



Creating A Single Global Electronic Market

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23

# ebXML E-Commerce Patterns v1.0

## Business Process Team

11 May 2001

### 1 Status of this Document

This Technical Report document has been approved by the ebXML Business Process Project Team and has been accepted by the ebXML Plenary. This document contains information to guide in the interpretation or implementation of ebXML concepts.

Distribution of the document is unlimited.

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

***This version:***

[www.ebXML.org/specs/bpPATT.pdf](http://www.ebXML.org/specs/bpPATT.pdf)

***Latest version:***

[www.ebXML.org/specs/bpPATT.pdf](http://www.ebXML.org/specs/bpPATT.pdf)

23 **2 ebXML Participants**

24

25 Business Process Project Team Co-Leads:

26 Paul Levine, Telcordia

27 Marcia McLure, McLure-Moynihan, Inc.

28

29 Business Process/Core Components Joint Delivery Analysis Team Lead:

30 Brian Hayes, Commerce One

31

32 We would like to recognize the following for their significant contributions to the  
33 development of this document.

34

35 Editor:

36 Jamie Clark, -McLure-Moynihan, Inc.

37

38 Contributors:

39 Bob Haugen, Logistical Software LLC

40 Nita Sharma, Iona

41 David Welsh, Nordstrom.com

42

42 **3 Table of Contents**

43

44 **STATUS OF THIS DOCUMENT ..... 1**

45 **EBXML PARTICIPANTS ..... 2**

46 **TABLE OF CONTENTS ..... 3**

47 **INTRODUCTION..... 4**

48 SUMMARY..... 4

49 AUDIENCE..... 4

50 RELATED DOCUMENTS..... 4

51 DOCUMENT CONVENTIONS ..... 5

52 **DESIGN OBJECTIVES ..... 5**

53 PROBLEM DESCRIPTION ..... 5

54 TERMINOLOGY..... 5

55 SIGNIFICANT TERMS DEFINED IN EBXML..... 5

56 TERMS DEFINED FOR THE PURPOSE OF THIS DOCUMENT. .... 5

57 ASSUMPTIONS AND CONSTRAINTS ..... 6

58 CONSTRAINTS FROM LEGAL AND AUDITING REQUIREMENTS ..... 6

59 CONSTRAINTS FROM EBXML STRUCTURE AND STANDARDS..... 8

60 **CONTRACT FORMATION IN EBXML..... 8**

61 EBBPSS CONTRACT FORMATION FUNCTIONALITY..... 9

62 SIMPLE CONTRACT FORMATION PATTERN..... 10

63 6.2.1 *Requirements for all Business Documents and Document Envelopes..... 10*

64 6.2.2 *Requirements for all Offers..... 10*

65 6.2.3 *Requirements for all Acceptances..... 11*

66 6.2.4 *Requirements for all Rejections and Counteroffers..... 11*

67 DROP SHIP BUSINESS PROCESS EXAMPLE ..... 13

68 **SIMPLE AUTOMATED CONTRACT NEGOTIATION IN EBXML..... 17**

69 EBBPSS CONTRACT NEGOTIATION FUNCTIONALITY..... 17

70 CPA NEGOTIATION AS AN INSTANCE..... 19

71 **DISCLAIMER..... 20**

72 **CONTACT INFORMATION ..... 20**

73 **COPYRIGHT STATEMENT..... 21**

74

75

## 75 **4 Introduction**

### 76 **4.1 Summary**

77 This document is a supporting document to the ebXML Business Process Specification  
78 Schema [ebBPSS], to address common pattern implementation issues and provide  
79 examples. The 'Simple Contract Formation Pattern' defined here demonstrates a non-  
80 normative rule-defined subset of BPSS use for practical contracting purposes. It also is  
81 aligned with the "drop ship vendor" model collaboration used by the Worksheets  
82 published by the ebXML BP/CC Analysis Team. The 'Simple Negotiation Pattern'  
83 defined here demonstrates a non-normative rule-defined subset of BPSS use to allow  
84 simple exchanges of 'dry run' transactions and collaborations that may result in a  
85 collective decision by trading patterns to use them on an enforceable basis. It also may  
86 be suitable to automate the negotiation of ebXML CPA terms from CPPs.

### 87 **4.2 Audience**

88 This document is intended to be read by designers and implementer of ebXML business  
89 processes.

### 90 **4.3 Related Documents**

- 91 ebXML Technical Architecture Specification, version 1.0.4, 16 February 2001. ebXML  
92 Technical Architecture Project Team. [ebTA]
- 93 ebXML Business Process Specification Schema, version 0.99, 19 March 2001. ebXML  
94 Context/Metamodel Group of the Business Process/Core Components Joint Delivery  
95 Team. [ebBPSS]
- 96 ebXML TA Glossary. Version 0.99 , 11 May 2001 . ebXML Technical Architecture  
97 Team. [ebGLOSS]
- 98 ebXML Collaboration Protocol Profile and Agreement Specification, version 0.95, 19  
99 April 2001. ebXML Trading Partners Team. [ebCPP]
- 100 ebXML Automatic CPA Negotiation, version 0.1, 14 February 2001. ebXML Trading  
101 Partners Team. [Automatic CPA Negotiation 2001]
- 102 UN/CEFACT Modelling Methodology, version 9.1. 2001. UN Economic Commission for  
103 Europe. (CEFACT/TMWG/N090R9.1) [UMM]
- 104 Commercial Use of Electronic Data Interchange: A Report and Model Training  
105 Partner Agreement. 1992. American Bar Association Section of Business Law.  
106 [<http://www.abanet.org/buslaw/catalog/5070258.html>] [ABA Model Trading  
107 Partner Agreement 1992]
- 108 The Commercial Use Of Interchange Agreements For Electronic Data  
109 Interchange, UN/ECE Recommendation No.26. 1995. UN Economic  
110 Commission for Europe. (TRADE/WP.4/R.1133/Rev.1)  
111 [[http://www.unece.org/trade/untdid/texts/d240\\_d.htm](http://www.unece.org/trade/untdid/texts/d240_d.htm)] [UN/ECE Interchange  
112 Agreements for EDI 1995]

## 113 **4.4 Document Conventions**

114 The keywords MUST, MUST NOT, REQUIRED, SHALL, SHALL NOT, SHOULD,  
115 SHOULD NOT, RECOMMENDED, MAY, and OPTIONAL, when they appear in this  
116 document, are to be interpreted as described in RFC 2119 [Bra97].

## 117 **5 Design Objectives**

### 118 **5.1 Problem Description**

119 The BP Specification Schema [ebBPSS] contemplates exchanges of *Business*  
120 *Documents* composed into atomic *Business Transactions* each between two parties. In  
121 order to achieve the desired legal and economic effects of these exchanges, the  
122 structure of the *Business Transactions* must

- 123
- 124 • generate a computable success or failure state for each transaction that can be  
125 derived solely from the application of the ebBPSS standard and the data  
126 exchanged in the Business Documents and Business Envelopes,
- 127 • permit the parties to exchange *legally binding* statements and terms,
- 128 • permit the parties to exchange *nonbinding* statements and terms, in order to  
129 negotiate, and
- 130 • permit a logical composition of those exchanges into *Collaboration* patterns that  
131 allow agreements about sequences of transactions to be formed.

132

### 133 **5.2 Terminology**

#### 134 **5.3 Significant terms defined in ebXML**

135 *Business Collaboration* -- The "Business Collaboration" object as defined in ebBPSS.

136

137 *Business Document* -- The "Business Document" object as defined in ebBPSS.

138

139 *Business Transaction* -- The "Business Transaction" object as defined in ebBPSS.

140

141 *Contract* -- Generally, a bounded set of statements and/or commitments between trading  
142 partners that are intended to be legally enforceable as between those parties.

143 [ebGLOSS]

144

145 *Legally Binding* -- An optional character of a statement or commitment exchanged  
146 between trading partners (such as an *offer* or *acceptance*), set by its sender, which  
147 indicates that the sender has expressed its intent to make the statement or commitment  
148 legally enforceable. [ebGLOSS]

149

#### 150 **5.4 Terms defined for the purpose of this document.**

151 *Acceptance* -- A responding party's *document* indicating agreement with a received *offer*.

152

153 *Binding* -- See "Legally Binding" above.

154

155 *Business Signal Parameters* -- The following parameters as defined in ebBPSS:

156	isAuthorizationRequired	timeToPerform
157	isIntelligibleCheckRequired	isAuthenticated
158	isNonRepudiationRequired	isConfidential
159	isNonRepudiationOfReceiptRequired	isLegallyBinding
160	timeToAcknowledgeReceipt	isTamperProof
161	timeToAcknowledgeAcceptance	isGuaranteedDeliveryRequired

162

163 *Collaboration* -- See "Business Collaboration" above.

164

165 *Counteroffer advice* -- A message bound to a *rejection*, indicating that the sender intends to send a new *offer* regarding the same subject matter.

166

167 *Document* -- See "Business Document" above.

168

169 *Offer* -- A *document* proposing business terms by a requesting party addressed to a responding recipient. A *binding* offer entitles the recipient to form a *contract* with the requesting party by responding with a *binding acceptance*.

170

171 *Nonbinding* -- An optional character of a statement or commitment exchanged between trading partners (such as an *offer* or *acceptance*), set by its sender, that indicates the intent to be *legally bound*. See "Legally Binding" above.

172

173 *Rejection* -- A responding party's *document* indicating that it rejects a received *offer*.

174

175 *Transaction* -- See "Business Transaction" above.

176

177

## 178 **5.5 Assumptions and Constraints**

179

180

## 181 **5.6 Constraints from legal and auditing requirements**

182

183

184 a) **Enforceability requires an expression of intent.** In order for a message to be given legally enforceable effect, whatever its form, the author must indicate his intent to be bound. The message's sender may accomplish this by intentional use of a standard that specifies a mark, attribute or protocol indicating legal assent. In a paper context, this might mean affixing a written signature, plus an absence of elements that qualify its enforceability. (Elements that might tend to do so could include a substantive precondition to enforceability, the omission of essential terms, or a 'draft' stamp on its face that impeaches the document's finality).

185

186 b) **Each offer must succeed or fail.** The *offer* in a *binary transaction* must be definitively resolved in order to end the *transaction*. (This is true whether or not the offers are *binding*.) Offers that are followed by an explicit acceptance must be resolved as accepted. All other responses – including time-outs, rejections and counteroffers – must be resolved as a type of *rejection*. Either resolution

187

- 200 should result in completion of the *transaction*, together with a suitably provable  
201 "success" or "failure" end state that informs further processing of the results of  
202 the *transaction*.
- 203 c) **Each acceptance must relate precisely to an offer.** Each *acceptance* of an  
204 *offer* (whether or not *binding*) must unambiguously refer to the *offer* accepted, in  
205 a manner that produces artifacts transmitted between the parties and suitable for  
206 proving the identity of the terms that were *accepted*.
- 207 d) **Replicable and computable transaction state closure.** In the foregoing  
208 context, "suitable proof" of the offer and acceptance events, means that  
209 determinable computation of the *transaction's* "success" or "failure" state must be  
210 replicable by both trading partners at run time, as well as third parties (such as a  
211 court) after the fact, using only artifacts transmitted within messages associated  
212 with the *transaction*.

213  
214  
215

---

#### *A sidebar: Nonrepudiation and Enforceability*

216 Users of this document should note that the *defined signals*  
217 *isNonrepudiation-Required*, *isNonRepudiationOfReceiptRequired* and  
218 *isLegallyBinding* are significantly distinct from the *generalized goals* of  
219 nonrepudiation and legal enforceability. Invoking the former should assist,  
220 but does not assure, the latter. The goal of a well-designed electronic  
221 commerce model is to *reduce* the risk of repudiation and unenforceability to  
222 a reasonable minimum. *No* system will completely eliminate either risk.  
223 See [ABA Model Trading Partner Agreement 1992] and [UN/ECE  
224 Interchange Agreements for EDI 1995].

225 Repudiation risk occurs *whenever* a trading partner has an opportunity to  
226 avoid the consequences of its commitments. For example, under the BPSS,  
227 if you impose an *timeToAcknowledgeAcceptance* parameter (*time*>0) on a  
228 trading partner's response to you, he may validly reply with an exception  
229 claiming that your requesting document does not conform to the relevant  
230 business rules. That claim may or may not be true: in fact, nothing in the  
231 standard *computationally* prevents him from making a false exception at  
232 runtime. That opportunity may be the functional equivalent for him of a  
233 chance to repudiate. Say your requesting document offers to buy 1000 units  
234 of X. Assume you and he have a pre-existing contract requiring him to sell  
235 you 1000 units of X whenever you offer to buy them. He may have  
236 received, parsed and understood your requesting document as a purchase  
237 order to buy X. But he is still in a position to inaccurately claim that your  
238 purchase order failed a business rule check. Perhaps he has a limited supply  
239 of X, and a buyer who will pay more than you. *At run time*, there likely is no  
240 way for you to tell.

241  
242 What *business signal parameters* offer, in that instance, is a set of process  
243 rules that require you or him to keep and store significant artifacts from the  
244 transactional messaging, that *later* may be impartially interpreted. Any  
245 "legally binding" obligation should, as a design matter, generate a set of  
246 those artifacts that would be useful in proving later in court that (for  
247 example) the claim of a failed business rule check was fraudulent.

248  
249 In the electronic commerce context, an evaluative judgment that a set of  
250 messages creates an *enforceable* or *nonrepudiatable* contract should be  
251 understood to mean that the quality and coherence of the evidentiary artifacts  
252 available to prove it are acceptably strong. We *cannot* prevent trading  
253 partners from lying. We *can* design signal structures that make it easier to  
254 prove later.  
255

---

257

## 258 **5.7 Constraints from ebXML structure and standards**

259

260 a) **Business Service Interface.** An ebXML *collaboration* is conducted by two or  
261 more parties, each using a human or an automated business service interface  
262 that interprets the *documents* and *document envelopes* transmitted and decides  
263 how to (or whether to) respond.

264 b) **Decomposition of business processes into binary pairs.** All *collaborations*  
265 are composed of one or more atomic *transactions*, each between two parties.  
266 Multi-party or multi-path economic arrangements are possible, and may be  
267 arranged in a single *collaboration*, but must be decomposed into bilateral  
268 *transactions* in order to be modeled and executed under the ebBPSS.

269 c) **Definitive use of visible end state machines.** The ebBPSS uses guard  
270 expressions that permit the reliable computation of *transaction* "success" or  
271 "failure" transaction end states. For the sake of reliability, these must be the  
272 exclusive source of instructions to the trading partner's business service  
273 interface, within the scope of that *transaction*. Any contingency or business  
274 logic that is to govern the reaction of the business service interface to a  
275 *transaction* must be expressed within the relevant *collaboration* in a manner that  
276 affects the end state, and that manner must be made visible to both trading  
277 partners in the business process specification referenced by the CPA to which  
278 the partners agreed.

279 d) **Function of digital signatures.** Several ebXML specifications permit electronic  
280 signatures (generally conforming to the W3C XML-DSIG standard) to be used for  
281 various purposes such as message integrity or sender identification. Therefore,  
282 the presence or absence of an electronic signature bound to a *document* by  
283 hashing or the like, cannot, by itself, be used to indicate the *document's binding*  
284 character.

285 e) **Ability to declare documents nonbinding.** The ebBPSS permits a trading  
286 partner to explicitly designate specific *documents* as *binding* or *nonbinding* by  
287 setting the Boolean parameter "isLegallyBinding".  
288

## 289 **6 Contract Formation in ebXML**

290



## 291 **6.1 ebBPSS contract formation functionality**

292

293 The constraints listed in Section 5.3.2 provide implementers with a specific set of tools  
294 for producing reliable artifacts to evidence contracts. The ebBPSS constrains process  
295 designers and implementers to two methods of affecting the determination of a  
296 transaction's "success" or "failure" end states:

297

298 1. The semantic contents of the *documents* and *document envelopes* that pass  
299 between the trading partners can be referenced and evaluated in a guard  
300 expression, and

301 2. The BPSS *business signal parameters* that resolve requests for  
302 acknowledgement and the like, short of substantive responses to  
303 BusinessDocuments.

304 In the context of simple contract formation, trading partners may explicitly form a  
305 contract by exchanging requesting documents constituting *binding offers*, and  
306 responding documents constituting *binding acceptances*, resulting in a demonstrably  
307 successful or failed negotiation of the business terms proposed in the offer.

308

309

310

311

---

### 311 *A sidebar: Explicit vs. implicit contracts*

312

313

314

315

316

317

313 There is an important distinction between the legal view of contracts and this  
314 document's definition of "contract". The former encompasses a much  
315 broader range of phenomena that may be interpreted as a enforceable  
316 agreement.

317

318

319

320

321

322

323

324

318 In commerce, some agreements are formed by reciprocal actions and implied  
319 promises, without any explicit messages in one or both directions. If one  
320 trading partner acts in a manner that reasonably seems to convey an offer to  
321 sell an object, and the other partner carts off the object, a court may conclude  
322 that the latter's behavior is acceptance by performance. In such a case, the  
323 *implicit contract* is formed by *inferring* acceptance, *as if* the latter party had  
324 explicitly accepted an explicit sale offer.

325

326

327

328

329

330

331

332

333

334

335

336

337

338

326 In this document we are only concerned with exchanges of explicit messages  
327 that, if they logically match, will produce an explicit contract expressed in  
328 and evidenced by the messages. However, process designers should bear in  
329 mind that the terms of those explicit contracts can suffer interference from  
330 subsequent interpretation of events. Courts are *not* barred from concluding,  
331 and trading partners are *not* barred from arguing, that a *course of behavior*  
332 between electronic trading partners gives rise to an implicit legally  
333 enforceable agreement, or an implicit enforceable change to an explicit  
334 electronically-formed contract, even in the absence of further exchanges of  
335 legally binding messages.

---

339 The next section describes a pattern that may be used to explicitly exchange a series of  
 340 one or more *transactions*, within a *collaboration*, to form a legally binding contract.  
 341

## 342 **6.2 Simple Contract Formation Pattern**

343  
 344 *Contracts* MAY be formed by ebXML *collaborations* by the inclusion of *offers* and  
 345 *acceptances* that conform to the Simple Contract Formation Pattern described here.  
 346 This section describes a pattern that may be used to explicitly exchange a series of one  
 347 or more *transactions*, within a *collaboration*, to form a legally binding contract. The  
 348 Simple Contract Formation Pattern is constrained by rules that define a constrained  
 349 subset of the alternative methods available for forming a contract under the ebBPSS  
 350 schema. The pattern illustrates a subset of functionality that a particular domain or  
 351 group of trading partners might elect.  
 352

### 353 **6.2.1 Requirements for all Business Documents and Document Envelopes**

354  
 355 To use this sample pattern, a business process must conform to the following rules,  
 356 which are elective ("non-normative") to the ebBPSS standard, but required by this  
 357 pattern:  
 358

- 359 1. Guard expressions in this pattern MUST refer only to one or more data fields that  
 360 reside within the *Business Document* contained in the *Document Envelope* being  
 361 evaluated. For example, this rules out a success or failure end state being  
 362 generated by guard expressions that rely on the *Document Envelope* name, or  
 363 the *isPositiveResponse* attribute of the *Document Envelope*.
- 364 2. *Business Documents* in this pattern MUST NOT set the *IsLegallyBinding* attribute  
 365 to "No". This simplifies the evaluation that each business service interface must  
 366 conduct of a document. Among other things, this rule also bars a number of  
 367 approaches, such as the negotiating function demonstrated in the Simple  
 368 Negotiation Pattern described in Section 7 of this document.
- 369 3. All *Business Transactions* and *Business Documents* in this pattern MUST  
 370 conform to the one of the six "transaction patterns" defined in Chapter [9] of the  
 371 UMM N90 metamodel. This is an example of re-use. The six recommended  
 372 N90 patterns dictate or constrain the use of certain ebBPSS *business signal*  
 373 *parameters* such as *timeToPerform* and *timeToAcknowledgeReceipt*. By re-  
 374 using well-defined permutations of the *business signal parameter* values, the  
 375 process designer and the process user can choose to rely on the UMM N90  
 376 standard designers, who have in the UMM documentation described the logical  
 377 relationship between the signals, and made suggestions about the suitability of  
 378 particular permutations to particular business needs.

### 379 380 **6.2.2 Requirements for all Offers**

381  
 382 Under this pattern:  
 383

- 384 1. A *document* constituting an *offer* MUST be the *Business Document* sent within  
385 the Requesting Business Activity.  
386
- 387 2. Any *Business Document* constituting an *offer* MUST NOT contain any data that is  
388 evaluated by a guard expression but is not transmitted with the *Document*  
389 *Envelope* that contains that *Business Document*. Another way of putting this is  
390 that the offer document may not incorporate data by reference that would not be  
391 captured by an archive of the message in which the document is sent and  
392 received. (While it certainly may be possible for trading partners to work out an  
393 acceptably safe protocol for incorporation by linking reference, that function  
394 would make more complex the archiving of contract formation evidence. This  
395 simple pattern prohibits the linking so as to keep those archiving requirements  
396 very simple.  
397

### 398 **6.2.3 Requirements for all Acceptances**

399

400 Under this pattern:

401

- 402 1. Business processes MUST define one and only one responding *Business*  
403 *Document* that is evaluated by the processes' guard expressions as producing a  
404 "success" end state (and thus the end of that atomic *transaction*). That  
405 *document* constitutes the *acceptance*, and MUST be the *Business Document*  
406 sent within the *Responding Business Activity* of the same *Business Transaction*  
407 in which the *offer* was sent as the *Requesting Business Activity*.  
408
- 409 2. Repeating the terms of an *offer*, in the *document* constituting an *acceptance* to  
410 that *offer*, is NOT RECOMMENDED. Repetition of terms previously transmitted  
411 creates ambiguity. If the terms sent "as accepted" are identical to those sent "as  
412 offered", a comparison by the offering party is redundant. The parties have  
413 already made provision for the desired level of message integrity and security by  
414 setting the business signal parameters. Therefore it is possible that the parties  
415 are already reflecting back acknowledgement messages. If the comparison  
416 reveals a difference, the comparing party is faced with ambiguity among the  
417 artifacts that might be its legally relevant evidence, and no clear rule for whether  
418 the document type or the document contents govern.  
419

### 420 **6.2.4 Requirements for all Rejections and Counteroffers**

421

#### **6.2.4.1 Handling of explicit substantive rejections**

422

423 Under this pattern:

424

- 425 1. A *document* constituting a *rejection* MUST be the *Business Document* sent within  
426 the *Responding Business Activity* of the same *Business Transaction* in which the  
427 *offer* was sent as the *Requesting Business Activity*.  
428
- 429 2. A *document* constituting a *rejection* terminates the *transaction* initiated by the  
430 *offer* being rejected, by transitioning to a "failure" end state.  
431

#### 6.2.4.2 Handling of counteroffers

The request-response paradigm of the BPSS (as well as the UMM N90 "transaction patterns" requires that all counteroffers be expressed in two documents or signals: (a) a *rejection*, to properly close the request-response pair initiated by the *offer*, and (b) a counteroffer, expressed as a new *offer* in which the rejecting party is the initiator of a new *transaction*.

Thus, under this pattern:

1. In order to propose new or modified terms, the rejecting party **MUST** send a new *offer* containing the proposed terms, thereby starting a new *transaction* response-request pair.
2. A *document* constituting a *rejection* **MAY** be bound to a signal indicating that a counteroffer is coming, which is called a "*counteroffer advice*" in this document.
3. A *counteroffer advice* **MUST NOT** be treated by itself as an *offer*, nor as a *binding document*.
4. A *counteroffer advice* **MAY** be communicated by a message *document* bound to the *rejection document* in a manner compliant with ebXML standards (such as in a common *Document Envelope*), or by a unique *rejection document* subtype used only to signify a *counteroffer advice* as well as a *rejection*. However, the method of indicating a *counteroffer advice* **MUST** be specified in the applicable CPA.
5. Receipt of a *counteroffer advice* **MUST NOT** toll or re-set a *transaction* time-out clock (such as *timeToPerform*) started by the rejected *offer*. The business service interface of an ebXML user **MAY** use the *counteroffer advice* for its own purposes.
6. It is **RECOMMENDED** that a *collaboration* handling system include a separate collaboration-oriented time-out clock, distinct from the ebBPSS *timeToPerform* rules applicable to an individual *transaction*. The rules for that clock may include an explicit manner for handling *counteroffer advice* messages. Under ebBPSS the time-out conclusions of that timer do not directly affect the timer objects in the schema's metamodel. However, it would likely inform the decisions of a business service interface decisions regarding, among other things, when to throw an explicit *rejection*, and when to rescind an *offer* (if the conditions of the *offer* permit it).
7. A separate *document* type for offers not capable of a counteroffer -- sometimes called "unalterable" offers -- is **NOT RECOMMENDED**. Under the ebBPSS schema, every *offer* must be simply *accepted* or *rejected* on a "take it or leave it" basis. Processing of counteroffers generally will be handled in a more robust and informative manner by the recipient's business service interface interpreting the rejection, not by a preemptive failure caused by a *document* type.

483 *A sidebar: The utility of patterns in handling business signal parameters*

484

485

486

487

488

489

490

491

492

493

494

495

496

497

498

499

500

501

502

503

504

505

As standards that attempts to permit interoperability with a wide range of current practices, ebXML's schemas almost certainly provide more functionality than most users will initially employ. The BPSS schema specifies some mandatory signals and state handling functions, and many more optional ones. Some potential users may wish to permit or support only a select subset. Some user domains may wish to provide a simple upgrade path, by constraining their use of the BPSS schema parameters to a subset that maps easily to the cognate functions of their legacy system.

The Simple Contract Formation Pattern is an illustrative example of a set of rules that might be voluntarily adopted to present a simpler set of process design options. This is a hypothetical pattern, not an actual recommendation of suitability. It merely illustrates how a process designer might further constrain the possible uses of BPSS functionality to make it more "user-friendly" to a particular user base. As a result, a process designer could (1) offer to this use base only business processes that conform to the pattern, and (2) advise users to interrogate new business processes to see if they require functionalities that this pattern excludes.

---

### 506 **6.3 Drop Ship Business Process example**

507

508

509

510

511

512

513

514

515

The following table illustrates the composition of a multiparty *collaboration* from multiple binary *collaborations* and *Business Transactions*, each composed of one or two *Business Documents*. This collaboration can be conducted under the Simple Contract Formation Pattern defined in the previous section. The UMM N90 transaction pattern applicable to each *transaction* is noted in brackets in the second column in the following table. The hypothetical *collaboration* is a superset of the same *Business Transactions* used as the illustrative values that populate the sample "Worksheets" in the ebXML Business Process Analysis Worksheet and Guidelines [bpWS].

**DROP SHIP SCENARIO  
SAMPLE USE OF BUSINESS PROCESS PATTERNS**

*Version 1  
10 May 2001  
Jamie Clark, Bob Haugen, Nita Sharma, Dave Welsh, Brian Hayes*

Notes on use of roles: Authorized Roles are assigned to each of the two roles in each Business Transaction. Each MUST be unique within a Business Process (or else you can't definitively point to them for process specification purposes). It is RECOMMENDED that Authorized Roles be named to facilitate resource discovery, by creating unique composite values from a controlled vocabulary. There is no normative rule for generating the names. In this table, we have used a *hypothetical* controlled vocabulary which includes "Inventory Buyer, Catalog Publisher, Merchandising, Buying Customer, Customer Service, Accounts Receivable, , Shipper, , , Payer, Payee, , , Credit Authority Service, , , " to promote resource discovery and re-use, and we have elected to use the Business Transaction names (and, where necessary, Collaboration names) to qualify and distinguish them.

BUSINESS PROCESS	BINARY COLLABORATION (protocol)	BUSINESS TRANSACTION (activity) <i>[Pattern per N090]<sup>1</sup></i>	INITIATING / REQUESTING SIDE	REQUESTING DOCUMENT	RESPONDING SIDE	RESPONDING DOCUMENT
BPUC-5.7 - Sales-Product-Notification  Actors: Retailer, DSVendor	BC-6.9-Sales-Product-Offering	BT-8.9-Product-Offering  <i>[Request / Confirm]</i>	PARTNER TYPE: DSVendor  AUTH ROLE: Catalog Publishing	Product Catalog Offering (e.g. X12 832, ver 4010)	PARTNER TYPE: Retailer  AUTH ROLE: Merchandising	Product Catalog Acceptance

<sup>1</sup> This column suggests use of one of the six demonstrative signal patterns offered in the UN/CEFACT TMWG N90 metamodel. Re-using these reduces our need to pay attention to the parameter values.

BPUC-5.6-Inventory-Management  Actors: Retailer, DSVendor	BC-6.7-Vendor-Inventory-Reporting	BT-8-5-Vendor-Inventory-Report  <i>[Notification]</i>	PARTNER TYPE: DSVendor  AUTH ROLE: Inventory Buyer	Inventory Report	PARTNER TYPE: Retailer  AUTH ROLE: Inventory Buyer	On Hand Product Availability
BPUC-5.1-Firm-Sales-Order  Actors: Customer, Retailer	BC-6.1-Create-Customer-Order <sup>2</sup>	BT-8.1-Firm-Customer-Sales-Order  <i>[Business Transaction]</i>	PARTNER TYPE: Customer  AUTH ROLE: Buying Customer	Sales Order <sup>3</sup>	PARTNER TYPE: Retailer  AUTH ROLE: Customer Service	Confirmation <sup>4</sup>
BPUC-5.2-Customer-Credit-Inquiry  Actors: Retailer, Credit Authority	BC-6.2-Check-Customer-Credit <sup>5</sup>	BT-8.2-Check-Customer-Credit  <i>[Request / Response]</i>	PARTNER TYPE: Retailer  AUTH ROLE: Customer Service	Credit Check	PARTNER TYPE: Credit Authority  AUTH ROLE: Credit Service	Credit Check Response

<sup>2</sup> In designing the business process, Retailer might choose to confirm the order only after successfully completing the *Product Fulfillment* collaboration. In that case *Order Fulfillment* would nest inside *Firm Order*.

<sup>3</sup> Provided via web browser.

<sup>4</sup> Provided via email

<sup>5</sup> The suggested pattern is "Request/Response", not "Commercial Transaction" in N90 usage, because information was transmitted on demand, but no economic commitment (credit allocation) was made.

BPUC-5.4- Purchase- Order- Managem ent  Actors: Retailer, Vendor	BC-6.4- Create - Vendor- Purchase- Order	BT-8.4-Create-Vendor- Purchase-Order  <i>[Business Transaction]</i>	PARTNER TYPE: Retailer  AUTH ROLE: Inventory Buyer	Purchase Order Request	PARTNER TYPE: DSVendor  AUTH ROLE: Customer Service	Purchase Order Acknowldgme nt
BPUC-5.5- Ship- Goods  Actors: DSVendor, Transport Carrier	BC-6.5- Shipment- Instruction	BT-8.7-Shipment- Notification  <i>[Business Transaction]</i>	PARTNER TYPE: DSVendor  AUTH ROLE: Shipper	Shipment Instruction	PARTNER TYPE: Transport Carrier  AUTH ROLE: Customer Service	Bill of Lading
	BC-6.6- Confirm- Shipment	BT-8.8-Confirm-Shipment  <i>[Notification]</i>	PARTNER TYPE: DSVendor  AUTH ROLE: Shipper	Advance Ship Notice	PARTNER TYPE: Retailer  AUTH ROLE: Customer Service	NONE
BPUC-5.3- Customer- Credit- Payment  Actors: Retailer, Credit Authority	BC-6.3- Process- Credit- Payment	BT-8.3-Charge- Customer-Credit  <i>[Business Transaction]</i>	PARTNER TYPE: Retailer  AUTH ROLE: Accounts Receivable	Charge Credit	PARTNER TYPE: Credit Authority  AUTH ROLE: Credit Authority Service	Confirm Credit
BPUC-5.8- Present- Invoice	BC-6.10- Invoice- Presentment	BT-8.11-Present-Invoice  <i>[Notification]</i>	PARTNER TYPE: DSVendor  AUTH ROLE: Payee	Invoice	PARTNER TYPE: Retailer  AUTH ROLE: Payor	NONE

519 Table 6-1 Inventory of Key Objects for Drop Ship Hypothetical MultiParty Collaboration



520

## 521 **7 Simple Automated Contract Negotiation in ebXML**

### 522 **7.1 ebBPSS Contract Negotiation Functionality**

523

524 In the prior section we examined contract formation by exchange of explicit, binding  
525 terms. At each step of the message exchange, the trading partners were making  
526 commitments that might (if properly met with a valid response) result in a "success" end  
527 state associated with an explicit contract formed by matching offer and acceptance.

528

529 Trading partners may also wish to exchange proposed terms, without making an  
530 assertion of intent to be legally bound. This is analogous to the paper contracting  
531 practice of exchanging unsigned drafts or term sheets.

532

533 Of course, trading parties may interrogate proposed business processes in a CPP or  
534 CPA independently, and then communicate in a human-readable fashion about the  
535 suitability and desirability of the specified process.

536

537 Under the ebBPSS, trading partners also have the opportunity to exchange Business  
538 Documents in a run-time fashion, with their isLegallyBinding parameter set to "No", and  
539 thereby test whether a particular sequence of exchanged BusinessDocuments results in  
540 a mutually satisfactory outcome.

541

542 Having done so, and concluded (independently) that the resulting collaboration is  
543 acceptable, the same partners are then in a position to efficiently duplicate the sequence  
544 by changing one parameter -- setting the isLegallyBinding parameter set to "Yes"  
545 throughout -- and thereby communicate the "dry run" contractual sequence as an  
546 enforceable transaction.

547

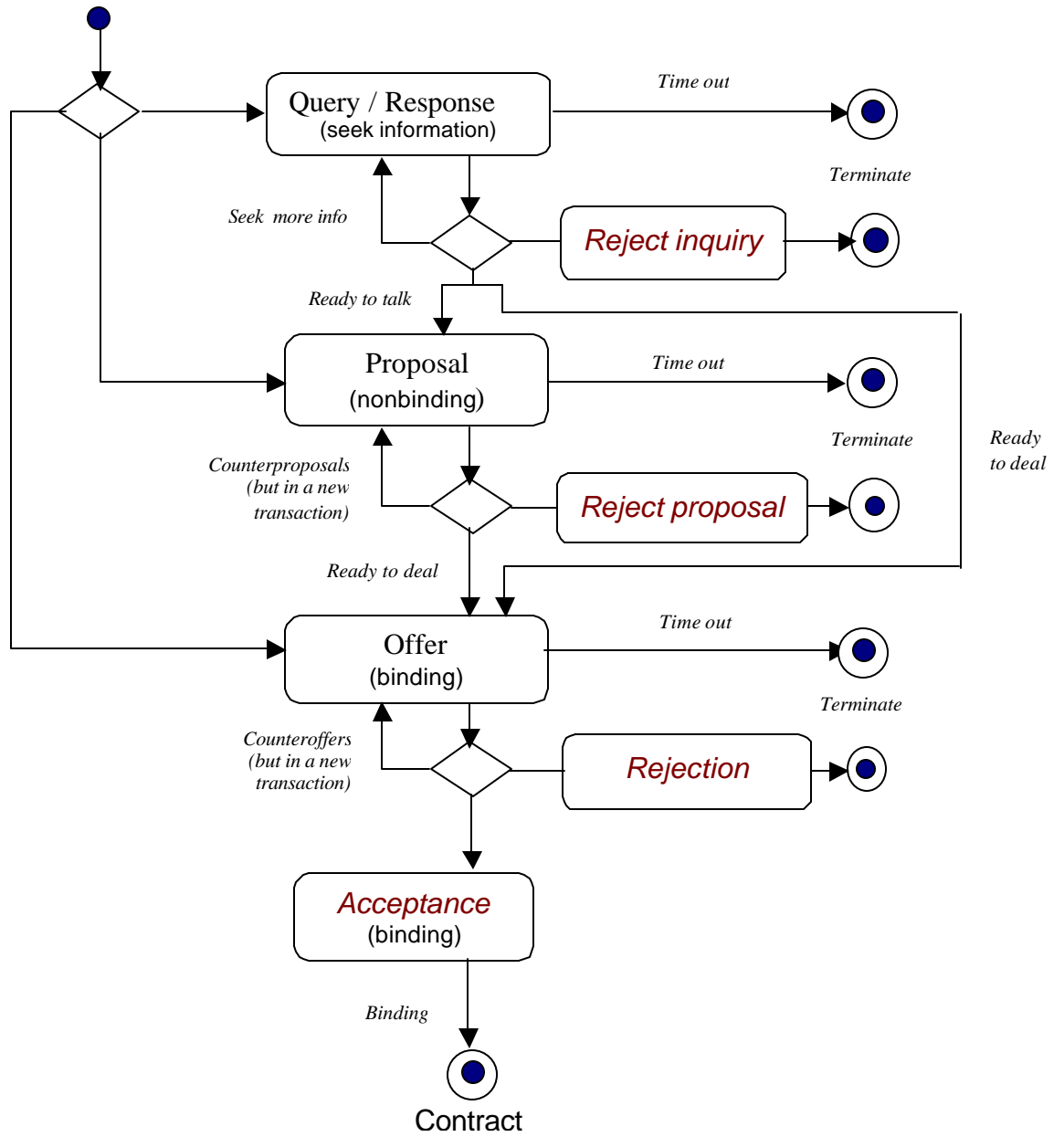
548 The generalized flow of events resulting from the foregoing approach is illustrated in the  
549 following activity diagram.

550

551

552

552



552

*Figure 7-1 Hybrid Activity Diagram for Simple Negotiation Pattern*

## 553 **7.2 CPA negotiation as an instance**

554

555 Some ebXML users may initiate communications by selecting from a sheaf of pre-set  
556 CPAs. Others may wish to negotiate a CPA dynamically by negotiating a choice from  
557 among a pre-set group of CPAs, or assembling a CPA from two CPPs. The Simple  
558 Negotiation Pattern may be used to perform such a negotiation, by sending a proposed  
559 CPA on a nonbinding basis (`isLegallyBinding="No"`) as a BusinessDocument to a  
560 proposed trading partner, in a single BusinessTransaction which indicates that the sole  
561 guard expression condition for a "success" end state is return of the identical  
562 BusinessDocument, followed (consistent with the foregoing pattern) by either:

563

564 1. A nonbinding substantive acceptance, indicated by the return of the CPA,  
565 which can then be formally agreed by a second similar exchange with the  
566 `isLegallyBinding parameter="Yes"`.

567 2. A rejection by explicit message, timeout or counteroffer advice, and in the  
568 latter case, a new exchange based on the CPA contained in the new offer  
569 heralded by the counteroffer advice.

570

571 The CPA Specification [ebCPP] requires signature of the CPA for substantive reasons.  
572 In order to satisfy that requirement, in the design of the foregoing process, the  
573 BusinessDocument containing the proposed CPA MUST bear a  
574 `"isNonrepudiationOfReceiptRequired" parameter="Yes"`.

575

576 In order to initiate an ebXML compliant transaction, trading partners must refer to a CPA.  
577 If potential trading partners are attempting to negotiate a CPA in such a transaction, they  
578 MUST nevertheless agree to a common CPP under which the CPA negotiation occurs.  
579 It is RECOMMENDED that the prospective trading partner who initiates that preliminary  
580 negotiation do so by specifying agreement to a CPP already offered by the non-initiating  
581 party (e.g., held out in a registry as being available for that party).

582

583 Potential trading partners who wish to be assured that their negotiation over competing  
584 prospective CPAs will computationally resolve to a CPA, without human intervention,  
585 may choose to employ the suggested set of default business rules described in the  
586 "Conflict resolution of equally weighted options" section of the [Automatic CPA  
587 Negotiation] document. However, parties are free to accept or reject the adoption of  
588 those rules.<sup>6</sup>

---

<sup>6</sup> Readers should note that the architects of the ebXML patterns *generally* seek to leave the selection of such matters up to the individual user. If I want to specify in a registry that I only transact in cuneiform on clay tablets, albeit wrapped in an ebXML data structure, the *standards* generally leave me free to do so. (As a practical matter, under the BPSS we would be looking at a "Business Document" constituting a conventional XML wrapper around a highly unconventional "Attachment". Also, to remain in compliance with the BPSS one would have to convert the cuneiform to transmittable form -- perhaps by shipping a JPEG file -- and setting the "spec" parameter of the "Attachment" object to a resolvable URI that allegedly informs a reader how to interpret the JPEG picture.) How the *market* may react to this is an

589

590 **8 Disclaimer**

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

595 **9 Contact Information**

596 Team Leader (of the CC/BP Analysis group of the Joint Delivery Team):

597

598 Brian Hayes  
 599 Commerce One  
 600 4440 Rosewood Drive  
 601 Pleasanton, CA  
 602 USA  
 603 +1 (925) 788-6304  
 604 *brian.hayes@commerceone.com*

605

606 Editor

607

608 James Bryce Clark  
 609 [McLure-Moynihan](#), Inc.  
 610 28720 Canwood Street Suite 208  
 611 Agoura Hills, CA 91301  
 612 USA  
 613 +1 (818) 706-3882  
 614 *Jamie.clark@mmiec.com*

615

616

---

entirely separate consideration.

## 616 **Copyright Statement**

617

618 Copyright © UN/CEFACT and OASIS, 2001. All Rights Reserved.

619

620 This document and translations of it may be copied and furnished to others, and  
621 derivative works that comment on or otherwise explain it or assist in its implementation  
622 may be prepared, copied, published and distributed, in whole or in part, without  
623 restriction of any kind, provided that the above copyright notice and this paragraph are  
624 included on all such copies and derivative works. However, this document itself may not  
625 be modified in any way, such as by removing the copyright notice or references to  
626 ebXML, UN/CEFACT, or OASIS, except as required to translate it into languages other  
627 than English.

628

629 The limited permissions granted above are perpetual and will not be revoked by ebXML  
630 or its successors or assigns. This document and the information contained herein is  
631 provided on an "AS IS" basis and ebXML DISCLAIMS ALL WARRANTIES, EXPRESS OR  
632 IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION  
633 HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY  
634 OR FITNESS FOR A PARTICULAR PURPOSE.