledgment.xsd
2400 </tp:NamespaceSupported>
2401 </tp:SimplePart>
2402 <tp:SimplePart tp:id="NegoResp_ReceiptAcknowledgment" tp:mimetype="application/xml">
2403     <tp:NamespaceSupported tp:location="http://www.ebxml.org/bpss/ReceiptAcknowledgment.xsd" tp:version="2.0">
2404         http://www.ebxml.org/bpss/ReceiptAcknowledgment.xsd
2405     </tp:NamespaceSupported>
2406 </tp:SimplePart>

```

```

2407      <!-- SimplePart corresponding to an Exception business signal -->
2408      <tp:SimplePart tp:id="Negolnit_Exception" tp:mimetype="application/xml">
2409        <tp:NamespaceSupported tp:location="http://www.oasis-open.org/committees/ebxml-msg/schema/msg-header-
2410 2_0.xsd" tp:version="2.0">
2411          http://www.ebxml.org/bpss/Exception.xsd
2412        </tp:NamespaceSupported>
2413      </tp:SimplePart>
2414      <tp:SimplePart tp:id="NegoResp_Exception" tp:mimetype="application/xml">
2415        <tp:NamespaceSupported tp:location="http://www.oasis-open.org/committees/ebxml-msg/schema/msg-header-
2416 2_0.xsd" tp:version="2.0">
2417          http://www.ebxml.org/bpss/Exception.xsd
2418        </tp:NamespaceSupported>
2419      </tp:SimplePart>
2420      <!-- SimplePart corresponding to a negotiation offer request action -->
2421      <tp:SimplePart tp:id="Negolnit_OfferRequest" tp:mimetype="application/xml">
2422        <tp:NamespaceSupported tp:location="http://www.ebxml.org/schemas/NegotiationOffer.xsd" tp:version="1.0">
2423          http://www.ebxml.org/schemas/NegotiationOffer.xsd
2424        </tp:NamespaceSupported>
2425      </tp:SimplePart>
2426      <!-- SimplePart corresponding to a Negotiation offer response action (accept) -->
2427      <tp:SimplePart tp:id="Negolnit_OfferAccept" tp:mimetype="application/xml">
2428        <tp:NamespaceSupported tp:location="http://www.ebxml.org/schemas/OfferAccept.xsd" tp:version="1.0">
2429          http://www.ebxml.org/schemas/OfferAccept.xsd
2430        </tp:NamespaceSupported>
2431      </tp:SimplePart>
2432      <!-- SimplePart corresponding to a Negotiation offer response action (reject) -->
2433      <tp:SimplePart tp:id="Negolnit_OfferReject" tp:mimetype="application/xml">
2434        <tp:NamespaceSupported tp:location="http://www.ebxml.org/schemas/OfferReject.xsd" tp:version="1.0">
2435          http://www.ebxml.org/schemas/OfferReject.xsd
2436        </tp:NamespaceSupported>
2437      </tp:SimplePart>
2438      <!-- SimplePart corresponding to a Negotiation offer response action (counter pending) -->
2439      <tp:SimplePart tp:id="Negolnit_OfferCounterPending" tp:mimetype="application/xml">
2440        <tp:NamespaceSupported tp:location="http://www.ebxml.org/schemas/OfferCounterPending.xsd" tp:version="1.0">
2441          http://www.ebxml.org/schemas/OfferCounterPending.xsd
2442        </tp:NamespaceSupported>
2443      </tp:SimplePart>
2444      <!-- SimplePart corresponding to a Negotiation Counter offer request action -->
2445      <tp:SimplePart tp:id="Negolnit_CounterOfferRequest" tp:mimetype="application/xml">
2446        <tp:NamespaceSupported tp:location="http://www.ebxml.org/schemas/CounterOffer.xsd" tp:version="1.0">
2447          http://www.ebxml.org/schemas/CounterOfferRequest.xsd
2448        </tp:NamespaceSupported>
2449      </tp:SimplePart>
2450      <!-- SimplePart corresponding to a Negotiation Final document being sent in the negotiation process -->
2451      <tp:SimplePart tp:id="Negolnit_FinalMessage" tp:mimetype="application/xml">
2452        <tp:NamespaceSupported tp:location="http://www.ebxml.org/schemas/FinalMessage.xsd" tp:version="1.0">
2453          http://www.ebxml.org/schemas/FinalMessage.xsd
2454        </tp:NamespaceSupported>
2455      </tp:SimplePart>
2456      <!-- SimplePart corresponding to a Negotiation Counter offer request action -->
2457      <tp:SimplePart tp:id="Negolnit_FinalMessageResponse" tp:mimetype="application/xml">
2458        <tp:NamespaceSupported tp:location="http://www.ebxml.org/schemas/FinalMessageResponse.xsd"
2459 tp:version="1.0">
2460          http://www.ebxml.org/schemas/FinalMessageResponse.xsd
2461        </tp:NamespaceSupported>
2462      </tp:SimplePart>
2463      <!-- An ebXML message with a SOAP Envelope only -->
2464      <tp:Packaging tp:id="Negolnit_MshSignalPackage">
2465        <tp:ProcessingCapabilities tp:parse="true" tp:generate="true"/>
2466        <tp:CompositeList>
2467          <tp:Composite tp:id="Negolnit_MshSignal" tp:mimetype="multipart/related" tp:mimeparameters="type=text/xml">
2468            <tp:Constituent tp:idref="Negolnit_MsgHdr"/>
2469          </tp:Composite>
2470        </tp:CompositeList>
2471      </tp:Packaging>
2472      <!-- An ebXML message with a SOAP Envelope plus a Offer action payload -->

```

```

2473     <tp:Packaging tp:id="Negolnit_OfferRequestPackage">
2474         <tp:ProcessingCapabilities tp:parse="true" tp:generate="true"/>
2475         <tp:CompositeList>
2476             <tp:Composite tp:id="Negolnit_OfferRequestMsgId" tp:mimetype="multipart/related"
2477 tp:mimeparameters="type=text/xml">
2478                 <tp:Constituent tp:idref="Negolnit_MsgHdr"/>
2479                 <tp:Constituent tp:idref="Negolnit_OfferRequest"/>
2480             </tp:Composite>
2481         </tp:CompositeList>
2482     </tp:Packaging>
2483     <!-- An ebXML message with a SOAP Envelope plus a offer response action payload -->
2484     <tp:Packaging tp:id="Negolnit_OfferResponsePackage">
2485         <tp:ProcessingCapabilities tp:parse="true" tp:generate="true"/>
2486         <tp:CompositeList>
2487             <tp:Composite tp:id="Negolnit_OfferResponseAcceptMsgId" tp:mimetype="multipart/related"
2488 tp:mimeparameters="type=text/xml">
2489                 <tp:Constituent tp:idref="Negolnit_MsgHdr"/>
2490                 <tp:Constituent tp:idref="Negolnit_OfferAccept"/>
2491             </tp:Composite>
2492         </tp:CompositeList>
2493         <tp:CompositeList>
2494             <tp:Composite tp:id="Negolnit_OfferResponseRejectMsgId" tp:mimetype="multipart/related"
2495 tp:mimeparameters="type=text/xml">
2496                 <tp:Constituent tp:idref="Negolnit_MsgHdr"/>
2497                 <tp:Constituent tp:idref="Negolnit_OfferReject"/>
2498             </tp:Composite>
2499         </tp:CompositeList>
2500         <tp:CompositeList>
2501             <tp:Composite tp:id="Negolnit_OfferResponsePendingMsgId" tp:mimetype="multipart/related"
2502 tp:mimeparameters="type=text/xml">
2503                 <tp:Constituent tp:idref="Negolnit_MsgHdr"/>
2504                 <tp:Constituent tp:idref="Negolnit_OfferCounterPending"/>
2505             </tp:Composite>
2506         </tp:CompositeList>
2507     </tp:Packaging>
2508     <!-- An ebXML message with a SOAP Envelope plus a counter offer request action payload -->
2509     <tp:Packaging tp:id="Negolnit_CounterOfferRequestPackage">
2510         <tp:ProcessingCapabilities tp:parse="true" tp:generate="true"/>
2511         <tp:CompositeList>
2512             <tp:Composite tp:id="Negolnit_CounterOfferRequestMsgId" tp:mimetype="multipart/related"
2513 tp:mimeparameters="type=text/xml">
2514                 <tp:Constituent tp:idref="Negolnit_MsgHdr"/>
2515                 <tp:Constituent tp:idref="Negolnit_CounterOfferRequest"/>
2516             </tp:Composite>
2517         </tp:CompositeList>
2518     </tp:Packaging>
2519     <!-- An ebXML message with a SOAP Envelope plus a counter offer response action payload -->
2520     <tp:Packaging tp:id="Negolnit_CounterOfferResponsePackage">
2521         <tp:ProcessingCapabilities tp:parse="true" tp:generate="true"/>
2522         <tp:CompositeList>
2523             <tp:Composite tp:id="Negolnit_CounterOfferResponseAcceptMsgId" tp:mimetype="multipart/related"
2524 tp:mimeparameters="type=text/xml">
2525                 <tp:Constituent tp:idref="Negolnit_MsgHdr"/>
2526                 <tp:Constituent tp:idref="Negolnit_OfferAccept"/>
2527             </tp:Composite>
2528         </tp:CompositeList>
2529         <tp:CompositeList>
2530             <tp:Composite tp:id="Negolnit_CounterOfferResponseRejectMsgId" tp:mimetype="multipart/related"
2531 tp:mimeparameters="type=text/xml">
2532                 <tp:Constituent tp:idref="Negolnit_MsgHdr"/>
2533                 <tp:Constituent tp:idref="Negolnit_OfferReject"/>
2534             </tp:Composite>
2535         </tp:CompositeList>
2536         <tp:CompositeList>
2537             <tp:Composite tp:id="Negolnit_CounterOfferResponsePendingMsgId" tp:mimetype="multipart/related"
2538 tp:mimeparameters="type=text/xml">

```

```

2539         <tp:Constituent tp:idref="Negolnit_MsgHdr"/>
2540         <tp:Constituent tp:idref="Negolnit_OfferCounterPending"/>
2541     </tp:Composite>
2542 </tp:CompositeList>
2543 </tp:Packaging>
2544 <!-- An ebXML message with a SOAP Envelope plus a Receipt Acknowledgment payload -->
2545 <tp:Packaging tp:id="Negolnit_ReceiptAcknowledgmentPackage">
2546     <tp:ProcessingCapabilities tp:parse="true" tp:generate="true"/>
2547     <tp:CompositeList>
2548         <tp:Composite tp:id="Negolnit_ReceiptAcknowledgmentMsg" tp:mimetype="multipart/related"
2549 tp:mimeparameters="type=text/xml">
2550             <tp:Constituent tp:idref="Negolnit_MsgHdr"/>
2551             <tp:Constituent tp:idref="Negolnit_ReceiptAcknowledgment"/>
2552         </tp:Composite>
2553     </tp:CompositeList>
2554 </tp:Packaging>
2555 <tp:Packaging tp:id="NegoResp_ReceiptAcknowledgmentPackage">
2556     <tp:ProcessingCapabilities tp:parse="true" tp:generate="true"/>
2557     <tp:CompositeList>
2558         <tp:Composite tp:id="NegoResp_ReceiptAcknowledgmentMsg" tp:mimetype="multipart/related"
2559 tp:mimeparameters="type=text/xml">
2560             <tp:Constituent tp:idref="NegoResp_MsgHdr"/>
2561             <tp:Constituent tp:idref="NegoResp_ReceiptAcknowledgment"/>
2562         </tp:Composite>
2563     </tp:CompositeList>
2564 </tp:Packaging>
2565 <!-- An ebXML message with a SOAP Envelope plus an Exception payload -->
2566 <tp:Packaging tp:id="Negolnit_ExceptionPackage">
2567     <tp:ProcessingCapabilities tp:parse="true" tp:generate="true"/>
2568     <tp:CompositeList>
2569         <tp:Composite tp:id="Negolnit_ExceptionMsg" tp:mimetype="multipart/related"
2570 tp:mimeparameters="type=text/xml">
2571             <tp:Constituent tp:idref="Negolnit_MsgHdr"/>
2572             <tp:Constituent tp:idref="Negolnit_Exception"/>
2573         </tp:Composite>
2574     </tp:CompositeList>
2575 </tp:Packaging>
2576 <tp:Packaging tp:id="NegoResp_ExceptionPackage">
2577     <tp:ProcessingCapabilities tp:parse="true" tp:generate="true"/>
2578     <tp:CompositeList>
2579         <tp:Composite tp:id="NegoResp_ExceptionMsg" tp:mimetype="multipart/related"
2580 tp:mimeparameters="type=text/xml">
2581             <tp:Constituent tp:idref="NegoResp_MsgHdr"/>
2582             <tp:Constituent tp:idref="NegoResp_Exception"/>
2583         </tp:Composite>
2584     </tp:CompositeList>
2585 </tp:Packaging>
2586 <tp:Comment xml:lang="en-US">CPPA negotiation between Negolnit.com and NegoResp.com</tp:Comment>
2587 </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 Accept 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 signed 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"
```



```

2648 isPositiveResponse="true" isTamperProof="none"/>
2649     <DocumentEnvelope businessDocument="CPA Reject Offer Doc"
2650 businessDocumentIDRef="CPA_Reject_Offer_Doc" isAuthenticated="none" isConfidential="none"
2651 isPositiveResponse="false" isTamperProof="none"/>
2652     <DocumentEnvelope businessDocument="CPA Counter Pending Offer Doc"
2653 businessDocumentIDRef="CPA_Counter_Pending_Offer_Doc" isAuthenticated="none" isConfidential="none"
2654 isPositiveResponse="true" isTamperProof="none"/>
2655     </RespondingBusinessActivity>
2656 </BusinessTransaction>
2657 <!-- Business Transaction for sending the counter offer -->
2658 <BusinessTransaction name="CPA Counter Offer BT" nameID="CPA_Counter_Offer_BT">
2659     <RequestingBusinessActivity name="CPA_Counter_Offer_BT_ReqBA" nameID="CPA_Counter_Offer_BT_ReqBA"
2660 isAuthorizationRequired="false" isIntelligibleCheckRequired="false" isNonRepudiationReceiptRequired="false"
2661 isNonRepudiationRequired="false">
2662         <DocumentEnvelope businessDocument="CPA Counter Offer Doc"
2663 businessDocumentIDRef="CPA_Counter_Offer_Doc" isAuthenticated="none" isConfidential="none"
2664 isTamperProof="none"/>
2665         </RequestingBusinessActivity>
2666         <RespondingBusinessActivity name="CPA_Counter_Offer_BT_RespBA"
2667 nameID="CPA_Counter_Offer_BT_RespBA" isAuthorizationRequired="false" isIntelligibleCheckRequired="false"
2668 isNonRepudiationRequired="false">
2669             <DocumentEnvelope businessDocument="CPA Accept Offer Doc"
2670 businessDocumentIDRef="CPA_Accept_Offer_Doc" isAuthenticated="none" isConfidential="none"
2671 isPositiveResponse="true" isTamperProof="none"/>
2672             <DocumentEnvelope businessDocument="CPA Reject Offer Doc"
2673 businessDocumentIDRef="CPA_Reject_Offer_Doc" isAuthenticated="none" isConfidential="none"
2674 isPositiveResponse="false" isTamperProof="none"/>
2675             <DocumentEnvelope businessDocument="CPA Counter Pending Offer Doc"
2676 businessDocumentIDRef="CPA_Counter_Pending_Offer_Doc" isAuthenticated="none" isConfidential="none"
2677 isPositiveResponse="true" isTamperProof="none"/>
2678             </RespondingBusinessActivity>
2679         </BusinessTransaction>
2680 <!-- Main collaboration for negotiation business process -->
2681 <BinaryCollaboration name="CPA Negotiation BC" nameID="CPA_Negotiation_BC"
2682 initiatingRole="CPA_Negotiation_Initiator_Role">
2683     <!-- Role for initiator for negotiation process -->
2684     <Role name="CPA Negotiation Initiator" nameID="CPA_Negotiation_Initiator_Role"/>
2685     <!-- Role for initial responder of business collaboration -->
2686     <Role name="CPA Negotiation Responder" nameID="CPA_Negotiation_Responder_Role"/>
2687     <Start toBusinessState="CPA_Offer_BTA"/>
2688     <BusinessTransactionActivity name="CPA Offer BTA" nameID="CPA_Offer_BTA" businessTransaction="CPA Offer
2689 BT" businessTransactionIDRef="CPA_Offer_BT" fromRole="CPA Negotiation Initiator"
2690 fromRoleIDRef="CPA_Negotiation_Initiator_Role" toRole="CPA Negotiation Responder"
2691 toRoleIDRef="CPA_Negotiation_Responder_Role" isLegallyBinding="false" isConcurrent="false"/>
2692     <CollaborationActivity name="CPA Counter Offer CA" binaryCollaboration="CPA Negotiation Counter Offer BC"
2693 binaryCollaborationIDRef="CPA_Negotiation_CounterOfferBC" fromRole="CPA Negotiation Counter Offer Initiator"
2694 fromRoleIDRef="CPA_Negotiation_CounterOfferInitiator_Role" toRole="CPA Negotiation Counter Offer Responder"
2695 toRoleIDRef="CPA_Negotiation_CounterOfferResponder_Role" preCondition="Initiating Role for this activity corresponds to
2696 Responding Role in CPA Offer BTA"/>
2697     <BusinessTransactionActivity name="CPA Final BTA" nameID="CPA_Final_BTA" businessTransaction="CPA Final
2698 BT" businessTransactionIDRef="CPA_Final_BT" fromRole="CPA Negotiation Responder"
2699 fromRoleIDRef="CPA_Negotiation_Responder_Role" toRole="CPA Negotiation Initiator"
2700 toRoleIDRef="CPA_Negotiation_Initiator_Role" isLegallyBinding="false" isConcurrent="false"/>
2701     <!-- If final CPA BTA goes through fine, then overall collaboration is marked success -->
2702     <Success fromBusinessState="CPA Final BTA" conditionGuard="Success"/>
2703     <!-- If inner collaboration goes through fine, then overall collaboration is marked success. Inner collaboration
2704         Would have gone through the transaction that ends up with either the final CPA (Signed if needed) -->
2705     <Success fromBusinessState="CPA Counter Offer CA" conditionGuard="Success"/>
2706     <!-- If Reject offer document is sent for offer bta collaboration is marked as failure-->
2707     <Failure fromBusinessState="CPA Offer BTA" conditionGuard="BusinessFailure">
2708         <ConditionExpression expressionLanguage="DocumentEnvelopeLanguage" expression="CPA Reject Offer
2709 Doc"/>
2710     </Failure>
2711     <!-- If Final CPA BTA fails for some reason, then collaboration is marked as failure -->
2712     <Failure fromBusinessState="CPA Final BTA" conditionGuard="Failure"/>
2713     <Failure fromBusinessState="CPA Counter Offer CA" conditionGuard="Failure"/>

```

```

2714      <!-- Transition to Final CPA offer binary Transaction if the responder for main transaction accepts the initial offer -->
2715      <Transition fromBusinessState="CPA Offer BTA" toBusinessState="CPA Final BTA">
2716          <ConditionExpression expressionLanguage="DocumentEnvelopeLanguage" expression="CPA Accept Offer
2717 Doc"/>
2718      </Transition>
2719      <!-- Transition to counter offer binary collaboration if the responder for main transaction returns a counter offer
2720 pending message -->
2721      <Transition fromBusinessState="CPA Offer BTA" toBusinessState="CPA Counter Offer CA">
2722          <ConditionExpression expressionLanguage="DocumentEnvelopeLanguage" expression="CPA Counter Pending
2723 Offer Doc"/>
2724      </Transition>
2725      </BinaryCollaboration>
2726      <BinaryCollaboration name="CPA Negotiation Counter Offer BC" nameID="CPA_Negotiation_CounterOfferBC"
2727 initiatingRole="CPA_Negotiation_CounterOfferInitiator_Role">
2728          <Role name="CPA Negotiation Counter Offer Initiator" nameID="CPA_Negotiation_CounterOfferInitiator_Role"/>
2729          <Role name="CPA Negotiation Counter Offer Responder"
2730 nameID="CPA_Negotiation_CounterOfferResponder_Role"/>
2731          <!-- This collaboration starts with the negotiation process responder sending the counter offer -->
2732          <Start toBusinessState="CPA Counter Offer 1 BTA"/>
2733          <!-- This transaction activity is for negotiation process responder sending the counter offer -->
2734          <BusinessTransactionActivity name="CPA Counter Offer 1 BTA" nameID="CPA_Counter_Offer_1_BTA"
2735 businessTransaction="CPA Counter Offer BT" businessTransactionIDRef="CPA_Counter_Offer_BT" fromRole="CPA
2736 Negotiation Counter Offer Initiator" fromRoleIDRef="CPA_Negotiation_CounterOfferInitiator_Role" toRole="CPA Negotiation
2737 Counter Offer Non Initiator" toRoleIDRef="CPA_Negotiation_CounterOfferResponder_Role" isLegallyBinding="false"
2738 isConcurrent="false" postCondition="Parties reverse roles they play"/>
2739          <!-- This transaction activity is for negotiation process initiator sending the counter offer -->
2740          <BusinessTransactionActivity name="CPA Counter Offer 2 BTA" nameID="CPA_Counter_Offer_2_BTA"
2741 businessTransaction="CPA Counter Offer BT" businessTransactionIDRef="CPA_Counter_Offer_BT" fromRole="CPA
2742 Negotiation Counter Offer Responder" fromRoleIDRef="CPA_Negotiation_CounterOfferResponder_Role" toRole="CPA
2743 Negotiation Counter Offer Initiator" toRoleIDRef="CPA_Negotiation_CounterOfferInitiator_Role" isLegallyBinding="false"
2744 isConcurrent="false" postCondition="Parties reverse roles they play"/>
2745          <BusinessTransactionActivity name="CPA Final BTA Init Initiator" nameID="CPA_Final_BTA_init_Initiator"
2746 businessTransaction="CPA Final BT" businessTransactionIDRef="CPA_Final_BT" fromRole="CPA Negotiation Counter
2747 Offer Initiator" fromRoleIDRef="CPA_Negotiation_CounterOfferInitiator_Role" toRole="CPA Negotiation Counter Offer
2748 Responder" toRoleIDRef="CPA_Negotiation_CounterOfferResponder_Role" isLegallyBinding="false" isConcurrent="false"/>
2749          <BusinessTransactionActivity name="CPA Final BTA Init Responder" nameID="CPA_Final_BTA_init_Responder"
2750 businessTransaction="CPA Final BT" businessTransactionIDRef="CPA_Final_BT" fromRole="CPA Negotiation Counter
2751 Offer Responder" fromRoleIDRef="CPA_Negotiation_CounterOfferResponder_Role" toRole="CPA Negotiation Counter
2752 Offer Initiator" toRoleIDRef="CPA_Negotiation_CounterOfferInitiator_Role" isLegallyBinding="false" isConcurrent="false"/>
2753          <!-- Inner collaboration succeeds if the final BTA which involves sending final CPA succeeds -->
2754          <Success fromBusinessState="CPA Final BTA Init Initiator" conditionGuard="Success"/>
2755          <!-- Inner collaboration succeeds if the final BTA which involves sending final CPA succeeds. This is
2756 the same as above but the difference is this initiated by a different party -->
2757          <Success fromBusinessState="CPA Final BTA Init Responder" conditionGuard="Success"/>
2758          <!-- Inner collaboration fails if the final BTA which involves sending final CPA fails -->
2759          <Failure fromBusinessState="CPA Final BTA Init Initiator" conditionGuard="Failure"/>
2760          <!-- Inner collaboration fails if the final BTA which involves sending final CPA fails. This is
2761 the same as above but the difference is this initiated by a different party -->
2762          <Failure fromBusinessState="CPA Final BTA Init Responder" conditionGuard="Failure"/>
2763          <Failure fromBusinessState="CPA Counter Offer 1 BTA" conditionGuard="BusinessFailure">
2764              <ConditionExpression expressionLanguage="DocumentEnvelopeLanguage" expression="CPA Reject Offer
2765 Doc"/>
2766          </Failure>
2767          <Failure fromBusinessState="CPA Counter Offer 2 BTA" conditionGuard="BusinessFailure">
2768              <ConditionExpression expressionLanguage="DocumentEnvelopeLanguage" expression="CPA Reject Offer
2769 Doc"/>
2770          </Failure>
2771          <!-- If the negotiation process responder (initiator in this innercollaboration) sends an acceptance offer, negotiation
2772 process responder sends the final CPA -->
2773          <Transition fromBusinessState="CPA Counter Offer 2 BTA" toBusinessState="CPA Final BTA Init Initiator">
2774              <ConditionExpression expressionLanguage="DocumentEnvelopeLanguage" expression="CPA Accept Offer
2775 Doc"/>
2776          </Transition>
2777          <!-- If the negotiation process initiator (responder in this inner collaboration) sends an acceptance offer, negotiation
2778 process initiator sends the final CPA -->
2779          <Transition fromBusinessState="CPA Counter Offer 1 BTA" toBusinessState="CPA Final BTA Init Responder">

```

```
2780         <ConditionExpression expressionLanguage="DocumentEnvelopeLanguage" expression="CPA Accept Offer
2781 Doc"/>
2782     </Transition>
2783     <!-- If the negotiation process responder sends counter offer and negotiation process initiator sends a counter offer,
2784 negotiation process initiator sends the counter offer next time -->
2785     <Transition fromBusinessState="CPA Counter Offer 1 BTA" toBusinessState="CPA Counter Offer 2 BTA">
2786         <ConditionExpression expressionLanguage="DocumentEnvelopeLanguage" expression="CPA Counter Pending
2787 Offer Doc"/>
2788     </Transition>
2789     <!-- If the negotiation process initiator sends a counter offer and negotiation process responds sends a counter offer,
2790 negotiation process responder sends the counter offer next time, hence the transition back to original BTA-->
2791     <Transition fromBusinessState="CPA Counter Offer 2 BTA" toBusinessState="CPA Counter Offer 1 BTA">
2792         <ConditionExpression expressionLanguage="DocumentEnvelopeLanguage" expression="CPA Counter Pending
2793 Offer Doc"/>
2794     </Transition>
2795 </BinaryCollaboration>
2796 </ProcessSpecification>
2797
2798
```



## 2799 **Appendix E Instance Documents for Business Signals**

2800 The XML Schemas of the business signals are defined in [ebBPSS].

### 2801 **E.1 Acceptance Acknowledgment**

2802 The instance document for the AcceptanceAcknowledgment business signal is available as a text  
2803 file at:

2804

2805

### 2806 **E.2 Exception**

2807 The instance document for the Exception business signal is available as a text file at:

2808

2809 **Appendix F Example of NDD Instance Document (Non-**  
2810 **Normative)**

2811 The text file for this example of an *NDD* instance document for automated negotiation is  
2812 available at:

2813 **Appendix G Examples of Negotiation-Message Instance**  
2814 **Documents (Non-Normative)**

2815 The text files for the examples of negotiation message instance documents are available at:

## 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 is defined in a BPSS instance document.

**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 representation of the negotiation-protocol process by means of an XML instance document that conforms to the ebXML Business Process Specification Schema specification.

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

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

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

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

**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.

**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.

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