



UN/CEFACT

**DRAFT**

United Nations Centre for Trade Facilitation and Electronic Business

1  
2  
3  
4  
5  
6

**UN/CEFACT  
STANDARD BUSINESS DOCUMENT HEADER  
Technical Specification**

Version 1.3

2004-6-04

## Table of Contents

7			
8			
9	<b>UN/CEFACT STANDARD BUSINESS DOCUMENT HEADER .....</b>	<b>1</b>	
10	<b>TABLE OF CONTENTS .....</b>	<b>2</b>	
11	<b>1 STATUS OF THIS DOCUMENT .....</b>	<b>5</b>	
12	<b>1.1 Disclaimer .....</b>	<b>5</b>	
13	<b>1.2 Contact Information .....</b>	<b>5</b>	
14	<b>2 INTRODUCTION .....</b>	<b>6</b>	
15	<b>2.1 Summary .....</b>	<b>6</b>	
16	<b>2.2 What is a Standard Business Document Header? .....</b>	<b>6</b>	
17	<b>2.3 How is it used in EDI and XML environments? .....</b>	<b>6</b>	
18	<b>2.4 The Scope of the Standard Business Document Header .....</b>	<b>7</b>	
19	2.4.1 What Makes the Standard Business Document Header Useful? ....	8	
20	<b>2.5 Business Opportunity and Benefits of the Standard Header .....</b>	<b>9</b>	
21	<b>2.6 Stakeholders and Audience .....</b>	<b>10</b>	
22	<b>2.7 Document Conventions .....</b>	<b>10</b>	
23	<b>3 OBJECTIVES .....</b>	<b>11</b>	
24	<b>3.1 Requirements .....</b>	<b>11</b>	
25	3.1.1 Constraints on the Standard Business Document Header .....	11	
26	<b>3.2 Principles of the Standard Business Document Header .....</b>	<b>12</b>	
27	<b>3.3 Layered Processing Model .....</b>	<b>12</b>	
28	<b>Figure 1 .....</b>	<b>13</b>	
29	<b>3.4 Services .....</b>	<b>13</b>	
30	<b>3.5 Routing .....</b>	<b>14</b>	
31	<b>3.6 Packaging .....</b>	<b>15</b>	
32	3.6.1 Access to the Standard Business Document Header Information		
33	when the Payload is Encrypted.....	16	
34	<b>4 MULTIPLE PARTNER ENVIRONMENT .....</b>	<b>17</b>	
35	<b>5 PROCESSING FLOW OVERVIEW .....</b>	<b>18</b>	
36	<b>Figure 2 .....</b>	<b>18</b>	
37	<b>6 USE CASE ANALYSIS .....</b>	<b>21</b>	
38	<b>Figure 3 .....</b>	<b>21</b>	
39	<b>6.1 Business Services .....</b>	<b>22</b>	
40	<b>Figure 4 .....</b>	<b>22</b>	
41	<b>6.2 Description .....</b>	<b>22</b>	
42	<b>6.3 Workflow Analysis .....</b>	<b>23</b>	
43	<b>Figure 5 .....</b>	<b>24</b>	
44	<b>Figure 6 .....</b>	<b>25</b>	
45	<b>7 HIGH LEVEL SCENARIO .....</b>	<b>25</b>	
46	<b>8 PATTERNS .....</b>	<b>26</b>	

47	<b>Figure 7</b> .....	<b>26</b>
48	<b>9 BUSINESS SCOPE</b> .....	<b>27</b>
49	<b>9.1 Technical Agreements and Business Agreements</b> .....	<b>28</b>
50	<b>9.2 Future Business Scopes</b> .....	<b>30</b>
51	<b>9.3 Scopes</b> .....	<b>31</b>
52	<b>Figure 8</b> .....	<b>32</b>
53	<b>Figure 9</b> .....	<b>33</b>
54	<b>Figure 10</b> .....	<b>34</b>
55	<b>10 STANDARD BUSINESS DOCUMENT HEADER DATA ELEMENTS</b> .....	<b>34</b>
56	<b>11 DETAILED USE CASE EXAMPLES</b> .....	<b>41</b>
57	<b>11.1 Use case 1. A non-ebXML environment</b> .....	<b>42</b>
58	<b>11.2 Use case 2. An ebXML environment</b> .....	<b>45</b>
59	<b>11.3 Use case 3. SBDH with EDI payload in an ebXML environment</b> .....	<b>52</b>
60	<b>11.4 Use of SBDH in Acknowledgement and Exception Situations</b> .....	<b>55</b>
61	<b>12 GLOSSARY</b> .....	<b>55</b>
62	<b>APPENDIX A SBDH SCHEMAS</b> .....	<b>59</b>
63	<b>A.1 BasicTypes.xsd</b> .....	<b>59</b>
64	<b>A.2 BusinessScope.xsd</b> .....	<b>59</b>
65	<b>A.3 DocumentIdentification.xsd</b> .....	<b>61</b>
66	<b>A.4 Manifest.xsd</b> .....	<b>62</b>
67	<b>A.5 StandardBusinessDocumentHeader.xsd</b> .....	<b>62</b>
68	<b>A.6 Partner.xsd</b> .....	<b>63</b>
69	<b>A.7 Schemas for Use with Samples</b> .....	<b>64</b>
70	<b>A.7.1 Simulated Order.xsd for Use with Sample 2</b> .....	<b>64</b>
71	<b>A.7.2 Simulated OrderResponse.xsd for Use with Sample 2</b> .....	<b>64</b>
72	<b>A.7.3 Simulated OrderProxy.xsd for Use with Sample 2</b> .....	<b>65</b>
73	<b>A.7.4 Simulated OrderResponseProxy.xsd for Use with Sample 2</b> .....	<b>65</b>
74	<b>A.7.5 Simulated EDIOrder.xsd for Use with Sample 3</b> .....	<b>65</b>
75	<b>A.7.6 Simulated EDIOrderProxy.xsd for Use with Sample 3</b> .....	<b>66</b>
76	<b>APPENDIX B SAMPLE XML INSTANCE FILES</b> .....	<b>67</b>
77	<b>B.1 Sample 1</b> .....	<b>67</b>
78	<b>B.2 Sample 2</b> .....	<b>69</b>
79	<b>B.2.1 Sample 2a Requesting Document</b> .....	<b>69</b>
80	<b>B.2.2 Sample 2b Responding Document</b> .....	<b>70</b>
81	<b>B.3 Sample 3</b> .....	<b>72</b>
82	<b>APPENDIX C THEORY BEHIND THE SBDH BUSINESS SCOPE</b> .....	<b>75</b>
83	<b>C.1 The Commonly Occurring Perspectives of Business Scope</b> .....	<b>76</b>
84	<b>C.2 Meta model</b> .....	<b>77</b>
85	<b>C.3 Wellformedness rules</b> .....	<b>78</b>
86	<b>C.4 Optional parts</b> .....	<b>78</b>
87	<b>C.5 NOTES</b> .....	<b>79</b>

88 **APPENDIX D RELATIONSHIP BETWEEN THE SBDH AND OTHER**  
89 **STANDARDS .....81**  
90 **Boundaries.....81**  
91 **Integration Points.....81**  
92 **Dependencies.....81**  
93

## 94 **1 STATUS OF THIS DOCUMENT**

95 This Technical Specification is being developed in accordance with the  
96 UN/CEFACT/TRADE/22 Open Development Process for Technical  
97 Specifications. The Standard Business Document Header specification is a result  
98 of a work project of the UN/CEFACT Applied Technology Group (ATG). This  
99 specification will be supported by the two working groups within ATG, ATG1  
100 (EDIFACT Syntax Structures) and ATG2 (XML Assembly Documents/Production  
101 Rules). The Standard Business Document Header (SBDH) [also known as  
102 Generic Header] Project Team has approved it for UN/CEFACT review.

103 This document contains information to guide in the interpretation or  
104 implementation of the specification.

105 This version: is Standard Business Document Header Technical Specification,  
106 Version 1.3 of 2004-06-04.

107 Previous versions: Standard Business Document Header Technical  
108 Specification, Version 1.2 of 2004-03-10.

109

### 110 **1.1 Disclaimer**

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

### 115 **1.2 Contact Information**

116 ATG Chair: Margaret Pemberton, Diskray Pty Ltd., diskray(at)w150.aone.net.au,  
117 Project Leader: Melanie Kudela, Uniform Code Council, mkudela(at)uc-council.org  
118 Document Editor: John Duker, Procter & Gamble, duker.jp(at)pg.com

## 119 2 INTRODUCTION

120

### 121 2.1 Summary

122 This specification defines the 'Standard Business Document Header' (SBDH)  
123 which will enable integration of documents between internal applications,  
124 enterprise applications, and business-to-business infrastructure by providing a  
125 consistent interface between applications. The standard header information will  
126 enable any application to determine the logical routing requirements and/or the  
127 logical processing requirements of a document based on information contained in  
128 the standard header. This can be accomplished with the use of key data  
129 elements including logical sender and recipient identifiers, a 'business document  
130 type', and other elements associated with a Standard Business Document (see  
131 Glossary) object.

132

133 Standard Business Documents (SBD) are used in supply chain, financial, and  
134 other processes to record and share data such as purchase orders, invoices, or  
135 item synchronizations. These business documents are typically created in one  
136 application and processed by one or more receiving applications, either within a  
137 single organization or an external organization (Trading Partner). A number of  
138 different proprietary approaches have been developed to route and process  
139 these documents.

140

141 The SBDH includes a set of standard elements necessary to determine the  
142 routing and processing of documents either as a header within, or linked with the  
143 document. The standard header can also optionally provide service and  
144 correlation information, at the business domain level, between trading partners.  
145 The standard header can provide the semantic information needed for the  
146 routing, processing and business domain context of documents, regardless of the  
147 data format of the document – XML or EDI or other format.

148

### 149 2.2 What is a Standard Business Document Header?

150

151 The SBDH contains information expressed in an XML format. The header  
152 provides information about the routing and processing of the Standard Business  
153 Document, whether the document is in an XML or EDI or other format. The  
154 SBDH is designed to be either an integral part of a Standard Business Document  
155 (e.g. either XML instance document or EDI interchange), or an object associated  
156 with the Standard Business Document itself.

157

### 158 2.3 How is it used in EDI and XML environments?

159

160 The UN/CEFACT Architecture supports both the EDI and XML communities. The  
161 Standard Business Document Header architecture will therefore support both  
162 EDI and XML e-business processes. Including a SBDH in each instance of the  
163 business document reduces the effort needed to route and process documents  
164 and permits trading partner organizations to use different implementation  
165 approaches.

166

167 When implementing EDI, the provision of an additional standard header may not  
168 always be necessary, since EDI interchanges already contain functionality for  
169 some of the information in the SBDH. An example is the EDIFACT UNB  
170 interchange header, the UNH message header, and the 'function' part of the  
171 BGM. The SBDH specification will allow for this existing approach and provides  
172 an option to express additional functionality, such as service and correlation  
173 information.

174

175 Trust relationships among business applications and middleware applications  
176 providing services for those business applications are admittedly complex. For  
177 example, middleware communications software components may provide and  
178 enforce cryptographic properties such as data confidentiality and digital  
179 signatures, and are often implicitly delegated authorizing functions both for  
180 authentication (by signing or other means) or for access control (submission of  
181 business documents for further processing).

182

183 There are no new security risks imposed by the use of a SBDH than are imposed  
184 by current middleware implicit delegation arrangements. The relationships  
185 between back end systems and middleware components are extremely diverse  
186 and heterogeneous. In such a situation, it is sufficient to allow the SBDH to work  
187 in two modes: no application level security and some application level security. In  
188 either case, the SBDH techniques can be made to work securely.

189

## 190 **2.4 The Scope of the Standard Business Document Header**

191

192 Many users, implementers and supporting industry standard bodies are in  
193 agreement on the need for a Standard Business Document Header. In their  
194 business-to-business activities, the SBDH will facilitate three business needs:

195

- 196 • The routing of business documents from one point to another. This refers  
197 not only to the transfer of information from an external originator to  
198 receiver, but also from one intermediate application to another.  
199 Information in the SBDH can help ensure that a document gets to the  
200 correct recipient.

- 201       • The simplified processing of documents. Processing refers to taking action  
202       on data, for example transforming it from one format into another.  
203       Information in the SBDH can reduce the effort required to determine the  
204       correct processing actions.
- 205       • Associating a data message with its originator is important from a  
206       business and legal perspective. It is especially important when using  
207       intermediaries for data transfer, as information from the transport protocol,  
208       may be lost after the initial transmission. Because information in the SBDH  
209       is retained, it can help ensure that a document's originator is correctly  
210       identified.

211

212   In addition to header functions provided by the SBDH for routing and/or  
213   processing of business documents, there is the need for a completely separate  
214   technical communications transport layer header which is defined by BCF/UMM  
215   as a message envelope. This technical communications layer header deals with  
216   communications protocols and physical addresses which are outside the scope  
217   of this technical specification. Transport specifications including EDIINT-AS2 and  
218   ebXML Message Service (ebMS) are among a number of possible transport  
219   options that address technical communications needs by defining a separate  
220   technical header. Transport layer headers are completely outside the scope and  
221   are a separate concern not addressed here (because they are in a different layer  
222   of the stack).

223

224   The SBDH is useful at the business application and middleware levels to provide  
225   for the routing and identifying of business documents. The information placed in  
226   the SBDH at the business payload level, will travel with the business information  
227   to many different workflows. In addition to the business payload information, it  
228   may be useful to the business application and middleware to know the original  
229   creator and intended receiver of the document. For the more complex creator  
230   and receiver business environments, there is a business need to use the SBDH  
231   for internal routing. The SBDH can enable this internal routing, eliminating the  
232   need to deeply parse and process an entire business document.

233

234   Within a legal context the terms 'Dispatch' and 'Reach' are commonly used to  
235   indicate when a data message leaves control of the originator and enters control  
236   of the recipient respectively. From a legal standpoint, these terms could replace  
237   the terms 'Send' and 'Receive' in some sections of this specification. These  
238   terms carry well defined semantics which are independent of any specific  
239   modeling methodology and technology. See UNCITRAL Model Law on Electronic  
240   Commerce < <http://www.uncitral.org/english/texts/electcom/ml-ecomm.htm> >.

#### 241   **2.4.1 What Makes the Standard Business Document Header Useful?**

242



243 The main purpose of the Standard Business Document Header is to bridge the  
244 gap for standards, such as the UN/CEFACT EDI standard, that do not have the  
245 functionality of ebXML standards to perform a complete collaboration framework.  
246 It gives other technical frameworks and other standards an ability to simply use  
247 the payload in a collaborative exchange. These other standards and frameworks  
248 do not easily allow a user to accomplish this collaborative exchange without  
249 utilizing the attributes of the SBDH.

250

251 The Business Collaboration Framework UN/CEFACT Modeling Methodology  
252 (BCF/UMM) header of a business document provides information related to  
253 address, security and signatures as may be required by the associated Business  
254 Transaction' (please refer to BCF/UMM Business Transactions View (BTV)).  
255 Although according to BCF/UMM, some kind of document header is mandatory,  
256 the use of the Standard Business Document Header is not a replacement for the  
257 technical communications header nor is it mandatory. It is rather a useful  
258 business level header, which may be used optionally. As such we have identified  
259 four use case scenarios, which warrant the existence of the SBDH information as  
260 a separate header for business information. The four use cases are:

261

- 262 1. the middleware translation and transport use case
- 263 2. the Business Service Interface use case
- 264 3. the XML header with EDI messaging convergence use case.
- 265 4. incorporation by reference as a legal aspect of data message exchange

266

#### 267 **2.4.1.1 Legal Aspects of Electronic Data Exchange**

268

269 A key use case for the SBDH is one where it may be used in a legal aspect  
270 to carry legal provisions and contract terms. UMM, ebXML and other  
271 collaboration frameworks provide only limited capabilities to associate the  
272 exchange of electronic information with legal provisions and contracts. A  
273 good example of this is an exchange of a "Price List" that may be  
274 accompanied by usage and confidentiality terms & conditions.

275

276 Associating messages with terms & conditions and legal documents is an  
277 important requirement and the SBDH may be useful in this role. The Unified  
278 Business Agreements and Contracts (UBAC) project is investigating the  
279 possibilities of adding an additional Business Scope in order to facilitate  
280 association between data messages and legal provisions. (See also section  
281 on Business Scopes in this document.) Likely candidates for this projected  
282 Agreement Scope are contract terms, signature reference and intent  
283 expression.

284

## 285 **2.5 Business Opportunity and Benefits of the Standard Header**

286 Although routing and processing instructions are not necessarily an integral part of a  
287 document, use of the Standard Business Document Header will allow organizations,  
288 with applications which are not yet fully process-centric, to take part in the process-  
289 centric approach and avoid wasted effort in developing customized routing and  
290 processing scenarios for each category of business data. Trading Partner  
291 organizations using different communication and integration approaches will find the  
292 SBDH a benefit since the business data payload will contain the information needed  
293 by the communication software to route and process this data in a standard way.

294 Operational decisions can be made by accessing the information in the SBDH and  
295 using that information to discover by which process context the business data should  
296 be driven. Routing and processing of Standard Business Documents (SBD) is  
297 facilitated regardless of whether all applications use a document driven, application  
298 programming interface (API), or agent approach. The use of logical parameters in the  
299 SBDH will minimize Trading Partner relationship management in both the Originating  
300 and Receiving organizations since the physical parameters can be derived from the  
301 values in the document.

## 302 **2.6 Stakeholders and Audience**

303 All organizations that manage infrastructure operations and business processes  
304 for various functional areas (e.g. ordering, invoicing, planning, or financial) which  
305 create, route and process Standard Business Documents can benefit from the  
306 use of the Standard Business Document Header.

307

## 308 **2.7 Document Conventions**

309 The keywords MUST, MUST NOT, REQUIRED, SHALL, SHALL NOT, SHOULD,  
310 SHOULD NOT, RECOMMENDED, MAY and OPTIONAL, when they appear in  
311 this document, are to be interpreted as described in [RFC2119] as quoted here:

- 312 • *MUST: This word, or the terms "REQUIRED" or "SHALL", means that the*  
313 *definition is an absolute requirement of the specification.*
- 314 • *MUST NOT: This phrase, or the phrase "SHALL NOT", means that the*  
315 *definition is an absolute prohibition of the specification.*
- 316 • *SHOULD: This word, or the adjective "RECOMMENDED", means that*  
317 *there may exist valid reasons in particular circumstances to ignore a*  
318 *particular item, but the full implications must be understood and carefully*  
319 *weighed before choosing a different course.*
- 320 • *SHOULD NOT: This phrase, or the phrase "NOT RECOMMENDED",*  
321 *means that there may exist valid reasons in particular circumstances when*  
322 *the particular behavior is acceptable or even useful, but the full*  
323 *implications should be understood and the case carefully weighed before*  
324 *implementing any behavior described with this label.*

- 325       • *MAY: This word, or the adjective "OPTIONAL", mean that an item is truly*  
326       *optional. One vendor may choose to include the item because a particular*  
327       *marketplace requires it or because the vendor feels that it enhances the*  
328       *product while another vendor may omit the same item. An implementation*  
329       *which does not include a particular option MUST be prepared to*  
330       *interoperate with another implementation which does include the option,*  
331       *though perhaps with reduced functionality. In the same vein an*  
332       *implementation which does include a particular option MUST be prepared*  
333       *to interoperate with another implementation which does not include the*  
334       *option (except, of course, for the feature the option provides).*

335

## 336 **3 OBJECTIVES**

337

### 338 **3.1 Requirements**

339 The objective of this specification is to define the attributes of a Standard  
340 Business Document Header. The SBDH will make it possible for originating and  
341 receiving applications to process Standard Business Documents in a way  
342 conformant to this specification. The objective of the SBDH specification is to  
343 facilitate the exchange of documents between applications in a standard way.

344 This specification will:

- 345       • Define SBDH semantics and associated values.
- 346       • Capture the details in a UN/CEFACT Modeling Methodology (UMM) logical  
347       information model for the SBDH.
- 348       • Assure the protocol independence of Message creation.
- 349       • Define standard, data driven processing and routing parameters in the SBDH.
- 350       • Define the role of the Business Information in the semantics and syntax  
351       transformation process.

352 The SBDH is a realization of the UMM meta model, with an example in XML  
353 syntax.

354

#### 355 **3.1.1 Constraints on the Standard Business Document Header**

356 When using the Standard Business Document Header, the following constraints  
357 apply to the values provided in the header:

- 358       • Independence from proprietary routing rules.
- 359       • Location transparency in all except the ultimate partner facing functions
- 360       • Addressing transparency in all except the ultimate partner facing functions

- 361       • All proprietary semantics, syntax, and formats must be transformed into  
 362       interoperable semantics and syntax.
- 363       • Protocol independence in all except the ultimate partner facing functions.  
 364

### 365   **3.2 Principles of the Standard Business Document Header**

366

367   The following table identifies the principles used to decide what kind of  
 368   information is stored in the Standard Business Document Header, and what is  
 369   not.  
 370

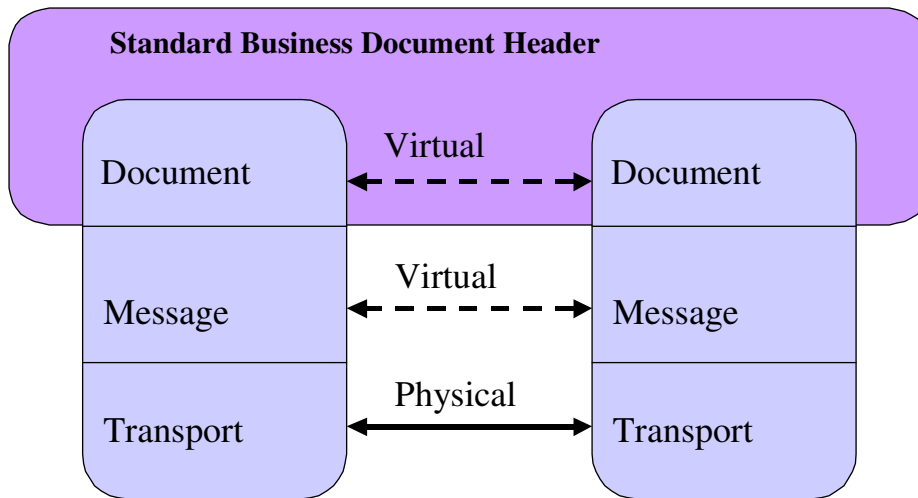
<b>IN</b>	<b>OUT</b>
1. Information known at the time of creation of the Standard Business Document (SBD) by the Business Data Creator Application (BDCA) or Translator/Parser. e.g., Standard Business Document (SBD) Type.	1. Information that can be known only at the time a message is sent. e.g., Transport Message Id.
2. Logical information that may be used to identify relevant physical information. e.g., partner name and role	2. Physical information useful for configuring the physical message transfer. e.g., channel information of partner such as protocol, port, etc. This physical information is to be extracted out of some profile, such as an OASIS CPP/A using the logical information provided.
3. Logical Information that may be used to route the document to specific external applications or services.	3. Physical Information identifying an external application such as its URL.
4. Logical Information that may be used to identify specific internal applications or services from where the document originated.	4. Physical Information identifying a specific internal application such as its IP address.

371  
 372  
 373

**Table 1**  
 In and Out Principles of the SBDH

### 374   **3.3 Layered Processing Model**

375   The layered processing model shows how the Standard Business Document  
 376   Header may be populated, extracted and processed.



377

378

**Figure 1**

379

380 An interesting Standard Business Document Header element to consider is  
 381 “Time Created” – each of the layers would have their own such element; for  
 382 example, “Document CreationDateTime”, “Message CreationDateTime”,  
 383 “Transport InitiationTime”. The Document processor at the receiving end needs  
 384 to worry or care about only the Document creation time, and not others.  
 385 However, for auditing purposes, the other information may need to be logged, but  
 386 such processing is outside the scope of SBDH.

387

### 388 3.4 Services

389

390 This section describes the use of the term “service” in the SBDH, Web Services,  
 391 and UMM Business Collaboration Framework BTV and BSV terminology from  
 392 UN/CEFACT. In the use of the SBDH, it is important to understand that the  
 393 services defined by the service information object, are different from the services  
 394 defined in ebXML and in web services. It is also important to understand that  
 395 these terms are related and that the user must ensure that the services at each  
 396 layer can map from one to the other.

397

398 EbXML Messaging Service (ebMS) and Web Services Description Language  
 399 (WSDL) both use the term “service,” but in slightly different ways. Here is a guide  
 400 to navigating the terminological differences.

401  
402 A WSDL file contains definitions and a wsdl:service is one element that can be  
403 defined. Within WSDL version 1.2, the decision has been made to have each  
404 service refer to only one wsdl:interface (formerly known as “portType”), and each  
405 wsdl:interface can aggregate one or more operations.  
406

407 ebMS does not itself define “service,” and allows for bilaterally agreed upon  
408 values for both service and its action components. However, when ebMS is used  
409 with the UN/CEFACT Business Process Specification Schema (BPSS) and  
410 OASIS CPP/A, then the values for “service” and “action” derive from values in the  
411 BPSS instance. Basically, the service value indicates the entire package of  
412 Business Processes described in a BPSS instance document. Action values  
413 identify particular requests or responses within the Business Process.  
414

415 So in both WSDL and ebMS, “service” is a kind of package of functionality, which  
416 can be defined by standards organizations or by members of a collaboration  
417 community. For ebMS, the package is of business processes, consisting of  
418 “actions”. For WSDL, the package is of elements, each called an “operation.”  
419 Operations bundle input, output, and fault definitions. Each input, output and fault  
420 at present gets associated with a “message” (and ultimately a schema defined  
421 type).  
422

423 However, the ebMS action cannot be simply equated with an operation, because  
424 each business level action at present pertains to what is in WSDL either an input  
425 or an output. So, when an interface (formerly called “portType”) has both an input  
426 and an output operation, one interface name in WSDL can pertain to what will  
427 have two action names in ebMS, the action request and action response. Despite  
428 this one terminological asymmetry, ebMS actions and WSDL operations are very  
429 similar.  
430

431 In the Standard Business Document Header, “service” is a kind of package of  
432 functionality, which is defined by standards organizations or by members of a  
433 collaboration community. It describes the business information in logical terms (it  
434 is similar to a requesting or responding business activity in BPSS or a group of  
435 operations in WSDL). However, it is not the same, because the SBDH provides a  
436 “syntax neutral” approach to facilitating the integration of the file systems of those  
437 users who need to preserve their current backend applications as they reformat  
438 their data into an XML format for transmitting it to their partners.  
439

## 440 **3.5 Routing**

441  
442 This section describes the use of the term routing at the technical messaging  
443 service level and at the Standard Business Document Header level, since the  
444 term is used differently in both of these aspects. At the business domain level,

445 which is the routing performed by the SBDH, routing describes the flow of a  
446 business document being transferred from one originating partner to another  
447 receiving partner.

448  
449 At the lower level, the technical messaging service uses predefined transfer  
450 mechanisms such as HTTP to move the data across the Internet. At the network  
451 protocol level, individual packets are transferred from one router to another  
452 across the Internet network.

453  
454 Because there are two kinds of routing - technical and business – it is useful to  
455 separate the headers into technical and business headers. The Standard  
456 Business Document Header handles business application level routing and  
457 specifying of business documents. The BCF/UMM which allows two business  
458 applications to have a virtual conversation, is another way of addressing this  
459 business need.

460  
461 Standard Business Document Header routing does not refer to the lower levels of  
462 routing as they are transparent to the SBDH. However, the routing fields in the  
463 SBDH are capable of being mapped to the technical headers so that the  
464 document can be transmitted successfully to the partner. For instance, the  
465 routing information in the SBDH contains information for Sender and Receiver in  
466 a shared, well-known format, such as, a Global Location Number (GLN) or Dun &  
467 Bradstreet's Data Universal Numbering System (DUNS) number. This  
468 information can be mapped to different technical transport header fields. These  
469 technical headers use MIME in the case of AS2, or messaging service headers in  
470 the case of ebXML Message Service (ebMS).

### 471 **3.6 Packaging**

472  
473 Since the Standard Business Document Header information is added to the  
474 business content that has been originally included in the business document, it is  
475 integral to the business document itself. It can be packaged as a part of the SBD,  
476 or for example as a separate MIME part.

477  
478 There are varied reasons why the implementer would choose an integrated  
479 packaging approach or a non-integrated approach. The following arguments  
480 favor the integrated approach:

- 481 • If the SBDH is an integral part of the XML instance document, the  
482 document can be parsed at a high level and routing and processing  
483 decisions can easily be made.
- 484 • In older systems, if the SBDH is contained in a separate MIME body part,  
485 once the message is received by the Communications application, the  
486 linkage between the two MIME body parts can be lost and the  
487 routing/processing functionality becomes more complex.

488

- 489 The next arguments favor a non-integrated (e.g. a separate MIME parts)  
490 approach:
- 491 • If the packaging is not integrated then the SBD can be easily encrypted  
492 separately from the SBDH, and the information in the SBDH can be more  
493 readily available to applications.
  - 494 • Modern middleware can handle the linking between separate MIME parts.  
495

### 496 **3.6.1 Access to the Standard Business Document Header Information** 497 **when the Payload is Encrypted**

498  
499 When using the integrated approach, once the message is inside one of the  
500 partner's firewalls, the issue of application layer security and confidentiality may  
501 arise under certain, special cases. This added concern over security and  
502 confidentiality may be an issue on the entire Standard Business Document  
503 Header and payload block or on some of the tags in the SBDH or payload.  
504 Specifically identifiers or keys or financial information are examples that may  
505 require additional security and confidentiality. The requirement may be that only  
506 certain authorized individuals have the permission to view the contents.  
507

508 For instance, a security requirement may be that the middleware environment  
509 administrators should not have visibility to the payload, which could contain  
510 sensitive trading partner data. In this requirement, only the receiving application  
511 would be able to decrypt the data, potentially long after the data transport  
512 process has ended. Some protocols may require the payload to be encrypted by  
513 the sender, prior to transport, and to remain encrypted once received. If the  
514 SBDH was received encrypted along with the payload, that would prevent further  
515 routing from occurring. In these situations, requiring strict security and  
516 confidentiality within the firewalls, there are two recommendations.  
517

518 The first is to utilize selective encryption. Selective encryption is an XML  
519 encryption option, which is available using the XML Encryption specification.  
520

521 When using the older protocols, such as PKCS7, it will be more difficult to use  
522 selective encryption. An alternative recommendation is that the SBDH is either  
523 not encrypted or decrypted upon receipt. In the case where the payload needs to  
524 be encrypted, there are two alternatives to handle this:

- 525 a) The first alternative is to send the SBDH and the attached,  
526 encrypted payload in the manifest block. Both objects are contained  
527 in one MIME part in one message.
- 528 b) The second alternative is to send the encrypted payload as a  
529 separate MIME part. This option allows multiple recipients to read  
530 the SBDH, while ensuring that only select recipients may read the  
531 sensitive contents in the payload.  
532



533 The manifest attachment is also the recommended way of sending a non-XML  
534 document or file. For example, an EDI document, with an SBDH should be sent  
535 as a manifest attachment. In this case, the non-XML payload can be encrypted  
536 and sent as the attachment, allowing the SBDH to be transported and received  
537 not encrypted or to be decrypted without impact to the rest of the payload.  
538

## 539 **4 MULTIPLE PARTNER ENVIRONMENT**

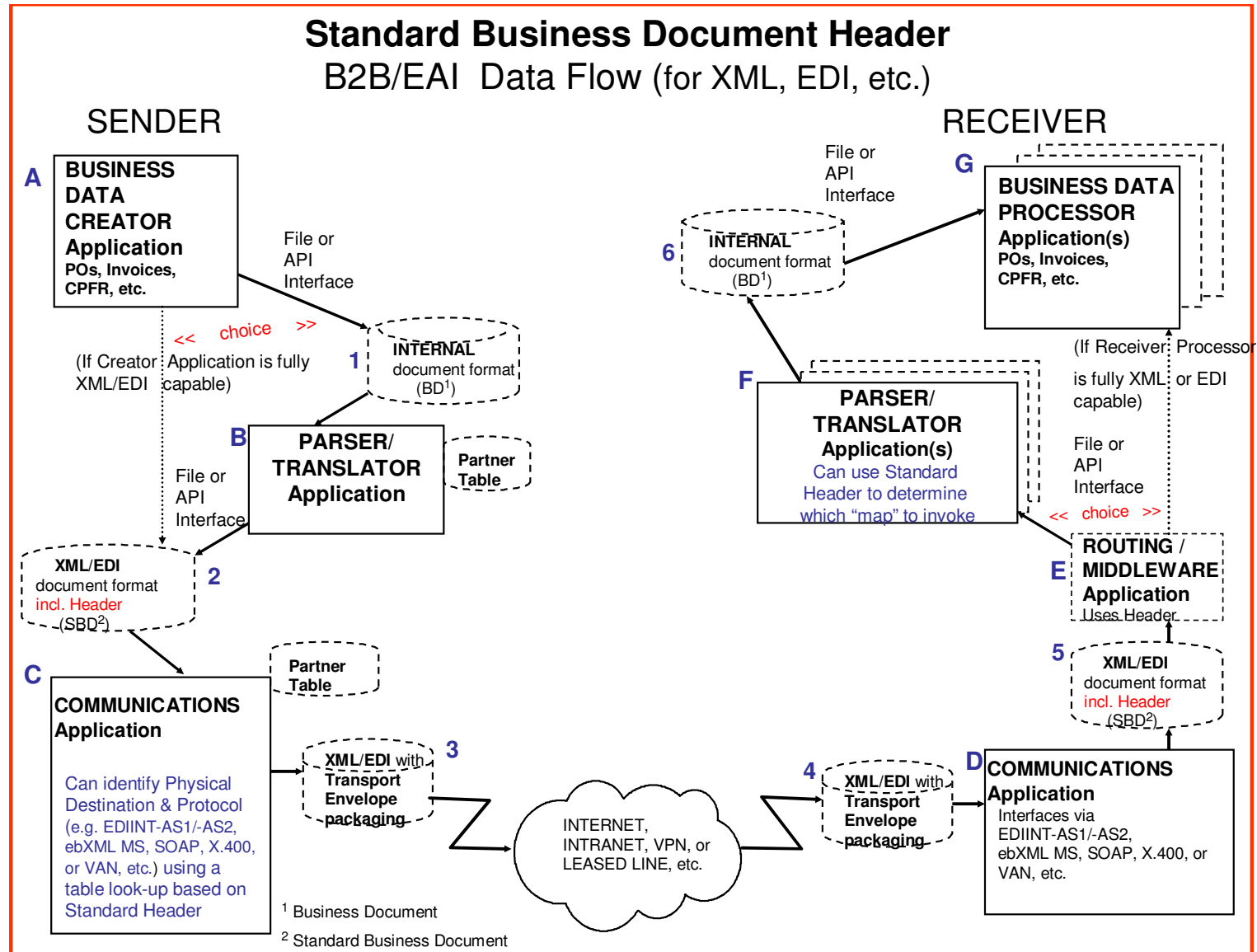
540  
541 The Standard Business Document Header could be used in the scenarios where  
542 a SBD has to be sent to multiple partners or information related to a SBD needs  
543 to be collected from multiple partners. In that case the logical Receiver value  
544 could represent a 'distribution list', and the sending Communications application  
545 could send the SBD to multiple receivers.  
546

547 The SBDH presupposes a point-to-point (sender-receiver) model. Effectively this  
548 infers that any hub-spoke or multi-party scenario will be broken down into  
549 collaborations between two partners. If it is extended to support an n-1 (hub-  
550 spokes) model, where n roles are interacting on a "business document" to do  
551 end-to-end processing, say order-to-cash, in a 'multi-hop' situation where the  
552 'middleman' strictly performs a store and forward function without changing the  
553 SBD contents, the business document creating application should be insensitive  
554 to the presence of the middleman. If the SBD is altered by an intermediate role  
555 player, the logical Recipient should be that role player, not a subsequent  
556 recipient.  
557

558 In a store and forward 'multi-hop' situation, legally relevant items such as the  
559 originator of a data message for example, may need to be retained with the  
560 identifying sender or receiver. The use of different types of technologies for  
561 example, the actions of an encryption service provider who unwraps and  
562 decrypts the message then re-encrypts it, may not preserve legally needed  
563 information that is needed when the payload arrives at the intended addressee.  
564 But by using the SBDH, the information is still preserved.

565

Figure 2



569 The figure and descriptions in this section are for illustrative purposes only, and  
570 are not normative. The various components depicted in Figure 2 are as follows:

571 **Applications A – G:**

572 Represent various applications in a data flow which move Business Documents (BD)  
573 from a Sender's back office application which creates data, to a Receiver's back  
574 office application which processes data.

575 **Data Stores 1–6:**

576 Represent various data storage locations indicating the format of data after it has  
577 been processed by one of the applications.

578 **Application A:**

579 Represents a '**Business Data Creator**' application (e.g. a legacy or ERP  
580 application) which creates business transactions for functional processes such as  
581 ordering, invoicing, planning, etc. either in:

- 582 a) Internal 'Business Document' (BD) format (shown in data store 1) e.g. a  
583 proprietary flat file which needs to be transformed into a SBD or,
- 584 b) If the creator application is fully XML or EDI capable, directly creates  
585 transactions in SBD format, including the standard header (shown in data  
586 store 2), and therefore bypasses Application B.

587 **Data Store 1:**

588 Represents one internally formatted BD which may contain one or more  
589 individual transactions of a single (or multiple closely related) business document  
590 type(s) such as purchase order, INVOIC/TAXCON, or shipment request, etc.

591 **Application B:**

592 Represents a '**Parser/Translator**' application that transforms a Business Document  
593 from its internal private format to an external Standard Business Document (SBD)  
594 format [shown in data store 2]. The SBD includes the Standard Business Document  
595 Header (SBDH). The SBDH provides logical information such as Sender, Receiver,  
596 Document Type, and optionally information such as business process identification.

597 Parser/Translator functions include optional parsing and transforming of Business  
598 Documents into standard semantics and syntax (i.e. a SBD). For example, a  
599 customer number is transformed into a Standard Partner Number, an internal stock  
600 keeping unit code is transformed into a Product Identification Number, and the  
601 structure is transformed from a proprietary flat file format into a standard format.

602 The transformation steps are optional. Not all Business Documents are created  
603 with proprietary semantics and syntax. Business Documents that are created in  
604 standard semantics or syntax will require fewer or no transformation steps.

605 **Data Store 2:**

606 Represents one externally formatted SBD, e.g. one XML instance document or  
607 one EDI interchange which includes the Standard Business Document Header.

**608 Application C:**

609 Represents a **Communications Application** that transmits the SBD from the  
610 Sender to the Receiver. The Communications Application can use logical  
611 information in the Standard Header to:

612 a) Determine the actual physical destination (i.e. where to route the SBD so  
613 that it gets to the Receiver, and

614 b) Determine the appropriate transport protocol, (e.g. ebXML MS, EDIINT-  
615 AS1/-AS2, SOAP, X.400, or a proprietary VAN protocol), managing the  
616 associated message creation, and protocol-specific envelope packaging.

617 Independence of transport protocol is provided by the syntax and protocol neutral  
618 Standard Business Document Header. Mapping of the SBDH logical values to the  
619 physical location and addressing parameters is handled by the Communications  
620 Application.

621 A Communication objective for the SBDH is to eliminate different proprietary  
622 approaches for determining transport protocol and destination. Providing a standard  
623 process will minimize the administration of Trading Partner relationships in the  
624 Communications Application by defining logical parameters in the SBDH.

**625 Data Store 3:**

626 Represents one transport message (as it is sent from Sender to Receiver) which  
627 contains the SBD plus the protocol specific envelope packaging.

**628 Data Store 4:**

629 Represents the same transport message (as it is accepted by the Receiver from the  
630 Sender.)

**631 Application D:**

632 Represents a Communications Application that receives the transport message,  
633 removes the protocol specific envelope packaging, and retrieves the SBD. The  
634 Communications Application can use information in the SBDH to determine further  
635 processing requirements.

**636 Data Store 5:**

637 Represents one externally formatted SBD, including the SBDH.

**638 Application E:**

639 Represents an optional routing and/or middleware application that uses the  
640 SBDH to determine which of several potential translator/parsers or back end  
641 applications to invoke or where to route the SBD. The application could also use  
642 the SBDH to determine Business Scope information such as Service Information  
643 and Correlation Information.

**644 Application F:**

645 Represents a 'Parser/Translator' application that transforms data from the external  
646 SBD format into a proprietary internal format. The 'Parser/Translator' can use

647 information in the SBDH to determine how to transform the SBD (i.e. which 'map'  
648 to invoke).

649 **Data Store 6:**

650 Represents one internally formatted 'Business Document' (BD) which may  
651 contain one or more individual transactions of a single (or multiple closely  
652 related) business document types(s) such as purchase order, INVOIC/TAXCON,  
653 or shipment request, etc.

654 **Application G:**

655 Represents a 'Business Data Processor' application (e.g. a legacy or ERP  
656 application) that receives data either in a Business Document, XML, or EDI format  
657 and processes business transactions.

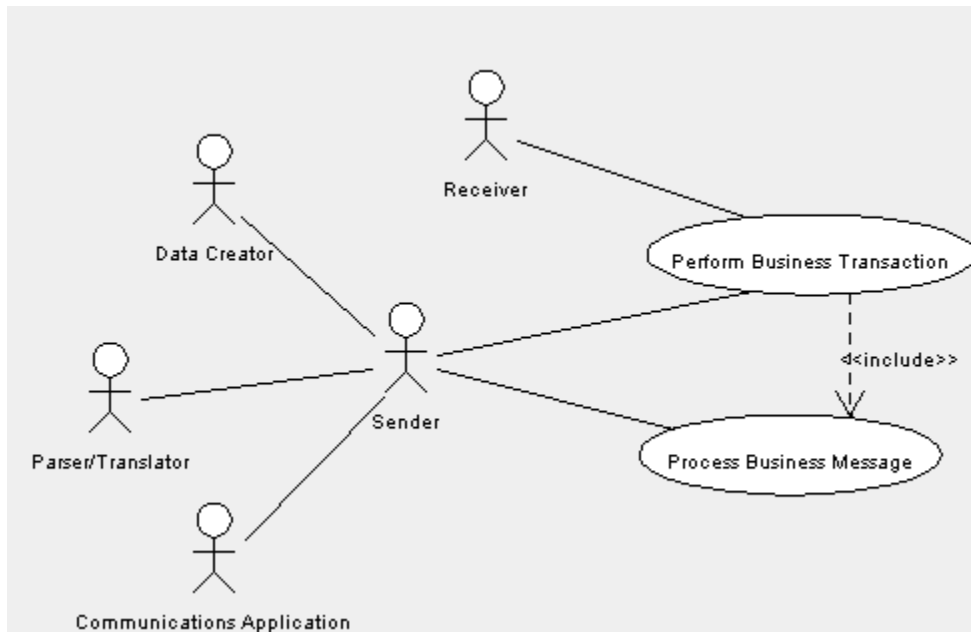
658

659 **6 Use Case Analysis**

660

661 The Standard Business Document Header is compliant to and defined by using  
662 modeling elements of the UMM-Metamodel. The UMM is part of the Business  
663 Collaboration Framework (BCF). Figure 3, below, describes the scenario that the  
664 SBDH solution addresses. Basically, two partners engage in a UMM compliant  
665 business transaction that mandates the mutual exchange of one or more  
666 business messages. These messages, in turn, must be processed for relevant  
667 business data.

668



669

670

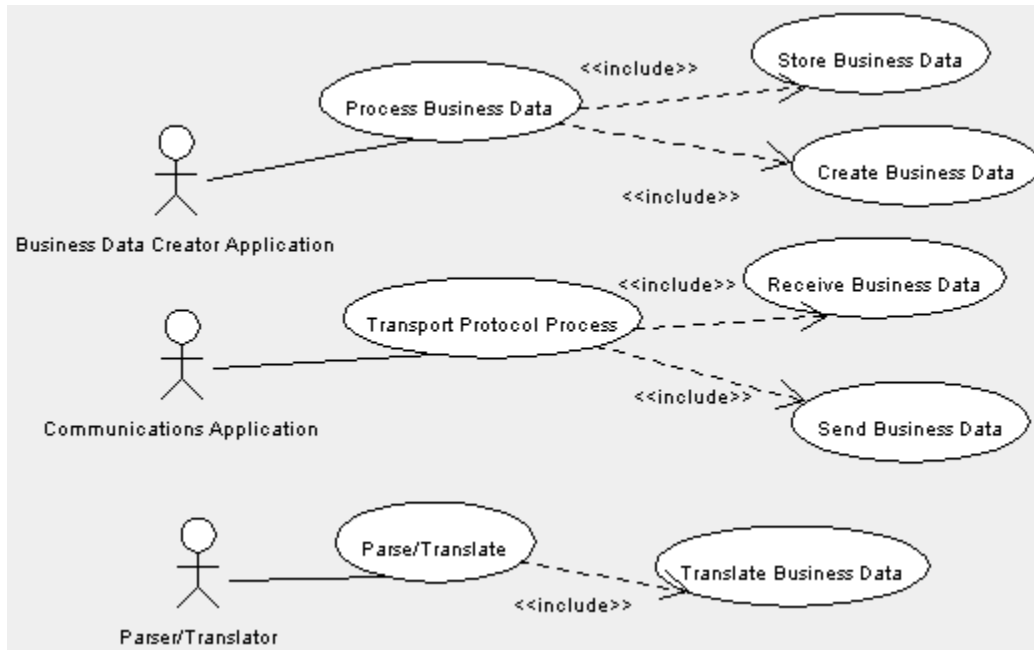
**Figure 3**

671 The use case diagram in Figure 3 illustrates the case where the Sender  
672 processes business messages, but note the receiver could follow the same

673 process being outlined. The remainder of this technical specification document  
 674 will focus on the analysis of the Sender's domain (composed of three services: a  
 675 Business Data Creator service, a Parser/Translator service and a transport or  
 676 Communications Service); and then on the analysis of the Receiver's domain  
 677 (composed of three services: a Communications Service, a Parser/Translator  
 678 service and a Business Data Processor application).

## 679 6.1 Business Services

680 The specific services addressed by the UN/CEFACT ATG SBDH Data workflow  
 681 are shown in Figure 4 below. To summarize, a Business Data Creator Service  
 682 will create a Business Document, a Parser/Translator service will transform the  
 683 Business Document into a SBD format, and a Communications Service will send  
 684 the SBD to the Receiver.  
 685



686

687

**Figure 4**

688

## 689 6.2 Description

690 Business Documents and their matching header data are created from data  
 691 residing in the private space of the sender. Therefore, the BDs may be created  
 692 using private semantics and syntax to describe and format the business data. The  
 693 BDs can be used for purposes such as creating a purchase order, or an invoice,  
 694 or some other purpose.

695 BDs can be created using:

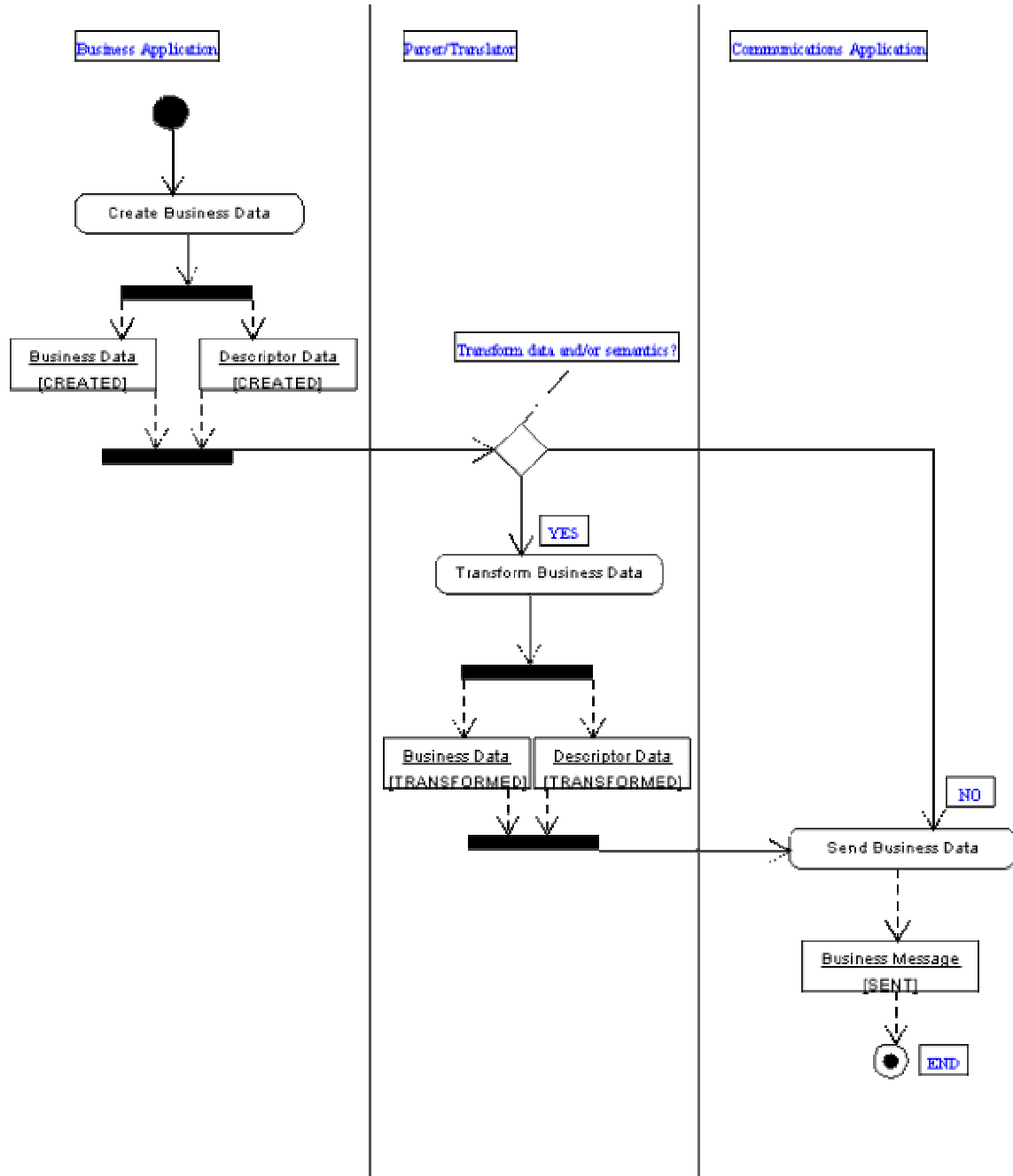
- 696 • legacy semantics
- 697 • legacy syntax
- 698 • standard semantics
- 699 • standard syntax, or
- 700 • some combination of the above.

701 The BD values will be derived from key semantics. The key semantic values must  
702 possess the intelligence required to:

- 703 • Ultimately derive the information for routing and processing the SBD.
- 704 • Map the BD logical values to the physical location and addressing parameters  
705 required by the Communications Services.
- 706 • Identify the appropriate Parser/Translator for this Business Document.  
707 Several parser/translators may exist depending upon the semantic and  
708 syntactical requirements of the BD. "Data-dependent routing" intelligence  
709 must be contained in the key values.

### 710 **6.3 Workflow Analysis**

711 There are two basic workflows for the ATG SBDH solution, each addressing a  
712 different, but complimentary, implicit UMM business function: originating and  
713 receiving business data. Figure 5, below, illustrates the prescribed ATG SBDH  
714 workflow for exchanging business data.



715

716

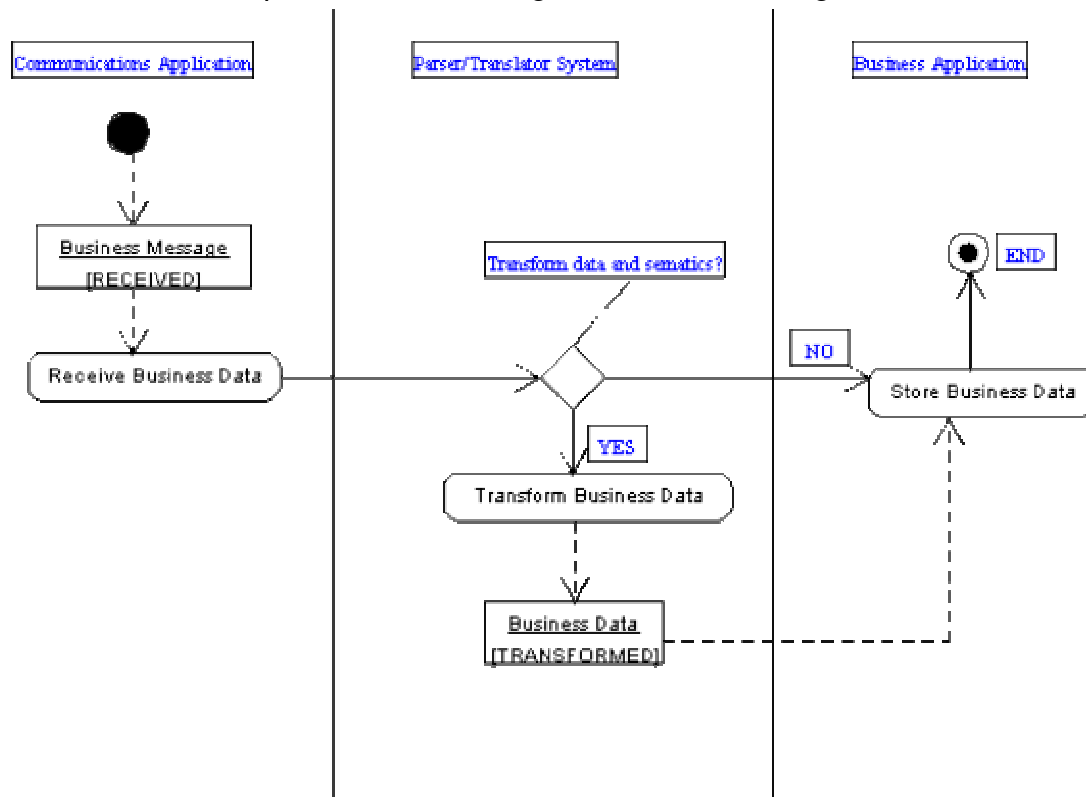
**Figure 5**

717 First, a Business Document and its matching header are created from  
 718 information residing in the private space of the sender (for example, one or more  
 719 internal business services). This data might be compliant (semantically and  
 720 syntactically) to some standard; otherwise it must undergo a data transformation  
 721 process. Note that the data and its corresponding header may initially contain  
 722 the information elements and semantics mandated by the ATG SBDH solution;  
 723 otherwise the data transformation service will ensure that such elements are



724 created. Finally, a communications service constructs a business message using  
 725 the SBD with its SBDH. This message is sent to a peer through a predefined  
 726 transport protocol.

727 The other workflow delineated by the ATG SBDH solution is shown in Figure 6  
 728 and illustrates the process of receiving a business message.



729

730

**Figure 6**

731 It is assumed that the message received by the Communications Service  
 732 contains the key data elements and semantics mandated by the ATG SBDH  
 733 solution. Key elements associated with information routing are then identified.  
 734 The message may be sent to a parser/translator service or directly to a Business  
 735 Data Processor service for processing and storage. If data transformation occurs,  
 736 certain ATG SBDH elements will facilitate the process.

737

## 738 **7 HIGH LEVEL SCENARIO**

739

740 Assumption: In order to facilitate the exchange of business information in an  
 741 electronic commerce environment, the specification addresses all the data flow in  
 742 the message creation and processing:

743

1. the creation of the content

744

2. the transformation of the content into standard form

745

3. the packaging of the content into a message

746

4. the transfer of the message

- 747 5. the receipt of the message  
 748 6. the processing of the message  
 749 7. the storing of the message.

750 The high level scenario:

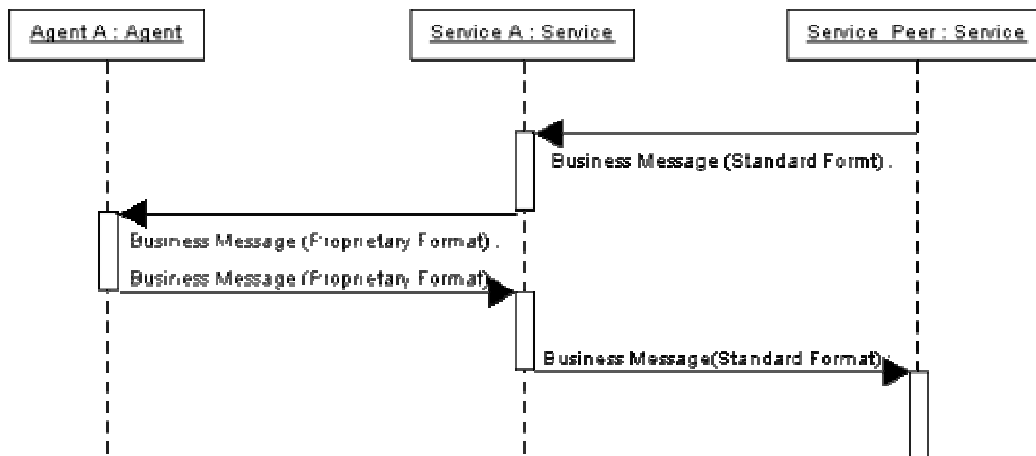
- 751 1. A BD is transformed and standardized into a SBD, e.g. standard EDI or XML  
 752 with standard semantics. Logical SBDH elements are populated with standard  
 753 semantic values.  
 754 2. The SBDH values are used to look up Message Envelope values to send the  
 755 SBD using the appropriate transport protocol.  
 756 3. The SBD is received by receiver.  
 757 4. The SBD is transformed from standard EDI or XML and standard semantics  
 758 to a proprietary BD format. Standard semantic values in the SBD are  
 759 populated with Logical BD proprietary values.  
 760 5. A Response is sent by the receiver to indicate receipt of a SBD or a rejection  
 761 indicates an exception has occurred with the sent SBD or the SBD has been  
 762 rejected by the receiver. Response must indicate acceptance or rejection of  
 763 the SBD.

764

## 765 8 PATTERNS

766

767 The UMM contains a series of message exchange patterns that rely on the  
 768 concepts of Services and Agents, where a UMM Service exchanges a SBD, via  
 769 messages, to another peer Service on behalf of an Agent.  
 770



771

772

Figure 7

773

774 In the scenario depicted in Figure 7 the Services exchange business messages  
775 which comply with some standard. A secondary role of a Service can be to  
776 communicate the SBD contained within standard business messages to a  
777 corresponding Agent in some proprietary manner. In an e-Business enterprise an  
778 Agent could represent some legacy business application while a Service could  
779 be an interface to that legacy application that communicates to other enterprises  
780 in some standard fashion. The SBDH may be used to place a business  
781 document in the proper context for the UMM/Business Collaboration Framework  
782 service layer and transaction layer.

783  
784 The ATG SBDH constructs a possible solution for a scenario that represents the  
785 UMM Service/Agent interaction patterns. It defines a generic workflow for the  
786 internal communication process between Service and Agent.

787

## 788 **9 Business Scope**

789

790 The business environment, circumstances, or scenario, in which trading partners  
791 conduct business is described by a set of domain context identifiers. This  
792 specification captures the information in the Business Scope block. The Business  
793 Scope specification being developed by the The UN/CEFACT Techniques and  
794 Methodologies Group (TMG) Unified Business Agreements and Contracts  
795 (UBAC) team. Business process information is one of the characterizations of  
796 scope about messages exchanged in a business collaboration. However, there  
797 are other relevant characterizations of scopes and contexts as well. For example,  
798 it is relevant to know which business domain the collaboration of executing  
799 messages is associated with. Scope constraints clearly identify the business  
800 domain within which the transaction is executing, providing a basis for  
801 determining which rules are applicable to the transaction. The Business Scope\*  
802 block in the SBDH provides the ability to associate a header and document with  
803 the proper business domain and thus constrain or extend its associated behavior.  
804 (\*See Appendix C for the theory behind the Business Scope.)

805

806 Scopes describe the environment within which transactions execute and allow a  
807 system to choose the correct environment. For example:

- 808 • Europe versus Asia,
- 809 • Direct-to-Consumer versus Replenishment, or
- 810 • Pre-Paid versus Credit.

811

812 Most systems, particularly legacy systems, have business domain rules coded  
813 into the application. By providing a Business Scope block in the SBDH, this  
814 information is forced up front so that all types of systems – no matter whether  
815 they are a Data Creator, a Parser/Translator or Communications Software – may  
816 select the rules correctly. The rules are selected depending on the scope  
817 received in the SBDH matched to the business domain selections within the  
818 implemented systems. When the system to be used to execute these  
819 transactions is being implemented, the implementer will write code against the

820 Business Scope and will have a very clear knowledge of which code needs to be  
821 triggered for execution of a specific domain rule.

822

823 The Business Scope in SBDH carries the information needed so that partners  
824 can identify and know which business rules to apply. There is a benefit to declare  
825 this information up front in the SBDH - partners can apply the rules even if the  
826 payload is encrypted. Knowing which of the domains the message is associated  
827 with allows business partners to make coordinated decisions for each context or  
828 business scope. For example, partners may agree that a transaction conducted  
829 with small businesses may require a credit card instead of a purchase order.  
830 The scope of that requirement constrains the business domain to be “small  
831 business”. Various scopes may select rules independently. For example, in  
832 addition to the “small business” scope, the partners may have an electronic  
833 collaboration mechanism in the form of an existing [Trading Partner Agreement](#)  
834 (TPA). The TPA identifies behaviour that is executed depending on the  
835 transaction exchange within the TPA domain. In the example, then, there are two  
836 scopes that are useful to identify the business domain of the collaboration:

- 837 • the small business domain and
- 838 • the domain of the pre-established TPA.

## 839 **9.1 Technical Agreements and Business Agreements**

840 Although partners may agree on technical agreements and pre-establish these  
841 agreements in a set-up step of the process, when it comes to business  
842 agreements, the partners’ behaviour during the collaboration runtime may vary  
843 depending on the business context being applied. This is the benefit of providing  
844 a Business Scope block in the SBDH. The required business behaviour for an  
845 exchange of messages is explicitly named in the Business Scope block. The  
846 business behaviour or relationship will vary in the instance of the transaction or  
847 collaboration. The same two partners, who submit replenishment purchase order  
848 collaborations, may exhibit similar technical behaviour but different business  
849 behavior with each other when the purchase order is Direct-to-Consumer. The  
850 business behaviour is constrained by execution of a replenishment process or  
851 direct-to-consumer process. Which business process is executing determines the  
852 scope that is associated with the business behaviour. Being able to identify  
853 business behaviour with respect to active scope allows partners to clearly identify  
854 expected business behaviour in multiple scenarios

855

856 During an exchange of data messages, a number of specifications and legal  
857 provisions govern the exact interpretation and execution of ‘Dispatch’ and  
858 ‘Reach’. Specifications and agreements on business and technical levels often  
859 form a linked documentation set where various provisions are formulated in  
860 different resources. The SBDH and BusinessScope provide the capabilities to  
861 find the starting point for such dependent documents. However the current  
862 version of SBDH supports only identification of such resources (node) and not  
863 their relationships (edges). It was deemed that specification of relationships is an

864 area that needed further consideration and elaboration. In future versions of the  
865 SBDH relationships between scopes may be defined.

866  
867 Pre-determined technical agreements describe technical protocols that partners  
868 will use when they conduct business electronically. In technical agreements,  
869 partners may decide upon using the OASIS CPP/A, a TPA, a RosettaNet PIP, or  
870 an AS2 connection. For example, a RosettaNet PIP and a CPP/A URI are used  
871 as two values in the filled out Business Scope block. This combination of PIP  
872 identification and CPA URI identifies the domain. This example is not  
873 exclusionary. The UN/CEFACT architecture describes a stack – a technical  
874 description at each layer of the stack. AS2 for example is at the bottom layer.  
875 Technical and business agreements can be declared going up the stack from  
876 AS2, following the UN/CEFACT architecture.

877  
878 The CPP/A will have elements that govern both. It contains an SLA used by  
879 ebusiness software to monitor whether a response came back in time. The  
880 RosettaNet PIP provides a set of possible values, for example, for an order type,  
881 and the translation software would use that. The PIP will translate relationship  
882 attributes based upon “roles”. In a system, every user has rights based upon their  
883 role. Access management software has information on the role the user is  
884 playing in the current domain. This could be, for example, Read, Write, or No  
885 Access to data. The combination of values in the PIP and the CPP/A will provide  
886 information to all three services in the SBDH: the Data Creator, the  
887 Parser/Translator and the Communications Software.

888  
889 It would be unrealistic to expect to renegotiate the technical agreement each time  
890 the business environment changes in some similar manner. The overhead of  
891 setting up numerous bindings and renegotiations to accommodate varied  
892 business perspectives would be prohibitive to the partners. Consider the case  
893 where a technical agreement is pre-arranged - in an existing TPA the business  
894 objective is to make deliveries from one partner to another partner's set of  
895 factories. In one particular exchange between the partners, the delivery must be  
896 made to one and only one specific factory. This specific business behavior would  
897 be accommodated using the Business Scope and the existing TPA.

898  
899 Behaviour is described by the business agreement, and then coded into the  
900 respective systems. By directly associating behaviour with scope, and then  
901 clearly identifying scope in the exchange, an agreed behaviour can be effectively  
902 triggered, monitored and enforced by the partners. They agree that when a  
903 particular value is detected in the business scope, the agreed upon business  
904 behaviour is exhibited. This behaviour is implemented in a variety of ways in the  
905 applications. The Business Scope class promotes this information up front in the  
906 partner facing part of the transaction. Most importantly, the Business Scope block  
907 makes the domain information available to both parties' systems in the same way  
908 so that both of them can make use of the information. In this way, business  
909 considerations drive the transaction via the SBDH.

910

911 In EDI, a relevant example is the Order type field in the BEG line. The Order type  
912 is used to trigger different rules depending on whether the order type indicates  
913 Replenishment or Direct-to-Consumer, for example. In this case the Order type is  
914 constraining rules by inferring the transaction is within the scope of a process.  
915 This inference can become problematic because the Order type by itself does not  
916 fully define the process. There can in fact be several different processes required  
917 to make that Order type correct. Therefore, to know the right set of rules to use,  
918 additional information in the order is required. In this example, the order itself  
919 contains the information:

920

- The Order type plus
- Dates (and whether they are *n* weeks apart and)
- Whether the transaction is executing in one country, and so on.

921

922

923

924 In contrast, the Business scope is a clear and unambiguous holder to place that  
925 information, give it a name, and present it up front so that more applications than  
926 just the Business Creator applications can make use of it. In fact, all applications  
927 participating in the SBDH scenario – the Business Creator, Parser/Translators,  
928 and Communication Software Applications – can make use of the business  
929 scope information.

930

931 The Business Scope block as defined in the SBDH is general because the ability  
932 to identify domain associations changes over time. Rather than describing  
933 discrete values such a process, industry, etc. the SBDH Business Scope actually  
934 associates a message with its domain, execution environments and constraints.  
935 The association is made with multiple domain values such as:

936

- The process the message is executing within;
- The industry constraining processor;
- And the geopolitical policies.

937

938

939

940 For this reason, the Scope block within the Business Scope is repeatable.

## 941 **9.2 Future Business Scopes**

942 The Business Scope block is used to describe the complete business  
943 environment in which the SBDH and SBD will be processed. Standards bodies  
944 addressing business concerns will come up with enumerations of supply chain  
945 processes. The UN/CEFACT Technical Business Group (TBG) and Techniques  
946 and Methodologies Group (TMG) Unified Business Agreements and Contracts  
947 (UBAC) will be some of the entities that will define codes for the Business Scope.  
948 These will be used to fill out the SBDH Scope. The standards bodies will agree  
949 on how processes can differ. They will define the different business behaviours  
950 for each domain. The groups such as TBG will provide the content for the  
951 repeatable yet unique Scope within the Business Scope. The instance of Scope  
952 will be optional and used only if one or more such instances provide value to the  
953 partners within the current domain they are executing in.

954

955 Apart from the Business Scope defined in this version of the SBDH specification,  
956 there are other types of Scopes governing the exchange of words, messages,  
957 documents and business information in general. Agreements and contracts give  
958 legality to the information exchanges and form yet another type of Scope in  
959 another business environment. Standards bodies will identify the Scopes of the  
960 behavior and their defined Scopes will impact implementation.

961

962 At the time of this specification, the defined extensions to Scopes are: Correlation  
963 and Service Information. In the future, additional scope extensions to the SBDH  
964 meta-model are probable. Business scopes such as "negotiation" may be added  
965 for example.

966

967 Another Scopes type and extension may be added to the Scope forming the  
968 concept of a Scope Profile. The Profile would contain various combinations of  
969 Scope Types and their extensions in an expression of a particular business  
970 domain within which an exchange of messages is occurring. This is described in  
971 the following figure.

972

973 Provisions are expressed in different resources that may logically and formally  
974 overlap each other. In order to achieve a clear and concise interpretation of the  
975 provisions, the dependencies between them must be exactly defined.

976 Relationships such as superiority, replacement, modification are possible  
977 relationship types.

978

### 979 **9.3 Scopes**

980 The repeatable and general Scopes within the Business Scope blocks gives a  
981 structure and provides one mechanism to implement business scope knowledge  
982 in the code and allow the system to traverse all the relevant information. The  
983 Business Scope provides a method that supports a highly scripted discovery –  
984 agreements are easier to manage up front. That is the key function of the  
985 Business Scope block. When exchanging business information, documentation  
986 of only the lowest current level of scope is required. From this information all  
987 information exchanged can be deduced.

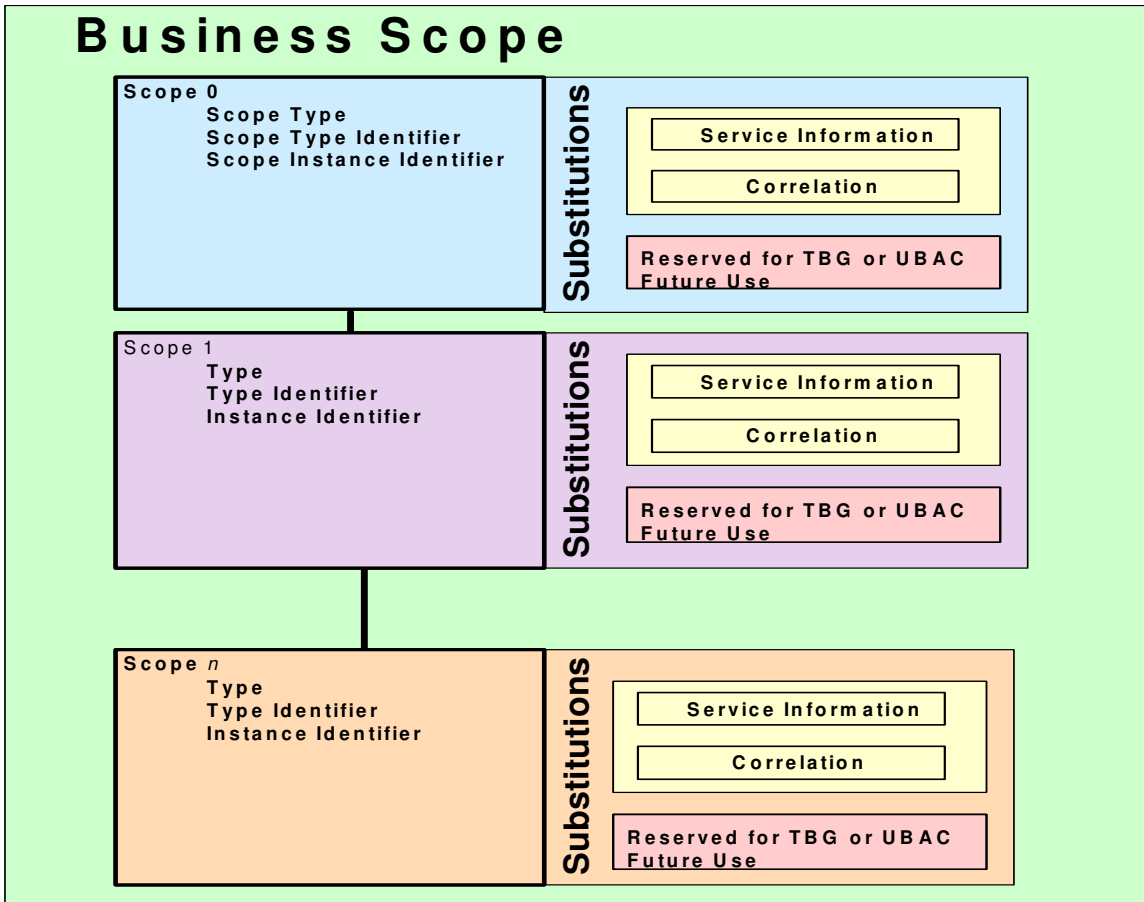
988

989 BusinessScope is a Scope reference mechanism and should not in general be  
990 used for Scope definitions. BusinessScope should be used to identify and  
991 reference the circumstances and scopes that govern a particular exchange of  
992 data messages. The referenced documents, resources, specifications etc.  
993 contain themselves complementary information relevant to the scope and  
994 information about relationships.

995

996 The BusinessScope is currently a list of governing Scopes. However such lists  
997 can handle flat structures as well as hierarchical structures (such UMM Business

998 Processes and ebXML Core Components), lattices and the more generic  
 999 directed acyclic graph structures. This is because a Scope considers the Scope  
 1000 itself and not the Scope's relationship to other Scopes (i.e. reference to a node).  
 1001 Currently, Scopes are a linear list; however, there may be a relationship shown  
 1002 between the Scopes in the future. This will be accomplished by an extension to  
 1003 this version of the SBDH specification. The structure is described in the figure  
 1004 below.  
 1005  
 1006



1007  
 1008

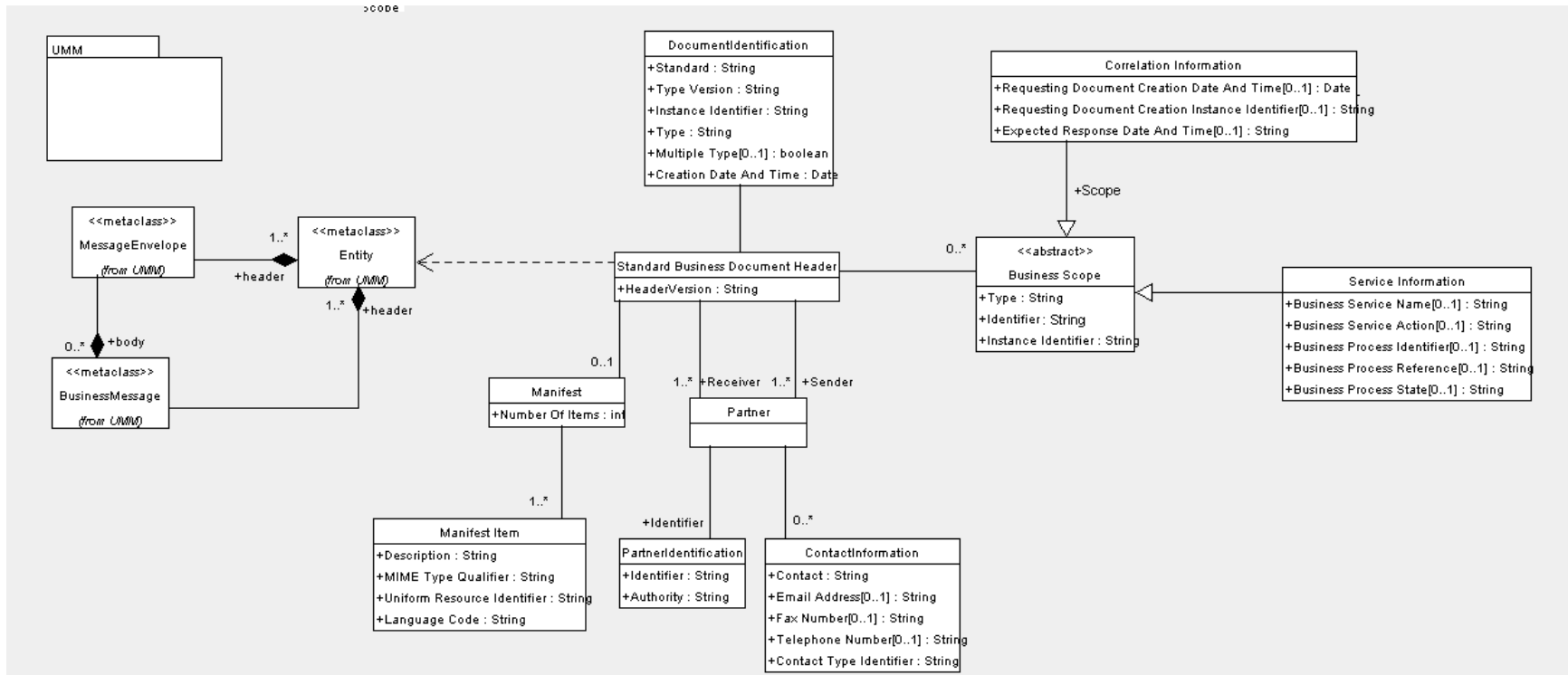
1009

Figure 8

1010  
 1011  
 1012

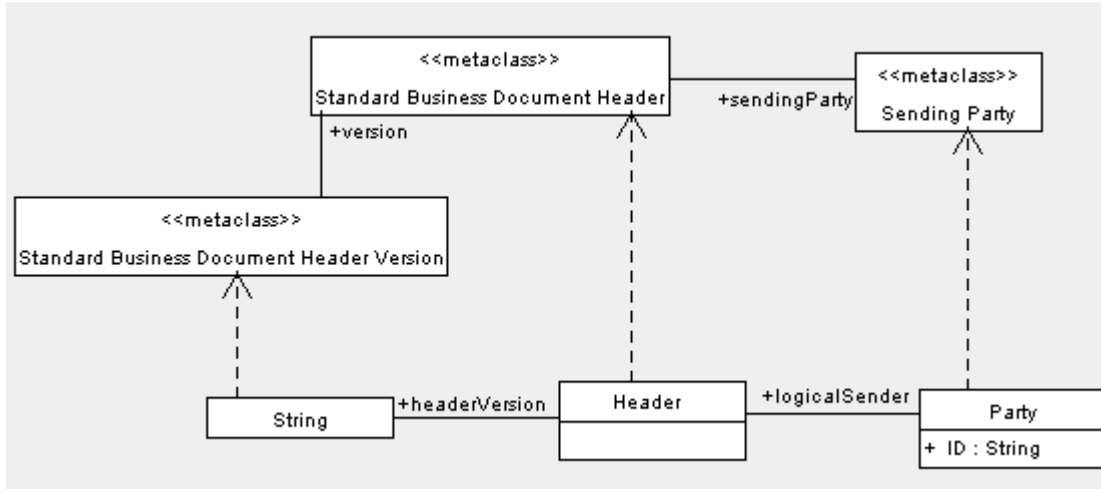


1013 10. The figure below provides the UMM meta-model for the SBDH.  
 1014



1015  
 1016 **Figure 9**  
 1017 SBDH implementation of UMM Meta-Model

1019 The next figure provides the UMM meta-class extension of the SBDH classes:  
1020



1021

1022

Figure 10

### UMM Meta-Class Extension of the SBDH Classes

1023

1024

1025

1026

## 10 Standard Business Document Header Data Elements

1027

1028

1029

1030

1031

1032

1033

1034

1035

1036

1037

1038

1039

1040

1041

1042

1043

1044

1045

1046

1047

1048

1049

The following Data Elements are components of the SBDH. The names here are the business terms, with (*proposed dictionary entry names*) in parenthesis. The proposed dictionary entry names are based on the Core Component Technical Specification Naming and Design Rules, version 2.01.

NOTE: The core components / business information entities may change after they have been processed through the UN/CEFACT harmonisation and approval process. In addition, the example schemas in Non-Normative Appendix A are for information only. These will be changed, and when published, will comply with the UN/CEFACT Naming and Design Rules and the UN/CEFACT UML to XML Transformation Rules, when available. The final version of this specification, after it has gone through the implementation verification process, will reflect these changes.

**StandardBusinessDocument** (*Standard Business Document Standard Business Document. Details*): The name of the XML tag required to wrap the SBDH and the SBD when the combined packaging into one instance file is used. This tag is only used under this packaging option, and in this case it becomes the root of the generated XML Instance Document. OPTIONAL, object.

**StandardBusinessDocumentHeader** (*Standard Business Document Standard Business Document Header. Details*): The name of the XML tag that contains the tags and contents of the SBDH. When the separate MIME part packaging

1050 approach is used this tag becomes the root of the generated XML Instance  
1051 Document. MANDATORY, object.

1052  
1053 **HeaderVersion** (*Business Document Header. Version. Identifier*): Descriptor  
1054 which contains version information for the SBDH (i.e. a number indicating the  
1055 version of the SBDH). This Header Version information is not the same as  
1056 the version information of the business document. REQUIRED, String.  
1057 **NOTE:** The HeaderVersion value is currently “1.0”. The HeaderVersion will  
1058 be updated any time that the schema defining the HeaderVersion changes.

1059  
1060 <**Sender Block**> (*Sender\_Party. Details*): Logical party representing the  
1061 organization that has created the standard business document. This block is  
1062 repeatable. If the Sender block is repeated then the first sender will be the  
1063 primary sender and the second sender will be the secondary sender. The  
1064 secondary sender will be used for internal routing purposes only to further  
1065 identify the internal routing. The primary sender is REQUIRED, object. The  
1066 secondary sender can repeat 1 to multiple times and is OPTIONAL, object.

1067  
1068 1. **Identifier** (*Sender\_Party. Identification. Identifier*): Descriptor with  
1069 information to identify this party; REQUIRED, String.  
1070 2. **Authority** (*Identification Scheme. Agency. Identifier*): Descriptor that  
1071 qualifies the identifier used to identify the sending party; REQUIRED,  
1072 String.  
1073 3. **ContactInformation** (*Sender\_Party. Contact. Contact*): Information about  
1074 the contact for this document; Can repeat 0 to multiple times.  
1075 OPTIONAL, object. Includes:  
1076 a) **Contact** (*Contact. Name. Name*): contact for business, REQUIRED,  
1077 String;  
1078 b) **EmailAddress** (*Contact. EMail Address. Text*): email address of  
1079 contact; OPTIONAL, String;  
1080 c) **FaxNumber** (*Contact. Fax Number. Text*): of contact; OPTIONAL,  
1081 String;  
1082 d) **TelephoneNumber** (*Contact. Telephone Number. Text*): of contact;  
1083 OPTIONAL, String;  
1084 e) **ContactTypeIdentifier** (*Contact. Role Identification. Identifier*): role of  
1085 the contact in this business process; OPTIONAL, String.

1086  
1087 <**Receiver Block**> (*Receiver\_Party. Details*): Logical party representing the  
1088 organization that receives the SBD. This block is repeatable. If the Receiver  
1089 block is repeated than the first receiver will be the primary receiver and the  
1090 second receiver will be the secondary receiver. The secondary receiver will  
1091 be used for internal routing purposes only to further identify the internal  
1092 routing. The primary sender is REQUIRED, object. The secondary sender  
1093 can repeat 1 to multiple times and is OPTIONAL, object.

1094  
1095

- 1096 1. **Identifier** (*Receiver\_Party. Identification. Identifier*): Descriptor with  
 1097 information to identify this party; REQUIRED, String.
- 1098 2. **Authority** (*Identification Scheme. Agency. Identifier*): Descriptor that  
 1099 qualifies the identifier used to identify the receiving party; REQUIRED,  
 1100 String. Includes:
- 1101 3. **ContactInformation** (*Receiver\_Party. Contact. Contact*): Information  
 1102 about the contact for this document; OPTIONAL, object. Can repeat 0 to  
 1103 multiple times. Includes:
- 1104 a) **Contact** (*Contact. Name. Name*): contact for business, REQUIRED,  
 1105 String;
- 1106 b) **EmailAddress** (*Contact. EMail Address. Text*): email address of  
 1107 contact; OPTIONAL, String;
- 1108 c) **FaxNumber** (*Contact. Fax Number. Text*): of contact; OPTIONAL,  
 1109 String;
- 1110 d) **TelephoneNumber** (*Contact. Telephone Number. Text*): of contact;  
 1111 OPTIONAL, String;
- 1112 e) **ContactTypeIdentifier** (*Contact. Role Identification. Identifier*): role of  
 1113 the contact in this business process; OPTIONAL, String.
- 1114
- 1115 <**DocumentIdentification** block> (*Standard Business Document. Details*)  
 1116 Characteristics containing identification about the document. REQUIRED,  
 1117 object.
- 1118
- 1119 1. **Standard** (*Standard Business Document. Standard Type. Code*): The  
 1120 originator of the type of the Business Data standard, e.g. SWIFT, OAG,  
 1121 EAN.UCC, EDIFACT, X12; references which Data Dictionary is being  
 1122 used. Used for the task of verifying that the grammar of a message is  
 1123 valid. Comment: This information may be provided in a URI if XML;  
 1124 probably not if EDI. REQUIRED, String.
- 1125 2. **TypeVersion** (*Standard Business Document. Standard Type Version.  
 1126 Identifier*): Descriptor which contains versioning information or number of  
 1127 the standard that defines the document which is specified in the 'Type'  
 1128 data element, e.g. values could be '1.3' or 'D.96A', etc. . This is the  
 1129 version of the document itself and is different than the HeaderVersion.  
 1130 REQUIRED, string.
- 1131 3. **InstanceIdentifier** (*Standard Business Document. Instance. Identifier*):  
 1132 Descriptor which contains reference information which uniquely identifies  
 1133 this instance of the SBD between the sender and the receiver. This  
 1134 identifier identifies this document as distinct from others. There is only  
 1135 one SBD instance per Standard Header. The Instance Identifier is usually  
 1136 automatically generated by the middleware. REQUIRED, string.
- 1137 4. **Type** (*Standard Business Document. Type. Code*): A logical indicator  
 1138 representing the type of Business Data being sent or the named type of  
 1139 business data. This attribute identifies the type of document and not the  
 1140 instance of that document. The instance document or interchange can  
 1141 contain one or more business documents of a single document type or

- 1142 closely related types. The industry standard body (as referenced in the  
 1143 'Standard' element) is responsible for defining the Type value to be used  
 1144 in this field (e.g. 'order', 'catalogItemNotification', 'INVOIC', etc.).  
 1145 Comment: The type may be linked to the service. REQUIRED, string.
- 1146 5. **MultipleType** (*Standard Business Document. Multiple Document Type.*  
 1147 *Indicator*): A flag to indicate that there is more than one type of Document  
 1148 in the instance. A "false" denotes that Type contains only one type of  
 1149 document; a "true" denotes that Type contains more than one type of  
 1150 document and that the name provided by the Standard authority identifies  
 1151 the multiple documents and not a single document. The instance  
 1152 document or interchange can contain one or more business documents of  
 1153 a single document type or multiple related document types. (E.g. Order,  
 1154 OrderSummary; or Invoice, TaxCon) Boolean, OPTIONAL.
- 1155 6. **CreationDateAndTime** (*Standard Business Document. Creation. Date*  
 1156 *Time*): Descriptor which contains date and time of SBDH/document  
 1157 creation. In the SBDH the parser translator or service component assigns  
 1158 the SBD a Date and Time stamp. The creation date and time expressed  
 1159 here most likely will be different from the date and time stamped in the  
 1160 transport envelope. REQUIRED, dateTime.
- 1161
- 1162 <**Manifest** block> (*Manifest. Details*): Manifest that describes the related items  
 1163 or attachments (i.e., binary files), if any, being sent in this package.  
 1164 OPTIONAL, Object.
- 1165
- 1166 1. **NumberOfItems** (*Manifest. Item Count Number. Numeric*): The count of  
 1167 number of items associated with this package. Includes the base payload  
 1168 and any attachments. REQUIRED, Integer
- 1169 2. **ManifestItem** (*Manifest. Item. Binary Object*): Provides information about  
 1170 the referenced item information; Repeatable if there is more than one item  
 1171 or attachments; REQUIRED, Object, Repeatable. Includes:
- 1172 a) **MimeTypeQualifierCode** (*Binary Object. Mime. Code*): Code  
 1173 describing whether the contents are XML or EDIFACT or X12, etc.  
 1174 syntax. Types are defined by IANA (see  
 1175 <http://www.iana.org/assignments/media-types/>) REQUIRED, String.
- 1176 b) **UniformResourceIdentifier** (*Binary Object. Uniform Resource.*  
 1177 *Identifier*): URI of the Manifest Item taken from its namespace; [For  
 1178 the useful guidance on how to reference external and internal  
 1179 message documents, the reader is referred to the RFC on Content  
 1180 Id URIs. This RFC 2392 (obsoletes 2111) can be found at the  
 1181 following location: <http://www.fags.org/rfc2392.html>];  
 1182 REQUIRED, String.
- 1183 c) **Description** (*Binary Object. Description. Text*): Text Description of  
 1184 Item; OPTIONAL, String.
- 1185 d) **LanguageCode** (*Binary Object. Language. Identifier*): Language of  
 1186 Item in ISO 639; OPTIONAL, String.
- 1187

1188  
1189 <**BusinessScope** block> (*Business Scope. Details*): The business scope  
1190 contains 1 to many [1..\*] scopes. It is not mandatory to put all intermediary  
1191 scopes in an SBDH. Only those scopes that the parties agree to are valid. The  
1192 following examples are all valid: transaction; business process; collaboration. A  
1193 Profile may be used to group well-formedness rules together. The business  
1194 scope block consists of the Scope block. OPTIONAL, Object.  
1195 1. <**Scope** block> (*Business Scope. Scope*): Indicates the type of scope,  
1196 the identifiers for the scope, other supporting information and the scope  
1197 content itself. The importance of the Scope is that it allows the SBDH to  
1198 operate under auspices of an agreement; that parties agree that they only  
1199 include reference agreements (i.e. make a reference of SBDH and  
1200 RosettaNet or OASIS CPP/A). Additional types of agreements are  
1201 expected to be defined in the future. OPTIONAL, Object.  
1202 a) **Type**: (*Business Scope. Scope Type. Code*): Indicates the kind of  
1203 scope; an attribute describing the Scope. Example entries include:  
1204 UN/CEFACT Transaction, UMM:BusinessCollaboration,  
1205 BusinessProcess, ebXML:BusinessService,  
1206 BusinessServiceAction, BCF:AuthorizedRole, or Role Party. Could  
1207 be used to indicate role reversal. MANDATORY, String.  
1208 b) **InstancelIdentifier**: (*Business Scope. Scope Instance. Identifier*):  
1209 A unique identifier that references the instance of the scope (e.g.  
1210 process execution instance, document instance). For example, the  
1211 Instance Identifier could be used to identify the specific instance of  
1212 a Business Process. This identifier would be used to correlate all  
1213 the way back to the business domain layer; it can be thought of as  
1214 a session descriptor at the business domain application level.  
1215 OPTIONAL, String.  
1216 c) **Identifier**: (*Business Scope. Scope. Identifier*) An optional unique  
1217 descriptor that identifies the "contract" or "agreement" that this  
1218 instance relates to. It operates at the level of business domain, not  
1219 at the transport or messaging level, by providing the information  
1220 necessary and sufficient to configure the service at the other  
1221 partner's end. Valid values for the Identifier may be in the form of  
1222 a: URI, URN, ebXML CPAID, RosettaNet TPA, EDIFIEC or Partner  
1223 Defined. Partners agree on how to describe the contract. A  
1224 reference to the definition of legal compliance can be used as  
1225 values in Identifier as well. It references the type of parent scope  
1226 (e.g. process model, document specification). Several methods  
1227 may be use to identify scopes: for example, Global identifiers  
1228 (GUID) , relative identifiers (role name sequence number, local  
1229 name). OPTIONAL, String.  
1230  
1231 The following objects are the first extensions of the Business Scope to be  
1232 defined:  
1233

- 1234 • the BusinessService block
- 1235 • and the CorrelationInformation block.

1236

1237 In the future, the BusinessScope block will be extended with additional business  
 1238 scope and context extensions or substitutions, as these become defined by the  
 1239 business.

1240

1241 < **BusinessService** block> (*Business Service. Details*): Initiator's description of  
 1242 the service to be carried out on the SBD by receiver. The SBDH may be used to  
 1243 place a business document in the proper context for the UMM/Business  
 1244 Collaboration Framework (BCF) service layer and transaction layer. The SBDH  
 1245 does not model the BCF environment; it places the document within the context  
 1246 of a BCF environment which is modeled elsewhere in UN/CEFACT  
 1247 specifications. As such, a particular document will be in the context of one  
 1248 service transaction and one business transaction (i.e. in two different layers of  
 1249 the stack). OPTIONAL, Object.

1250

1251 1. **BusinessServiceName** (*Business Service. Name*): Initiator's description  
 1252 of service to be carried out on the SBD by receiver. Comment: A business  
 1253 service is a network component responding to business transaction  
 1254 requests initiated by other services. It has network identity as a business  
 1255 service. Business services monitor the execution of service collaborations.  
 1256 The service protocol implemented in the SBDH operates only in the  
 1257 document layer of the e-business network; it is not concerned with  
 1258 Transport or Message Layers. In the context of an ebXML business  
 1259 process model, a service is a set of related actions for an authorized role  
 1260 within a party. OPTIONAL, String.

1261

1262 2. **ServiceTransaction** (*Business Service. Service Transaction. Name*):  
 1263 BusinessServiceTransaction is a specific instruction to be executed by the  
 1264 'BusinessServiceName' on the received Standard Business Document.  
 1265 The ServiceTransaction element identifies a process within a  
 1266 BusinessService that processes the SBD. BusinessServiceTransaction  
 1267 SHALL be unique within the Service in which it is defined. OPTIONAL,  
 1268 Object.

1269

1270 (The following elements are an expression at a business level of what  
 1271 service an application wants and should not be construed as  
 1272 instructions to an infrastructure application.)

1272

a) **TypeOfServiceTransaction** (*BusinessService.  
 1273 ServiceTransaction. TypeOfServiceTransaction. Identifier*): The  
 1274 value of the TypeOfServiceTransaction element is specified by  
 1275 UMM as: 'Requesting Service Transaction' or 'Responding Service  
 1276 Transaction'. OPTIONAL, String.

1277

b) **IsNonRepudiationRequired** (*Business Service. Service  
 1278 Transaction. Is Non Repudiation Required. Indicator*): Non-  
 1279 repudiation of origin and content means that the originator must

- 1280 digitally sign the business data and the recipient must store the  
1281 business data (including the digital signature) in its original form for  
1282 the duration mutually agreed to in a trading partner agreement.  
1283 OPTIONAL, Boolean
- 1284 c) **IsAuthenticationRequired** (*Business Service. Service*  
1285 *Transaction. Is Authentication Required, Indicator*): If  
1286 IsNonRepudiationRequired is true, this tag is superfluous.  
1287 Otherwise, the tag indicates whether the identity of the sending role  
1288 is verified. OPTIONAL, Boolean
- 1289 d) **IsNonRepudiationOfReceiptRequired** (*Business Service. Service*  
1290 *Transaction. Is Nonrepudiation Of Receipt Required. Indicator*):  
1291 Indicates that both partners agree to mutually verify receipt of  
1292 requested business data and that the receipt must be non-  
1293 reputable. OPTIONAL, Boolean
- 1294 e) **IsIntelligibleCheckRequired** (*Business Service. Service*  
1295 *Transaction. Is Intelligible Check Required. Indicator*): Both  
1296 partners agree that a responding partner role must check (e.g. via  
1297 use of a document digest) that received data is not garbled  
1298 (unreadable, unintelligible) and has integrity (i.e. has not been  
1299 altered) before acknowledgment of proper receipt is returned to the  
1300 requesting partner. OPTIONAL, Boolean
- 1301 e) **IsApplicationErrorResponseRequested** (*Business Service.*  
1302 *Service Transaction. Is Application Error Response Requested.*  
1303 *Indicator*): Both partners agree that a responding partner's  
1304 receiving business application must check for application level  
1305 errors; and if any are detected, must respond with an Error  
1306 Response Acknowledgment noting the errors detected. OPTIONAL,  
1307 Boolean
- 1308 f) **TimeToAcknowledgeReceipt** (*Business Service. Service*  
1309 *Transaction. Time To Acknowledge Receipt*): Specifies the time  
1310 period by which a Receipt Acknowledgment must be returned by  
1311 the responding partner's receiving business application. The  
1312 requesting and responding partners jointly agree on the time  
1313 period. It is measured from the time a business data request is sent  
1314 by a requesting partner until the time verification of receipt is  
1315 "properly received" by the requesting business partner. The Receipt  
1316 Acknowledgment only indicates receipt of data by the business  
1317 application; it does not indicate business acceptance of the  
1318 contents of the message. If the TimeToAcknowledgeReceipt  
1319 element is used, it indicates that a Receipt Acknowledgment is  
1320 requested. OPTIONAL, TimeExpression
- 1321 g) **TimeToAcknowledgeAcceptance** (*Business Service. Service*  
1322 *Transaction. Time To Acknowledge Acceptance*): Specifies the time  
1323 period that an Acceptance Acknowledgment (which indicates  
1324 business acceptance of the contents of the document) must be  
1325 returned by the responding role. The requesting and responding



1326 partners jointly agree on the time period. It is measured from the  
 1327 time a requesting partner sends business data until the time an  
 1328 acknowledgement of acceptance is "properly received" by the  
 1329 requesting partner. If the TimeToAcknowledgeAcceptance element  
 1330 is used, it indicates that an Acceptance Acknowledgment is  
 1331 requested. OPTIONAL, TimeExpression  
 1332 h) **TimeToPerform** (*Business Service. Service Transaction. Time To*  
 1333 *Perform*): Specifies the time period by which this transaction must  
 1334 be completed (measured from the time the business data is  
 1335 "properly received"). The requesting and responding partners jointly  
 1336 agree on the time period. OPTIONAL, TimeExpression  
 1337 i) **Recurrence** (*Business Service. Service Transaction. Recurrence*):  
 1338 OPTIONAL, Unsigned Integer  
 1339

1340 <**CorrelationInformation** block> (*Correlation. Details*): A block of information  
 1341 used to correlate a requesting SBD to a responding SBD and to the contract in  
 1342 an executing choreography. A requesting document in the choreography could  
 1343 have: no response, a notification, or a response document. Therefore, the  
 1344 requesting and responding part of the choreography is not always one unit of  
 1345 activity. Using the correlation block, parties explicitly identify the document being  
 1346 responded to, rather than having only the content of the document to identify the  
 1347 requesting document. UN/CEFACT BPSS correlates information at the  
 1348 transaction level but not at the business domain level, which is the function of this  
 1349 block. This is valuable information for both parties' business data creator  
 1350 applications to correlate their document exchanges. The requesting document is  
 1351 often, but not necessarily, the very first document in the sequence. If the  
 1352 Requesting document is being sent, some of the information in this block is not  
 1353 necessary - the block attributes are OPTIONAL, Object. Includes:

- 1354 1. **RequestingDocumentCreationDateTime** (*Correlation Requesting*  
 1355 *Document. Creation. Date Time*): Descriptor which contains date and time  
 1356 of the requesting SBDH and SBD, assigned to the requesting SBDH and  
 1357 SBD by the parser translator or service component. OPTIONAL,  
 1358 DateTime.
- 1359 2. **RequestingDocumentInstanceIdentifier** (*Correlation Requesting*  
 1360 *Document. Identification. Identifier*): Identifier of requesting SBDH and  
 1361 SBD instance. OPTIONAL, String.
- 1362 3. **ExpectedResponseDateTime** (*Correlation. Expected Response. Date*  
 1363 *Time*): Date and time when response is expected. This element could be  
 1364 populated in an initial message of a correlation sequence, and should be  
 1365 echoed back in a subsequent response. OPTIONAL, DateTime.  
 1366

## 1367 11 DETAILED USE CASE EXAMPLES

1368

1369 Note: These examples are subject to change by UN/CEFACT. Dictionary entry  
 1370 names for the core component / basic information entity names may change after

1371 they have been processed through the UN/CEFACT harmonisation / approval  
 1372 process.  
 1373

## 1374 **11.1 Use case 1. A non-ebXML environment**

### 1375 **Assumptions**

1376 • In this use case, the SBDH will be sent in a separate MIME Part from the  
 1377 rest of the payload. Therefore, the StandardBusinessDocument tag is not  
 1378 used in this example. The rest of the payload is not shown in this example.

1379 • This use case requires the use of the optional Manifest object because  
 1380 there are two attachments to be sent.

1381 • The middleware processing this use case does not require the information  
 1382 in the BusinessScope object; therefore, this information is not part of the  
 1383 payload.

1384 • In this use case 2 sender blocks and 2 receiver blocks are shown. The  
 1385 first sender is the primary used for primary routing; the second sender is  
 1386 the secondary routing sender. There may be additional sender blocks and  
 1387 they would also be used for routing purposes. This same holds true for the  
 1388 receiver.

1389 This use case shows the values that are known by the Business Data Creator in  
 1390 the first table. The second table shows the standard values after the original Data  
 1391 Creator values are transformed.

1392

1393 1) The Business Data Creator is the source of SBD creation and creates data in  
 1394 "Internal Business Document" format. The Business Data Creator application  
 1395 populates logical information only in the SBDH REQUIRED fields:

1396 The following field values are populated by the Business Creator Application.

1397 **Table 2. Business Creator Application Business Terms and Values**

Business Term		Example Value
Sender	Identifier	XYZ Retailer -12345
	Authority	XYZ Retailer
	ContactInformation	Contact Corporate Headquarters
		EmailAddress <a href="mailto:CorporateHeadquarters@XYZretailer.com">CorporateHeadquarters@XYZretailer.com</a>
		FaxNumber 1-212-555-1212
		TelephoneNumber 1-212-555-2121

		ContactTypeIdentifier	Corporate Organization
Sender	Identifier		XYZ Retailer Purchasing Department
	Authority		XYZ Retailer
	ContactInformation	Contact	John Doe
		EmailAddress	John_Doe@purchasing. XYZretailer.com
		FaxNumber	1-212-555-1213
		TelephoneNumber	1-212-555-2122
		ContactTypeIdentifier	Buyer
Receiver	Identifier		WidgetsMarket
	Authority		Widgets
	ContactInformation	Contact	Mary Smith
		EmailAddress	Mary_Smith@widgets.c om
		FaxNumber	1-312-555-1214
		TelephoneNumber	1-312-555-2125
		ContactTypeIdentifier	Seller
Receiver	Identifier		WidgetsSales-123
	Authority		Widgets
	ContactInformation	Contact	Jane Austin
		EmailAddress	Jane_Austin@widgets.c om
		FaxNumber	1-312-555-1216
		TelephoneNumber	1-312-555-2127
		ContactTypeIdentifier	Assistant Seller
Document Identification	Standard		Trade Item Information Record
	TypeVersion		2.1.3.4
	Type		Trade Item Information Record
	CreationDateAndTime		Sept. 15, 2003 at 10:00:00
Manifest	NumberOfItems		2
	ManifestItem	MIMETypeQualifierCode	video/mpeg
		UniformResourceIdentifier	http://www.widgets.com/ /ProductImage
		Description	MPEG Video Image of Product
		LanguageCode	English

1398

1399

**Table 3. Parser/Translator Transformed Business Terms**

1400

(see [Sample 1](#) in Appendix B)

Business Term	Transformed Example Value
StandardBusinessDocumentHeader	The root tag of the instance containing the SBDH information.

HeaderVersion		1.0
Sender	Identifier	690314800007
	Authority	EAN.UCC
	ContactInformation	Contact
		Corporate Headquarters
		EmailAddress
		<a href="mailto:CorporateHeadquarters@XYZretailer.com">CorporateHeadquarters@XYZretailer.com</a>
		FaxNumber
		1-212-555-1212
		TelephoneNumber
		1-212-555-2121
		ContactTypeIdentifier
		Corporate Headquarters
Sender	Identifier	6903148000008
	Authority	EAN.UCC
	ContactInformation	Contact
		John Doe
		EmailAddress
		<a href="mailto:John.Doe@purchasing.XYZretailer.com">John.Doe@purchasing.XYZretailer.com</a>
		FaxNumber
		1-212-555-1213
		TelephoneNumber
		1-212-555-2122
		ContactTypeIdentifier
		Buyer
Receiver	Identifier	2203148000007
	IdentifierAuthority	EAN.UCC
	ContactInformation	Contact
		Mary Smith
		EmailAddress
		<a href="mailto:Mary.Smith@widgets.com">Mary.Smith@widgets.com</a>
		FaxNumber
		1-312-555-1214
		TelephoneNumber
		1-312-555-2125
		ContactTypeIdentifier
		Seller
Receiver	Identifier	2203148000008
	IdentifierAuthority	EAN.UCC
	ContactInformation	Contact
		Jane Austin
		EmailAddress
		<a href="mailto:Jane.Austin@widgets.com">Jane.Austin@widgets.com</a>
		FaxNumber
		1-312-555-1216
		TelephoneNumber
		1-312-555-2127
		ContactTypeIdentifier
		Assistant Seller
Document Identification	Standard	<a href="http://www.uncouncil.org/smp/schemas/simple-eb">http://www.uncouncil.org/smp/schemas/simple-eb</a>
	TypeVersion	1.3
	Instance Identifier	100001
	Type	tradeltemDocument
	MultipleType	false
	CreationDateAndTime	2003-09-15T10:05:00Z
Manifest	NumberOfItems	2
	ManifestItem	MIMETypeQualifierCode
		video/mpeg
		UniformResourceIdentifier
		<a href="http://www.widgets.com/ProductImage">http://www.widgets.com/ProductImage</a>
		Description
		MPEG Video Image of Product
		LanguageCode
		EN

1401  
1402

\* = No transformation changes between the Business Document and the Standard Business Document

1403

1404 There are attachments to be sent along with the document; therefore the  
1405 Business Data Creator populates the optional Manifest object.

1406 2) The Business Data Creator collects the SBDH with the payload and  
1407 attachments and passes all the data to the Parser Translator.

1408 3) The Parser Translator receives the data and transforms the internal Business  
1409 Document values into external SBDH semantic and format values and syntax  
1410 and updates the Header with the new values. Logical information only is placed  
1411 in the SBDH. The following field values are populated by the translator/parser to  
1412 ensure that the values represent a well-known, shared standard. In this example,  
1413 an XML syntax will be created.

1414 Document Identification: In our example, this information is known from  
1415 the URI of the namespace, so in this case it is redundant and does not  
1416 need to be used. Still, we provide the URI as an example of the Standard.

1417 4) The Parser Translator sends the data to the Communications Application.

1418 5) The Communications Application receives the data and uses the SBDH to  
1419 determine the physical destination of the document for external routing and the  
1420 transport protocol used to move the data from the sender to the receiver.  
1421 Typically, the Communications Application uses a table to lookup the destination  
1422 and protocol.

1423 Transport Headers are created by the Communications Application from the  
1424 SBDH.

1425 Schema examples for Sample 1 are located in Appendix A. The Instance  
1426 Document is located in Appendix B.

1427

## 1428 **11.2 Use case 2. An ebXML environment**

### 1429 **Assumptions**

1430 • In this use case, the SBDH and the SBD will be packaged in one XML  
1431 instance document. Therefore, the StandardBusinessDocument tag is  
1432 used in this example. The rest of the payload is shown is a fragment of an  
1433 Order document.

1434 • This use case does not require the use of the optional Manifest object  
1435 because there are no attachments to be sent.

- 1436       • The middleware processing this use case requires that the information in  
1437       the BusinessScope object is populated.
- 1438       • In this use case there is only a primary routing for sender and a primary  
1439       routing for receiver.
- 1440       This use case shows only the standard values after the original Data Creator  
1441       values are transformed in a single table. It does not include the Business Data  
1442       Creator values.
- 1443       This example contains a requesting and a responding example, useful in  
1444       showing the use of the Business scopes.
- 1445       The roles of the Business Data Creator, Parser/Translator and Communications  
1446       Applications are the same as in the previous use case, even though the eventual  
1447       SBD contents and packaging are somewhat different.
- 1448       In this scenario, the Business Service to be carried out on the SBD is the Order-  
1449       Sell service. Order-Sell service will invoke the Original-Order action. The  
1450       Business Process that the Scopes are an instance of is the End-to-End-Order-to-  
1451       Sell-Collaboration. The definitive reference to this Business Process is found at  
1452       the location <http://www.XYZretailer.com/ProcessReference/Order-Sell/version2>.  
1453       The current state of the executing Business Process from the sender's  
1454       perspective is Pending. The receiver, having received communication of the  
1455       Service Information from the sender's perspective, will act accordingly upon  
1456       receipt of the SBD.
- 1457       Once the Document Identification and Service Information are established, the  
1458       parser/translator will use the Correlation object to establish explicit information  
1459       about the requesting SBD (which contains the SBDH). Having the information  
1460       explicitly stated allows both the sender and receiver to correlate the business  
1461       domain information as the collaboration is in the process of execution. The date  
1462       and time stamp of the Requesting SBD is: 2003-09-17T12:10:00Z as known from  
1463       the Document Identification/ CreationDateAndTime. Therefore the requesting  
1464       SBD will contain the same date and time stamp in the  
1465       RequestingDocumentCreationDateTime. Since this information is redundant in  
1466       this example, because it is the requesting example, the optional tag may be  
1467       omitted. Likewise, the CorrelationInformation/  
1468       RequestingDocumentInstanceIdentifier is the same as the Document  
1469       Identification/InstanceIdentifier in this requesting example. The response is  
1470       expected by 2003-09-22T12:10:00Z (within 5 days from the 17<sup>th</sup> of September),  
1471       and this is provided in the CorrelationInformation/ ExpectedResponseDateTime  
1472       tag. The remainder of the values for Correlation object are shown in the table  
1473       below.

1474

1475

**Table 4.**

1476

**Parser/Translator Transformed Business Terms for Requesting SBD**

1477

(see [Sample 2a](#) in Appendix B)

1478

Business Term			Transformed Example Value
StandardBusinessDocument			The root tag of the instance containing the SBDH and the SBD.
StandardBusinessDocumentHeader			The tag wrapping only the SBDH part.
HeaderVersion			1.0
Sender	Identifier		6903148000007
	Authority		EAN.UCC
	ContactInformation	Contact	John Doe
		EmailAddress	<a href="mailto:John.Doe@purchasing.XYZretailer.com">John.Doe@purchasing.XYZretailer.com</a>
		FaxNumber	1-212-555-1213
		TelephoneNumber	1-212-555-2122
		ContactTypeIdentifier	Buyer
Receiver	Identifier		2203148000007
	IdentifierAuthority		EAN.UCC
	ContactInformation	Contact	Mary Smith
		EmailAddress	<a href="mailto:Mary.Smith@widgets.com">Mary.Smith@widgets.com</a>
		FaxNumber	1-312-555-1214
		TelephoneNumber	1-312-555-2125
		ContactTypeIdentifier	Seller
Document Identification	Standard		<a href="http://www.un-council.org/smp/schemas/simpl-eb/Order">http://www.un-council.org/smp/schemas/simpl-eb/Order</a>
	TypeVersion		1.3
	Instance Identifier		100002
	Type		order
	MultipleType		false
	CreationDateAndTime		2003-09-17T12:10:00Z
BusinessScope	Scope	Type	BusinessProcess
		InstanceIdentifier	End-to-End-Order-to-Sell-Collaboration
		Identifier	<a href="http://www.XYZretailer.com/ScopeIdentifier/Order-Sell/version2-123">http://www.XYZretailer.com/ScopeIdentifier/Order-Sell/version2-123</a>
	BusinessService		
	BusinessServiceName		Order-Sell

	ServiceTransaction	Original-Order
	TypeOfServiceTransaction	RequestingServiceTransaction
	IsNonRepudiationRequired	false
	IsAuthenticationRequired	true
	IsNonRepudiationOfReceiptRequired	true
	IsIntelligibleCheckRequired	true
	IsApplicationErrorResponseRequested	true
	TimeToAcknowledgeReceipt	P12H <sup>+</sup>
	TimeToAcknowledgeAcceptance	P2D <sup>+</sup>
	TimeToPerform	P5D <sup>+</sup>
	Recurrence	3
	CorrelationInformation	
	RequestingDocumentCreationDateTime	2003-09-17T12:10:00Z
	RequestingDocumentInstanceIdentifier	100002
	ExpectedResponseDateTime	2003-09-22T12:10:00Z
Order	This sample includes a fragment of an XML Order packaged as part of the Standard Business Document	

1479 \* = No transformation changes between the Business Document and the Standard Business  
1480 Document

1481 <sup>+</sup> See W3C Datatypes specification for the duration of time format.

1482  
1483 In the Responding Document, Mary Smith is now the Sender and John Doe is  
1484 now the Receiver. The type of document is an Order Response. The Document  
1485 Identification/ InstanceIdentifier is 550001. The Document Identification/  
1486 CreationDateAndTime is May 9<sup>th</sup>, within the time allocated for a response. The  
1487 Business Scope type is a Business Process with a new Instance Identifier. The  
1488 Parent Scope is the same as the Scope for the Requesting Document.

1489  
1490 The Correlation/ CreationDateAndTime, / InstanceIdentifier and /  
1491 ExpectedResponseDateTime are not redundant in this responding example. The  
1492 same information as found in the original requesting document is placed here. If  
1493 there were several transactions in this collaboration, the original or first  
1494 requesting document information would be placed here in all the SBDH  
1495 instances. There could be several ongoing Request-Response collaborations  
1496 between the two partners. This information "correlates" this response to the  
1497 correct original request.

1498  
1499  
1500



1501

**Table 5.**

1502

**Parser/Translator Transformed Business Terms for Responding SBD.**

1503

(see [Sample 2b](#) in Appendix B)

1504

<b>Business Term</b>			<b>Transformed Example Value</b>
StandardBusinessDocument			The root tag of the instance containing the SBDH and the SBD.
StandardBusinessDocumentHeader			The tag wrapping only the SBDH part.
HeaderVersion			1.0
Sender	Identifier		2203148000007
	Authority		EAN.UCC
	ContactInformation	Contact	Mary Smith
		EmailAddress	<a href="mailto:Mary_Smith@widges.com">Mary_Smith@widges.com</a>
		FaxNumber	1-312-555-1214
		TelephoneNumber	1-312-555-2125
		ContactTypeIdentifier	Seller
Receiver	Identifier		6903148000007
	IdentifierAuthority		EAN.UCC
	ContactInformation	Contact	John Doe
		EmailAddress	<a href="mailto:John_Doe@purchasing.XYZretailer.com">John_Doe@purchasing.XYZretailer.com</a>
		FaxNumber	1-212-555-1213
		TelephoneNumber	1-212-555-2122
		ContactTypeIdentifier	John Doe
Document Identification	Standard		http://www.uc-council.org/smp/schemas/simpl-eb/Order
	TypeVersion		1.3
	Instance Identifier		550001
	Type		Order
	MultipleType		false
	CreationDateAndTime		2003-09-17T12:10:00Z

BusinessScope	Scope	Type	BusinessProcesses
		InstanceIdentifier	End-to-End-Order-to-Sell-Collaboration
		Identifier	http://www.XYZretailer.com/ScopeIdentifier/Order-Sell/version2-123
		BusinessService	
		BusinessServiceName	Order-Sell
		ServiceTransaction	Original-Order-Response
		TypeOfServiceTransaction	RespondingServiceTransaction
		IsNonRepudiationRequired	false
		IsAuthenticationRequired	true
		IsNonRepudiationOfReceiptRequired	true
		IsIntelligibleCheckRequired	true
		IsApplicationErrorResponseRequested	true
		TimeToAcknowledgeReceipt	P12H <sup>+</sup>
		TimeToAcknowledgeAcceptance	P2D <sup>+</sup>
		TimeToPerform	P5D <sup>+</sup>
		Recurrence	3
	Scope	Type	BusinessProcesses
		InstanceIdentifier	XYZ
		Identifier	BP346
		ParentScope	
		Type	BusinessProcesses
		InstanceIdentifier	ABC
		Identifier	BP345
	This is a placeholder for additional Business Scopes that will be defined by UN/CEFACT TBG, UN/CEFACT UBAC or other industry standards bodies.		
	orderResponse		This sample includes a fragment of an XML Order Response packaged as part of the Standard Business Document

1505 \* = No transformation changes between the Business Document and the Standard Business  
 1506 Document

1507 <sup>+</sup> See W3C Datatypes specification for the duration of time format.

1508

1509 Schemas for the SBDH and Sample order and order response are located in  
1510 Appendix A. Instances for Sample 2 requesting and responding documents are  
1511 located in Appendix B.

1512

1513

1514 **11.3 Use case 3. SBDH with EDI payload in an ebXML environment**

1515 **Goal of this use case scenario**

1516 This scenario shows how the SBDH will work with an EDIFACT ORDERS  
1517 message payload in a CEFAC Business Service to Business service and  
1518 Business Process. In this Use Case, an EDI message is wrapped in the SBDH,  
1519 in order to solve the problem of having no process information in EDI.

1520 This scenario will show how to use the SBDH in an ebXML scenario and also  
1521 how to help bring legacy systems forward by bringing collaborative knowledge in  
1522 conjunction with the processes to non ebXML messages, such as EDI.

1523 As an example, the following EDI messages form a process:

1524

1525 ORDERS Purchase Orders

1526 INVOIC Invoices

1527 In this scenario, those EDI messages could be handled as ebXML "business  
1528 processes".

1529 The Business Data Creator is the source of message creation and creates data  
1530 in "Internal Business Document" format. Because this is an ebXML environment,  
1531 there is reason to use the Service Information. The Business Data Creator  
1532 populates the ServiceInformation object. The Business Data Creator declares a  
1533 Business Service using the EDI processes listed above. The Business Data  
1534 Creator:

- 1535 • establishes a context for the message;
- 1536 • establishes a collaboration in which the established services are now  
1537 capable of participation. The collaboration becomes associated with the  
1538 set of information exchanges. The Business Data Creator and its partner  
1539 on the other side will associate the set of EDI messages with the  
1540 collaboration - the processes and instances of messages exchanged  
1541 within the process.

1542 The Business Data Creator sends all the data in "Internal Business Document"  
1543 format to the Middleware.

1544 The Middleware Parser Translator function receives the data and transforms the  
1545 internal Business Document values into external SBDH format values. Only  
1546 logical information is placed in the SBDH.

1547 The Parser Translator sends the data to the Communications Application.

1548 The Communications Application receives the data and uses the SBDH to  
 1549 determine the physical destination of the document (external routing) and the  
 1550 transport protocol used to move the data from the sender to the receiver.  
 1551 Typically, the Communications Application uses a table to lookup the destination  
 1552 and protocol.

1553 Transport envelope values are created by the Communications Application from  
 1554 information in the SBDH.

1555  
 1556 An example of exchanging BP state information for a group of EDI transaction  
 1557 sets forming an “Order-Sell” process follows.

1558  
 1559 Below are the SBDH fields and their data values.

1560 **Assumptions:**

- 1561
- 1562 • This use case will pass all the payload information as one instance  
 1563 document. The StandardBusinessDocument tag is used as the root.

1564  
 1565 This use case example shows only the requesting document.

1566 **Table 6. Parser/Translator Transformed Business Terms**  
 1567 [\(see Sample 3 in Appendix B\)](#)

Business Term		Transformed Example Value in its XML Representation
StandardBusinessDocument		Tag used to include the entire contents of the SBDH and the EDI Order.
StandardBusinessDocumentHeader		Tag used to wrap the contents of the SBDH
HeaderVersion		1.0
Sender	Identifier	6903148000007
	Authority	14
	ContactInformation	Contact
		John Doe
		EmailAddress
		John_Doe@XYZretailer.com
		FaxNumber
		1-212-555-1213
		TelephoneNumber
		1-212-555-2122
		ContactTypeIdentifier
		Buyer
Receiver	Identifier	2203148000007
	Authority	14
	ContactInformation	Contact
		Mary Smith

	Information		
		EmailAddress	Mary_Smith@widgets.com
		FaxNumber	1-312-555-1214
		TelephoneNumber	1-312-555-2125
		ContactTypeIdentifier	Seller
Document Identification	Standard		EDIFACT
	TypeVersion		D.96A
	InstanceIdentifier		100002
	Type		Order
	MultipleType		false
	CreationDateAndTime		2003-05-02T00:31:52Z
Business Scope	Scope	Type	BusinessProcess
		InstanceIdentifier	Order-Sell/version2-251
		Identifier	EDI Order-Sell
		BusinessService	
		BusinessServiceName	Order-Sell
		ServiceTransaction	Original-Order
		TypeOfServiceTransaction	RequestingServiceTransaction
		IsNonRepudiationRequired	false
		IsAuthenticationRequired	true
		IsNonRepudiationOfReceiptRequired	true
		IsIntelligibleCheckRequired	true
		IsApplicationErrorResponseRequested	true
		TimeToAcknowledgeReceipt	P12H <sup>+</sup>
		TimeToAcknowledgeAcceptance	P2D <sup>+</sup>
		TimeToPerform	P5D <sup>+</sup>
		Recurrence	3
	CorrelationInformation	RequestingDocumentCreationDateTime	2003-05-02T00:31:52Z
		RequestingDocumentInstanceIdentifier	100002
		ExpectedResponseDateTime	2003-05-10T00:31:52Z
Order			This sample includes an EDI Order converted to an XML String packaged as part of the Standard Business Document

--	--	--	--

1568

1569 <sup>+</sup> See W3C Datatypes specification for the duration of time format.

1570

1571 An XML instance document with an embedded EDI interchange matching Use  
1572 Case 3 can be found in Appendix B Sample 3. The EDI data could have also  
1573 been sent as an attachment in the Manifest.

1574

#### 1575 **11.4 Use of SBDH in Acknowledgement and Exception Situations**

1576 Use of the SBDH in acknowledgements and exception situations depends on the  
1577 use case. If reliable messaging is used (e.g. AS2 or ebMS), then the transport  
1578 acknowledgement signal would be part of the transport protocol. In that case, the  
1579 SBDH would not be used.

1580

1581 However, if a business application generates an acknowledgment or exception  
1582 message, then the inclusion of the SBDH would be useful. This deployment  
1583 scenario would make the responding message just like any other business  
1584 message with a SBDH included.

1585

## 1586 **12 GLOSSARY**

1587

Applied Technology Group (ATG)	The purpose of the Applied Technologies Group (ATG) is to be responsible for the creation and maintenance of the trade, business and administration document structures that are based on a specific technology or standard. The function of the ATG is the design, assembly and production of syntax specific solutions based on identified business and/or technical requirements from the empowered groups of UN/CEFACT.
BCF	UN/CEFACT Business Collaboration Framework.
Business Document (BD)	A document used by a back office application, typically expressed in a proprietary format. The BD is typically transformed into a SBD via a middleware application such as a parser or a translator.

BPSS	Business Process Specification Schema. A UN/CEFACT requirements document.
Business Data Creator	The legacy, ERP or other application that creates business transactions for functional processes, such as ordering, invoicing, etc.
Business Service Interface (BSI)	The business layer interface described in ebXML.
Collaboration-Protocol Profile / Agreement (CPP/A)	An explicit TPA format defined by OASIS.
Communications Application	The application that transmits the SBD from the Sender to the Receiver.
DUNS	The identifier within the Dun & Bradstreet Universal Numbering System for partner identification.
ebMS	The electronic business Messaging Service specified by ebXML. Also known as ebXML-MS
EDI	Electronic Data Interchange
EDIFACT	Electronic Data Interchange for Administration, Commerce and Transport
GLN	The EAN Global Location Number for partner identification.
Messaging Service Interface (MSI)	The messaging interface described in ebXML
Parser/Translator	The application that transfers BDs from internal private format to an external SBD format including the SBDH.
Standard Business Document (SBD)	A document expressed in a format approved by a standards organization such as UN/CEFACT, EAN.UCC, Rosettanet, etc. Documents are typically shared between external trading partners in a SBD format.
Standard Business Document Header (SBDH)	The business level header in a standard format as described in this document. The SBDH is



	designed to be either an integral part of a Standard Business Document, or an object associated with the Standard Business Document.
Trading Partner Agreement (TPA)	An agreement between trading partners describing how they will exchange business information.
UN/CEFACT	United Nations Centre for Trade Facilitation and Electronic Business
UMM	UN/CEFACT Modeling Methodology
WSDL	W3C Web Services Definition Language.
XML	eXtensible Markup Language

1588

1589 Copyright Statement

1590 Copyright © UN/CEFACT 2003. All Rights Reserved.

1591 This document and translations of it may be copied and furnished to others, and  
1592 derivative works that comment on or otherwise explain it or assist in its  
1593 implementation may be prepared, copied, published and distributed, in whole or  
1594 in part, without restriction of any kind, provided that the above copyright notice  
1595 and this paragraph are included on all such copies and derivative works.  
1596 However, this document itself may not be modified in any way, such as by  
1597 removing the copyright notice or references to UN/CEFACT except as required to  
1598 translate it into languages other than English.

1599 The limited permissions granted above are perpetual and will not be revoked by  
1600 UN/CEFACT or its successors or assigns.

1601 This document and the information contained herein is provided on an "AS IS"  
1602 basis and UN/CEFACT DISCLAIMS ALL WARRANTIES, EXPRESS OR  
1603 IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE  
1604 USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR  
1605 ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A  
1606 PARTICULAR PURPOSE.

## 1607 Appendix A SBDH Schemas

1608

1609 NOTE: The example schemas in Appendix A are Non-Normative and are for  
 1610 information only. These will be changed, and when published, will comply with  
 1611 the UN/CEFACT Naming and Design Rules and the UN/CEFACT UML to XML  
 1612 Transformation Rules, when available.

### 1613 A.1 BasicTypes.xsd

1614

```
1615 <?xml version="1.0"?>
```

```
1616 <xs:schema
```

```
1617   targetNamespace="http://www.unece.org/cefact/namespaces/StandardBusiness
1618   DocumentHeader"
```

```
1619   xmlns="http://www.unece.org/cefact/namespaces/StandardBusinessDocumentH
1620   eader" xmlns:xs="http://www.w3.org/2001/XMLSchema"
```

```
1621   elementFormDefault="qualified" attributeFormDefault="unqualified">
```

```
1622     <xs:simpleType name="MimeTypeQualifier">
```

```
1623       <xs:annotation>
```

```
1624         <xs:documentation>The MIME type as defined by IANA. Please refer to
1625         http://www.iana.org/assignments/media-types/ for a list of types.
```

```
1626       </xs:documentation>
```

```
1627     </xs:annotation>
```

```
1628     <xs:restriction base="xs:string"/>
```

```
1629   </xs:simpleType>
```

```
1630   <xs:simpleType name="Language">
```

```
1631     <xs:annotation>
```

```
1632       <xs:documentation>ISO 639-2; 1998 representation of Language name.
1633       Refer to http://www.loc.gov/standards/iso639-2/iso639jac.html to get the latest
1634       version of the standard.
```

```
1635     </xs:documentation>
```

```
1636   </xs:annotation>
```

```
1637   <xs:restriction base="xs:string"/>
```

```
1638 </xs:simpleType>
```

```
1639 </xs:schema>
```

1640

### 1641 A.2 BusinessScope.xsd

1642

```
1643 <?xml version="1.0"?>
```

```
1644 <xs:schema
```

```
1645   targetNamespace="http://www.unece.org/cefact/namespaces/StandardBusiness
1646   DocumentHeader" xmlns:xs="http://www.w3.org/2001/XMLSchema"
```

```
1647 xmlns="http://www.unece.org/cefact/namespaces/StandardBusinessDocumentH
1648 eader" elementFormDefault="qualified" attributeFormDefault="unqualified">
1649   <xs:complexType name="BusinessScope">
1650     <xs:sequence>
1651       <xs:element name="Scope" type="Scope" minOccurs="0"
1652 maxOccurs="unbounded"/>
1653     </xs:sequence>
1654   </xs:complexType>
1655   <xs:complexType name="Scope">
1656     <xs:sequence>
1657       <xs:group ref="ScopeAttributes"/>
1658       <xs:element ref="ScopeInformation" minOccurs="0"
1659 maxOccurs="unbounded"/>
1660     </xs:sequence>
1661   </xs:complexType>
1662   <xs:group name="ScopeAttributes">
1663     <xs:sequence>
1664       <xs:element name="Type" type="xs:string"/>
1665       <xs:element name="InstanceIdentifier" type="xs:string"/>
1666       <xs:element name="Identifier" type="xs:string" minOccurs="0"/>
1667     </xs:sequence>
1668   </xs:group>
1669   <xs:element name="ScopeInformation" type="xs:anyType" abstract="true"/>
1670   <xs:element name="CorrelationInformation" type="CorrelationInformation"
1671 substitutionGroup="ScopeInformation"/>
1672   <xs:complexType name="CorrelationInformation">
1673     <xs:sequence>
1674       <xs:element name="RequestingDocumentCreationDateTime"
1675 type="xs:dateTime" minOccurs="0"/>
1676       <xs:element name="RequestingDocumentInstanceIdentifier"
1677 type="xs:string" minOccurs="0"/>
1678       <xs:element name="ExpectedResponseDateTime" type="xs:dateTime"
1679 minOccurs="0"/>
1680     </xs:sequence>
1681   </xs:complexType>
1682   <xs:element name="BusinessService" type="BusinessService"
1683 substitutionGroup="ScopeInformation"/>
1684   <xs:complexType name="BusinessService">
1685     <xs:sequence>
1686       <xs:element name="BusinessServiceName" type="xs:string"
1687 minOccurs="0"/>
1688       <xs:element name="ServiceTransaction" type="ServiceTransaction"
1689 minOccurs="0"/>
1690     </xs:sequence>
1691   </xs:complexType>
1692   <xs:complexType name="ServiceTransaction">
```

```

1693     <xs:attribute name="TypeOfServiceTransaction"
1694 type="TypeOfServiceTransaction" use="optional"/>
1695     <xs:attribute name="IsNonRepudiationRequired" type="xs:string"/>
1696     <xs:attribute name="IsAuthenticationRequired" type="xs:string"/>
1697     <xs:attribute name="IsNonRepudiationOfReceiptRequired"
1698 type="xs:string"/>
1699     <xs:attribute name="IsIntelligibleCheckRequired" type="xs:string"/>
1700     <xs:attribute name="IsApplicationErrorResponseRequested"
1701 type="xs:string"/>
1702     <xs:attribute name="TimeToAcknowledgeReceipt" type="xs:string"/>
1703     <xs:attribute name="TimeToAcknowledgeAcceptance" type="xs:string"/>
1704     <xs:attribute name="TimeToPerform" type="xs:string"/>
1705     <xs:attribute name="Recurrence" type="xs:string"/>
1706 </xs:complexType>
1707 <xs:simpleType name="TypeOfServiceTransaction">
1708   <xs:restriction base="xs:string">
1709     <xs:enumeration value="RequestingServiceTransaction"/>
1710     <xs:enumeration value="RespondingServiceTransaction"/>
1711   </xs:restriction>
1712 </xs:simpleType>
1713 </xs:schema>
1714

```

### 1715 A.3 DocumentIdentification.xsd

```

1716
1717 <?xml version="1.0"?>
1718 <xs:schema
1719 targetNamespace="http://www.unece.org/cefact/namespaces/StandardBusiness
1720 DocumentHeader"
1721 xmlns="http://www.unece.org/cefact/namespaces/StandardBusinessDocumentH
1722 eader" xmlns:xs="http://www.w3.org/2001/XMLSchema"
1723 elementFormDefault="qualified" attributeFormDefault="unqualified">
1724   <xs:complexType name="DocumentIdentification">
1725     <xs:sequence>
1726       <xs:element name="Standard" type="xs:string"/>
1727       <xs:element name="TypeVersion" type="xs:string"/>
1728       <xs:element name="InstanceIdentifier" type="xs:string"/>
1729       <xs:element name="Type" type="xs:string"/>
1730       <xs:element name="MultipleType" type="xs:boolean" minOccurs="0"/>
1731       <xs:element name="CreationDateAndTime" type="xs:dateTime"/>
1732     </xs:sequence>
1733   </xs:complexType>
1734 </xs:schema>
1735

```

**1736 A.4 Manifest.xsd**

```
1737
1738 <?xml version="1.0"?>
1739 <xs:schema
1740   targetNamespace="http://www.unece.org/cefact/namespaces/StandardBusiness
1741   DocumentHeader"
1742   xmlns="http://www.unece.org/cefact/namespaces/StandardBusinessDocumentH
1743   eader" xmlns:xs="http://www.w3.org/2001/XMLSchema"
1744   elementFormDefault="qualified" attributeFormDefault="unqualified">
1745   <xs:include schemaLocation="BasicTypes.xsd"/>
1746   <xs:complexType name="Manifest">
1747     <xs:sequence>
1748       <xs:element name="NumberOfItems" type="xs:integer"/>
1749       <xs:element name="ManifestItem" type="ManifestItem"
1750   maxOccurs="unbounded"/>
1751     </xs:sequence>
1752   </xs:complexType>
1753   <xs:complexType name="ManifestItem">
1754     <xs:sequence>
1755       <xs:element name="MimeTypeQualifierCode"
1756   type="MimeTypeQualifier"/>
1757       <xs:element name="UniformResourceIdentifier" type="xs:anyURI"/>
1758       <xs:element name="Description" type="xs:string minOccurs="0"/>
1759       <xs:element name="LanguageCode" type="Language" minOccurs="0"/>
1760     </xs:sequence>
1761   </xs:complexType>
1762 </xs:schema>
1763
1764
```

**1765 A.5 StandardBusinessDocumentHeader.xsd**

```
1766
1767 <?xml version="1.0"?>
1768 <xs:schema
1769   targetNamespace="http://www.unece.org/cefact/namespaces/StandardBusiness
1770   DocumentHeader"
1771   xmlns="http://www.unece.org/cefact/namespaces/StandardBusinessDocumentH
1772   eader" xmlns:xs="http://www.w3.org/2001/XMLSchema"
1773   elementFormDefault="qualified" attributeFormDefault="unqualified">
1774   <xs:include schemaLocation="DocumentIdentification.xsd"/>
1775   <xs:include schemaLocation="Partner.xsd"/>
1776   <xs:include schemaLocation="Manifest.xsd"/>
1777   <xs:include schemaLocation="BusinessScope.xsd"/>
1778   <xs:complexType name="StandardBusinessDocumentHeader">
1779     <xs:sequence>
```

```

1780     <xs:element name="HeaderVersion" type="xs:string"/>
1781     <xs:element name="Sender" type="Partner" maxOccurs="unbounded"/>
1782     <xs:element name="Receiver" type="Partner" maxOccurs="unbounded"/>
1783     <xs:element name="DocumentIdentification" type="DocumentIdentification"/>
1784     <xs:element name="Manifest" type="Manifest" minOccurs="0"/>
1785     <xs:element name="BusinessScope" type="BusinessScope"
1786     minOccurs="0"/>
1787     </xs:sequence>
1788 </xs:complexType>
1789 <xs:element name="StandardBusinessDocumentHeader"
1790 type="StandardBusinessDocumentHeader"/>
1791 <xs:element name="StandardBusinessDocument"
1792 type="StandardBusinessDocument"/>
1793 <xs:complexType name="StandardBusinessDocument">
1794 <xs:sequence>
1795 <xs:element ref="StandardBusinessDocumentHeader" minOccurs="0"/>
1796 <xs:any namespace="##other" processContents="lax"/>
1797 </xs:sequence>
1798 </xs:complexType>
1799 </xs:schema>
1800
1801

```

## 1802 A.6 Partner.xsd

```

1803
1804 <?xml version="1.0"?>
1805 <xs:schema
1806 targetNamespace="http://www.unece.org/cefact/namespaces/StandardBusiness
1807 DocumentHeader"
1808 xmlns="http://www.unece.org/cefact/namespaces/StandardBusinessDocumentH
1809 eader" xmlns:xs="http://www.w3.org/2001/XMLSchema"
1810 elementFormDefault="qualified" attributeFormDefault="unqualified">
1811 <xs:complexType name="Partner">
1812 <xs:sequence>
1813 <xs:element name="Identifier" type="PartnerIdentification"/>
1814 <xs:element name="ContactInformation" type="ContactInformation"
1815 minOccurs="0" maxOccurs="unbounded"/>
1816 </xs:sequence>
1817 </xs:complexType>
1818 <xs:complexType name="PartnerIdentification">
1819 <xs:simpleContent>
1820 <xs:extension base="xs:string">
1821 <xs:attribute name="Authority" type="xs:string"/>
1822 </xs:extension>
1823 </xs:simpleContent>
1824 </xs:complexType>

```

```

1825     <xs:complexType name="ContactInformation">
1826         <xs:sequence>
1827             <xs:element name="Contact" type="xs:string"/>
1828             <xs:element name="EmailAddress" type="xs:string" minOccurs="0"/>
1829             <xs:element name="FaxNumber" type="xs:string" minOccurs="0"/>
1830             <xs:element name="TelephoneNumber" type="xs:string"
1831 minOccurs="0"/>
1832             <xs:element name="ContactTypeIdentifier" type="xs:string"
1833 minOccurs="0"/>
1834         </xs:sequence>
1835     </xs:complexType>
1836 </xs:schema>
1837

```

## 1838 A.7 Schemas for Use with Samples

1839

### 1840 A.7.1 Simulated Order.xsd for Use with Sample 2

```

1841
1842 <?xml version="1.0" encoding="UTF-8"?>
1843 <xsd:schema targetNamespace="http://www.ean-ucc.org/schemas/1.3/eanucc"
1844 xmlns="http://www.ean-ucc.org/schemas/1.3/eanucc"
1845 xmlns:xsd="http://www.w3.org/2001/XMLSchema"
1846 elementFormDefault="unqualified" attributeFormDefault="unqualified">
1847     <xsd:element name="order" type="OrderType"/>
1848     <xsd:complexType name="OrderType">
1849         <xsd:sequence>
1850             <xsd:element name="orderIdentification" type="xsd:string"/>
1851             <!-- rest of content model would go here -->
1852         </xsd:sequence>
1853     </xsd:complexType>
1854 </xsd:schema>
1855

```

### 1856 A.7.2 Simulated OrderResponse.xsd for Use with Sample 2

```

1857
1858 <?xml version="1.0" encoding="UTF-8"?>
1859 <xsd:schema targetNamespace="http://www.ean-ucc.org/schemas/1.3/eanucc"
1860 xmlns="http://www.ean-ucc.org/schemas/1.3/eanucc"
1861 xmlns:xsd="http://www.w3.org/2001/XMLSchema"
1862 elementFormDefault="unqualified" attributeFormDefault="unqualified">
1863     <xsd:element name="orderResponse" type="OrderResponseType"/>
1864     <xsd:complexType name="OrderResponseType">
1865         <xsd:sequence>
1866             <xsd:element name="orderResponseIdentification" type="xsd:string"/>

```



```
1867         <!-- rest of content model would go here -->
1868     </xsd:sequence>
1869 </xsd:complexType>
1870 </xsd:schema>
1871
1872
1873
```

### 1874 **A.7.3 Simulated OrderProxy.xsd for Use with Sample 2**

```
1875
1876 <?xml version="1.0"?>
1877 <xs:schema targetNamespace="http://www.ean-ucc.org/schemas/1.3/eanucc"
1878 xmlns:unece="http://www.unece.org/cefact/namespaces/StandardBusinessDocu
1879 mentHeader" xmlns="http://www.ean-ucc.org/schemas/1.3/eanucc"
1880 xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
1881 attributeFormDefault="unqualified">
1882     <xs:import
1883 namespace="http://www.unece.org/cefact/namespaces/StandardBusinessDocum
1884 entHeader" schemaLocation="StandardBusinessDocumentHeader.xsd"/>
1885     <xs:include schemaLocation="Order.xsd"/>
1886 </xs:schema>
1887
```

### 1888 **A.7.4 Simulated OrderResponseProxy.xsd for Use with Sample 2**

```
1889
1890 <?xml version="1.0"?>
1891 <xs:schema targetNamespace="http://www.ean-ucc.org/schemas/1.3/eanucc"
1892 xmlns:unece="http://www.unece.org/cefact/namespaces/StandardBusinessDocu
1893 mentHeader" xmlns="http://www.ean-ucc.org/schemas/1.3/eanucc"
1894 xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
1895 attributeFormDefault="unqualified">
1896     <xs:import
1897 namespace="http://www.unece.org/cefact/namespaces/StandardBusinessDocum
1898 entHeader" schemaLocation="StandardBusinessDocumentHeader.xsd"/>
1899     <xs:include schemaLocation="OrderResponse.xsd"/>
1900 </xs:schema>
1901
1902
```

### 1903 **A.7.5 Simulated EDIOrder.xsd for Use with Sample 3**

```
1904
1905 <?xml version="1.0"?>
1906 <xs:schema targetNamespace="http://www.edi-order.org/schemas"
1907 xmlns:unece="http://www.unece.org/cefact/namespaces/StandardBusinessDocu
1908 mentHeader" xmlns="http://www.edi-order.org/schemas"
```

```
1909 xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
1910 attributeFormDefault="unqualified">
1911   <xs:import
1912     namespace="http://www.unece.org/cefact/namespaces/StandardBusinessDocum
1913     entHeader" schemaLocation="StandardBusinessDocumentHeader.xsd"/>
1914   <xs:include schemaLocation="EDIOrder.xsd"/>
1915 </xs:schema>
1916
```

### 1917 **A.7.6 Simulated EDIOrderProxy.xsd for Use with Sample 3**

```
1918
1919 <?xml version="1.0"?>
1920 <xs:schema targetNamespace="http://www.edi-order.org/schemas"
1921 xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns="http://www.edi-
1922 order.org/schemas"
1923 xmlns:unece="http://www.unece.org/cefact/namespaces/StandardBusinessDocu
1924 mentHeader" elementFormDefault="qualified"
1925 attributeFormDefault="unqualified">
1926   <xs:import
1927     namespace="http://www.unece.org/cefact/namespaces/StandardBusinessDocum
1928     entHeader" schemaLocation="StandardBusinessDocumentHeader.xsd"/>
1929   <xs:include schemaLocation="EDIOrder.xsd"/>
1930 </xs:schema>
1931
1932
```

## 1933 Appendix B Sample XML Instance Files

1934

1935 NOTE: The sample XML instance files in Appendix B are Non-Normative and  
 1936 are for information only. These will be changed, and when published, will comply  
 1937 with the UN/CEFACT Naming and Design Rules and the UN/CEFACT UML to  
 1938 XML Transformation Rules, when available.

1939

### 1940 B.1 Sample 1

1941 (see Table 3 in the Use Case examples)

1942

```
1943 <?xml version="1.0" encoding="UTF-8"?>
```

```
1944 <sh:StandardBusinessDocumentHeader
```

```
1945 xmlns:sh="http://www.unece.org/cefact/namespaces/StandardBusinessDocumen
```

```
1946 tHeader" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
```

```
1947 xsi:schemaLocation="http://www.unece.org/cefact/namespaces/StandardBusines
```

```
1948 sDocumentHeader StandardBusinessDocumentHeader.xsd">
```

```
1949   <sh:HeaderVersion>1.0</sh:HeaderVersion>
```

```
1950   <sh:Sender>
```

```
1951     <sh:Identifier Authority="EAN.UCC">690314800007</sh:Identifier>
```

```
1952     <sh:ContactInformation>
```

```
1953       <sh:Contact>Corporate Headquarters</sh:Contact>
```

1954

```
1955     <sh:EmailAddress>Corporate_Headquarters@XYZretailer.com</sh:EmailAddr
```

```
1956     ess>
```

```
1957       <sh:FaxNumber>+1-212-555-1212</sh:FaxNumber>
```

```
1958       <sh:TelephoneNumber>+1-212-555-2121</sh:TelephoneNumber>
```

```
1959       <sh:ContactTypeIdentifier>Corporate
```

```
1960 Organization</sh:ContactTypeIdentifier>
```

```
1961     </sh:ContactInformation>
```

```
1962   </sh:Sender>
```

```
1963   <sh:Sender>
```

```
1964     <sh:Identifier Authority="EAN.UCC">690314800008</sh:Identifier>
```

```
1965     <sh:ContactInformation>
```

```
1966       <sh:Contact>John Doe</sh:Contact>
```

1967

```
1968     <sh:EmailAddress>John_Doe@purchasing.XYZretailer.com</sh:EmailAdres
```

```
1969     s>
```

```
1970       <sh:FaxNumber>+1-212-555-1213</sh:FaxNumber>
```

```
1971       <sh:TelephoneNumber>+1-212-555-2122</sh:TelephoneNumber>
```

```
1972       <sh:ContactTypeIdentifier>Buyer</sh:ContactTypeIdentifier>
```

```
1973     </sh:ContactInformation>
```

```
1974   </sh:Sender>
```

```

1975     <sh:Receiver>
1976         <sh:Identifier Authority="Widgets">2203148000007</sh:Identifier>
1977         <sh:ContactInformation>
1978             <sh:Contact>Mary Smith</sh:Contact>
1979             <sh:EmailAddress>Mary_Smith@widgets.com</sh:EmailAddress>
1980             <sh:FaxNumber>+1-312-555-1214</sh:FaxNumber>
1981             <sh:TelephoneNumber>+1-312-555-2125</sh:TelephoneNumber>
1982             <sh:ContactTypeIdentifier>Seller</sh:ContactTypeIdentifier>
1983         </sh:ContactInformation>
1984     </sh:Receiver>
1985     <sh:Receiver>
1986         <sh:Identifier Authority="Widgets">2203148000008</sh:Identifier>
1987         <sh:ContactInformation>
1988             <sh:Contact>Jane Austin</sh:Contact>
1989             <sh:EmailAddress>Jane_Austin@widgets.com</sh:EmailAddress>
1990             <sh:FaxNumber>+1-312-555-1216</sh:FaxNumber>
1991             <sh:TelephoneNumber>+1-312-555-2127</sh:TelephoneNumber>
1992             <sh:ContactTypeIdentifier>Assistant Seller</sh:ContactTypeIdentifier>
1993         </sh:ContactInformation>
1994     </sh:Receiver>
1995     <sh:DocumentIdentification>
1996         <sh:Standard>http://www.uc-council.org/smp/schemas/simpl-
1997     eb/Order</sh:Standard>
1998         <sh:TypeVersion>1.3</sh:TypeVersion>
1999         <sh:InstanceIdentifier>100001</sh:InstanceIdentifier>
2000         <sh:Type>tradelItemDocument</sh:Type>
2001         <sh:MultipleType>>false</sh:MultipleType>
2002         <sh:CreationDateAndTime>2003-09-
2003     15T10:05:00Z</sh:CreationDateAndTime>
2004     </sh:DocumentIdentification>
2005     <sh:Manifest>
2006         <sh:NumberOfItems>2</sh:NumberOfItems>
2007         <sh:ManifestItem>
2008
2009             <sh:MimeTypeQualifierCode>application/xml</sh:MimeTypeQualifierCode>
2010             <sh:UniformResourceIdentifier> http://www.widgets.com//ProductImage
2011     </sh:UniformResourceIdentifier>
2012             <sh>Description>MPEG Video Image of Product</sh>Description>
2013             <sh:LanguageCode>EN</sh:LanguageCode>
2014         </sh:ManifestItem>
2015     </sh:Manifest>
2016 </sh:StandardBusinessDocumentHeader>
2017
2018
2019
2020

```

2021 **B.2 Sample 2**

2022

2023 **B.2.1 Sample 2a Requesting Document**

2024 (see Table 4 in the Use Case examples)

2025

2026 `<?xml version="1.0" encoding="UTF-8"?>`2027 `<sh:StandardBusinessDocument`2028 `xmlns:sh="http://www.unece.org/cefact/namespaces/StandardBusinessDocumen`2029 `tHeader" xmlns:eanucc="http://www.ean-ucc.org/schemas/1.3/eanucc"`2030 `xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"`2031 `xsi:schemaLocation="http://www.unece.org/cefact/namespaces/StandardBusines`2032 `sDocumentHeader OrderProxy.xsd">`2033 `<sh:StandardBusinessDocumentHeader>`2034 `<sh:HeaderVersion>1.0</sh:HeaderVersion>`2035 `<sh:Sender>`2036 `<sh:Identifier Authority="EAN.UCC">690314800007</sh:Identifier>`2037 `<sh:ContactInformation>`2038 `<sh:Contact>John Doe</sh:Contact>`

2039

2040 `<sh:EmailAddress>John_Doe@purchasing.XYZretailer.com</sh:EmailAdres`2041 `s>`2042 `<sh:FaxNumber>+1-212-555-1213</sh:FaxNumber>`2043 `<sh:TelephoneNumber>+1-212-555-2122</sh:TelephoneNumber>`2044 `<sh:ContactTypeIdentifier>Buyer</sh:ContactTypeIdentifier>`2045 `</sh:ContactInformation>`2046 `</sh:Sender>`2047 `<sh:Receiver>`2048 `<sh:Identifier Authority="EAN.UCC">220314800007</sh:Identifier>`2049 `<sh:ContactInformation>`2050 `<sh:Contact>Mary Smith</sh:Contact>`2051 `<sh:EmailAddress>Mary_Smith@widgets.com</sh:EmailAddress>`2052 `<sh:FaxNumber>+1-312-555-1214</sh:FaxNumber>`2053 `<sh:TelephoneNumber>+1-312-555-2125</sh:TelephoneNumber>`2054 `<sh:ContactTypeIdentifier>Seller</sh:ContactTypeIdentifier>`2055 `</sh:ContactInformation>`2056 `</sh:Receiver>`2057 `<sh:DocumentIdentification>`2058 `<sh:Standard>http://www.uc-council.org/smp/schemas/simpl-`2059 `eb/Order</sh:Standard>`2060 `<sh:TypeVersion>1.3</sh:TypeVersion>`2061 `<sh:InstanceIdentifier>100002</sh:InstanceIdentifier>`2062 `<sh:Type>order</sh:Type>`2063 `<sh:MultipleType>>false</sh:MultipleType>`2064 `<sh:CreationDateAndTime>2003-09-`2065 `17T12:10:00Z</sh:CreationDateAndTime>`

```

2066     </sh:DocumentIdentification>
2067     <sh:BusinessScope>
2068         <sh:Scope>
2069             <sh:Type>BusinessProcess</sh:Type>
2070             <sh:InstanceIdentifier>Order-Sell/version2-
2071 123</sh:InstanceIdentifier>
2072             <sh:Identifier>Contract Order-Sell</sh:Identifier>
2073             <sh:BusinessService>
2074                 <sh:BusinessServiceName>Order-
2075 Sell</sh:BusinessServiceName>
2076                 <sh:ServiceTransaction
2077 TypeOfServiceTransaction="RequestingServiceTransaction"
2078 IsAuthenticationRequired="true" IsNonRepudiationRequired="true"
2079 IsNonRepudiationOfReceiptRequired="true" IsIntelligibleCheckRequired="true"
2080 IsApplicationErrorResponseRequested="true"
2081 TimeToAcknowledgeReceipt="P12H" TimeToAcknowledgeAcceptance="P2D"
2082 TimeToPerform="P5D" Recurrence="3"/>
2083             </sh:BusinessService>
2084             <sh:CorrelationInformation>
2085                 <sh:RequestingDocumentCreationDateTime>2003-09-
2086 17T12:10:00Z</sh:RequestingDocumentCreationDateTime>
2087
2088             <sh:RequestingDocumentInstanceIdentifier>100002</sh:RequestingDocumen
2089 tInstanceIdentifier>
2090                 <sh:ExpectedResponseDateTime>2003-09-
2091 22T12:10:00Z</sh:ExpectedResponseDateTime>
2092             </sh:CorrelationInformation>
2093         </sh:Scope>
2094     <sh:Scope>
2095         <sh:Type>BusinessProcess</sh:Type>
2096         <sh:InstanceIdentifier>XYZ</sh:InstanceIdentifier>
2097         <sh:Identifier>BP346</sh:Identifier>
2098         <!--<sh:AsYetUndefined>...</sh:AsYetUndefined-->
2099     </sh:Scope>
2100 </sh:BusinessScope>
2101 </sh:StandardBusinessDocumentHeader>
2102 <eanucc:order>
2103     <orderIdentification>5412345000013</orderIdentification>
2104     <!-- rest of order document goes here -->
2105 </eanucc:order>
2106 </sh:StandardBusinessDocument>
2107
2108

```

## 2109 B.2.2 Sample 2b Responding Document

2110 (see Table 5 in the Use Case examples)

2111

```

2112 <?xml version="1.0" encoding="UTF-8"?>
2113 <sh:StandardBusinessDocument
2114 xmlns:sh="http://www.unece.org/cefact/namespaces/StandardBusinessDocumen
2115 tHeader" xmlns:eanucc="http://www.ean-ucc.org/schemas/1.3/eanucc"
2116 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
2117 xsi:schemaLocation="http://www.unece.org/cefact/namespaces/StandardBusines
2118 sDocumentHeader OrderResponseProxy.xsd">
2119   <sh:StandardBusinessDocumentHeader>
2120     <sh:HeaderVersion>1.0</sh:HeaderVersion>
2121     <sh:Sender>
2122       <sh:Identifier Authority="EAN.UCC">220314800007</sh:Identifier>
2123       <sh:ContactInformation>
2124         <sh:Contact>Mary Smith</sh:Contact>
2125         <sh:EmailAddress>Mary_Smith@widgets.com</sh:EmailAddress>
2126         <sh:FaxNumber>+1-312-555-1214</sh:FaxNumber>
2127         <sh:TelephoneNumber>+1-312-555-2125</sh:TelephoneNumber>
2128         <sh:ContactTypeIdentifier>Seller</sh:ContactTypeIdentifier>
2129       </sh:ContactInformation>
2130     </sh:Sender>
2131     <sh:Receiver>
2132       <sh:Identifier Authority="EAN.UCC">690314800007</sh:Identifier>
2133       <sh:ContactInformation>
2134         <sh:Contact>John Doe</sh:Contact>
2135       <sh:EmailAddress>John_Doe@purchasing.XYZretailer.com</sh:EmailAdres
2136 s>
2137       <sh:FaxNumber>+1-212-555-1213</sh:FaxNumber>
2138       <sh:TelephoneNumber>+1-212-555-2122</sh:TelephoneNumber>
2139       <sh:ContactTypeIdentifier>Buyer</sh:ContactTypeIdentifier>
2140     </sh:ContactInformation>
2141   </sh:Receiver>
2142   <sh:DocumentIdentification>
2143     <sh:Standard>http://www.uc-council.org/smp/schemas/simpl-
2144 eb/OrderResponse</sh:Standard>
2145     <sh:TypeVersion>1.3</sh:TypeVersion>
2146     <sh:InstanceIdentifier>550001</sh:InstanceIdentifier>
2147     <sh:Type>OrderResponse</sh:Type>
2148     <sh:MultipleType>false</sh:MultipleType>
2149     <sh:CreationDateAndTime>2003-05-
2150 09T00:31:52Z</sh:CreationDateAndTime>
2151   </sh:DocumentIdentification>
2152   <sh:BusinessScope>
2153     <sh:Scope>
2154       <sh:Type>BusinessProcess</sh:Type>
2155       <sh:InstanceIdentifier>Order-Sell/version2-
2156 130</sh:InstanceIdentifier>

```

```

2158         <sh:Identifier>Contract Order-Sell</sh:Identifier>
2159         <sh:BusinessService>
2160             <sh:BusinessServiceName>Order-
2161 Sell</sh:BusinessServiceName>
2162             <sh:ServiceTransaction
2163 TypeOfServiceTransaction="RequestingServiceTransaction"
2164 IsAuthenticationRequired="true" IsNonRepudiationRequired="true"
2165 IsNonRepudiationOfReceiptRequired="true" IsIntelligibleCheckRequired="true"
2166 IsApplicationErrorResponseRequested="true"
2167 TimeToAcknowledgeReceipt="P12H" TimeToAcknowledgeAcceptance="P2D"
2168 TimeToPerform="P5D" Recurrence="3"/>
2169         </sh:BusinessService>
2170         <sh:CorrelationInformation>
2171             <sh:RequestingDocumentCreationDateTime>2003-05-
2172 02T00:31:52Z</sh:RequestingDocumentCreationDateTime>
2173
2174             <sh:RequestingDocumentInstanceIdentifier>100002</sh:RequestingDocumen
2175 tInstanceIdentifier>
2176             <sh:ExpectedResponseDateTime>2003-05-
2177 10T00:31:52Z</sh:ExpectedResponseDateTime>
2178         </sh:CorrelationInformation>
2179     </sh:Scope>
2180     <sh:Scope>
2181         <sh>Type>BusinessProcess</sh>Type>
2182         <sh:InstanceIdentifier>XYZ</sh:InstanceIdentifier>
2183         <sh:Identifier>BP346</sh:Identifier>
2184         <!--<sh:AsYetUndefined>...</sh:AsYetUndefined-->
2185     </sh:Scope>
2186 </sh:BusinessScope>
2187 </sh:StandardBusinessDocumentHeader>
2188 <eanucc:orderResponse>
2189
2190     <orderResponseIdentification>5412345000013</orderResponseIdentification>
2191     <!-- rest of order document goes here -->
2192 </eanucc:orderResponse>
2193 </sh:StandardBusinessDocument>
2194
2195
2196

```

### 2197 B.3 Sample 3

2198 (see Table 6 in the Use Case examples)

2199

```
2200 <?xml version="1.0" encoding="UTF-8"?>
```

```
2201 <sh:StandardBusinessDocument
```

```
2202 xmlns:sh="http://www.unece.org/cefact/namespaces/StandardBusinessDocumen
```



```

2203 tHeader" xmlns:ediorder="http://www.edi-order.org/schemas"
2204 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
2205 xsi:schemaLocation="http://www.unece.org/cefact/namespaces/StandardBusines
2206 sDocumentHeader EDIOrderProxy.xsd">
2207   <sh:StandardBusinessDocumentHeader>
2208     <sh:HeaderVersion>1.0</sh:HeaderVersion>
2209     <sh:Sender>
2210       <sh:Identifier Authority="EAN.UCC">220314800007</sh:Identifier>
2211       <sh:ContactInformation>
2212         <sh:Contact>Mary Smith</sh:Contact>
2213         <sh:EmailAddress>Mary_Smith@widgets.com</sh:EmailAddress>
2214         <sh:FaxNumber>+1-312-555-1214</sh:FaxNumber>
2215         <sh:TelephoneNumber>+1-312-555-2125</sh:TelephoneNumber>
2216         <sh:ContactTypeIdentifier>Seller</sh:ContactTypeIdentifier>
2217       </sh:ContactInformation>
2218     </sh:Sender>
2219     <sh:Receiver>
2220       <sh:Identifier Authority="EAN.UCC">690314800007</sh:Identifier>
2221       <sh:ContactInformation>
2222         <sh:Contact>John Doe</sh:Contact>
2223       </sh:ContactInformation>
2224       <sh:EmailAddress>John_Doe@purchasing.XYZretailer.com</sh:EmailAdres
2225 s>
2226         <sh:FaxNumber>+1-212-555-1213</sh:FaxNumber>
2227         <sh:TelephoneNumber>+1-212-555-2122</sh:TelephoneNumber>
2228         <sh:ContactTypeIdentifier>Buyer</sh:ContactTypeIdentifier>
2229       </sh:ContactInformation>
2230     </sh:Receiver>
2231     <sh:DocumentIdentification>
2232       <sh:Standard>http://www.uc-council.org/smp/schemas/simpl-
2233 eb/OrderResponse</sh:Standard>
2234       <sh:TypeVersion>D.96A</sh:TypeVersion>
2235       <sh:InstanceIdentifier>100003</sh:InstanceIdentifier>
2236       <sh:Type>ORDERS</sh:Type>
2237       <sh:MultipleType>false</sh:MultipleType>
2238       <sh:CreationDateAndTime>2003-05-
2239 09T00:31:52Z</sh:CreationDateAndTime>
2240     </sh:DocumentIdentification>
2241     <sh:BusinessScope>
2242       <sh:Scope>
2243         <sh:Type>BusinessProcess</sh:Type>
2244         <sh:InstanceIdentifier>Order-Sell/version2-
2245 251</sh:InstanceIdentifier>
2246         <sh:Identifier>EDI-Order-Sell</sh:Identifier>
2247         <sh:BusinessService>

```

```

2248         <sh:BusinessServiceName>Order-
2249 Sell</sh:BusinessServiceName>
2250         <sh:ServiceTransaction
2251 TypeOfServiceTransaction="RequestingServiceTransaction"
2252 IsAuthenticationRequired="true" IsNonRepudiationRequired="true"
2253 IsNonRepudiationOfReceiptRequired="true" IsIntelligibleCheckRequired="true"
2254 IsApplicationErrorResponseRequested="true"
2255 TimeToAcknowledgeReceipt="P12H" TimeToAcknowledgeAcceptance="P2D"
2256 TimeToPerform="P5D" Recurrence="3"/>
2257     </sh:BusinessService>
2258     <sh:CorrelationInformation>
2259         <sh:RequestingDocumentCreationDateTime>2003-05-
2260 02T00:31:52Z</sh:RequestingDocumentCreationDateTime>
2261
2262     <sh:RequestingDocumentInstanceIdentifier>100002</sh:RequestingDocumen
2263 tInstanceIdentifier>
2264     <sh:ExpectedResponseDateTime>2003-05-
2265 10T00:31:52Z</sh:ExpectedResponseDateTime>
2266     </sh:CorrelationInformation>
2267     </sh:Scope>
2268     </sh:BusinessScope>
2269     </sh:StandardBusinessDocumentHeader>
2270     <ediorder:Order>
2271
2272 UNB+UNOA:3+6907777000001:14+6903148000007:14+030608:2206+811'
2273     UNH+1+ORDERS:D:96A:UN'
2274     ...
2275     UNT+37+5'
2276     UNZ+5+811'
2277 </ediorder:Order>
2278 </sh:StandardBusinessDocument>
2279
2280
2281
2282 UNB+UNOA:3+6907777000001:14+6903148000007:14+030608:2206+811'
2283     UNH+1+ORDERS:D:96A:UN'
2284     ...
2285     UNT+37+5'
2286     UNZ+5+811'
2287 </ediorder:Order>
2288 </sh:StandardBusinessDocument>
2289

```

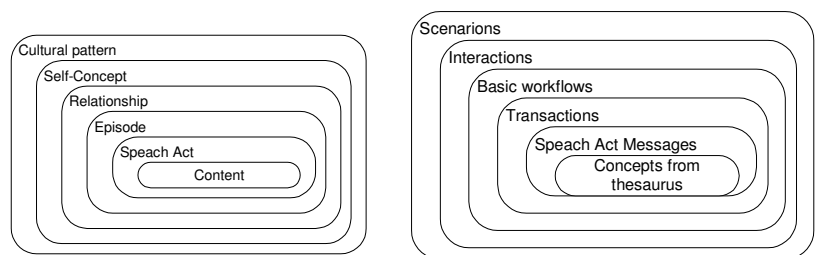
## 2290 Appendix C Theory Behind the SBDH Business Scope

2291

2292 Information about a Business Process specifies the scope or context of a single  
 2293 message exchange, however there are other types of *governing scopes* and  
 2294 contexts. Examples are TPA, economic contracts, technical agreements, and  
 2295 transaction specification. They are all governing message exchanges and are  
 2296 relevant to processing, parsing, translation, and routing etc. The following  
 2297 generalized header meta model provides for other types of business scopes and  
 2298 contexts which have business relevance to the sender and receiver.

2299

2300 From a philosophical and  
 2301 theoretical point of view, scope  
 2302 and context are commonly  
 2303 occurring. In order to interpret  
 2304 and process a message it is  
 2305 important to know in which  
 2306 business scope or context a  
 2307 business dialog is being conducted.



2308

2309

**Figure C.1.**

2310

From theory of Coordinated Management Meaning

2311

2312 The business scopes and contexts often form natural hierarchies such as  
 2313 depicted in the diagram below. Often an exchange of words or business  
 2314 information, in the world of e-business, is conducted within several contexts:

2315

2316

2317

2318

2319

2320

2321

- Within supply chains there may be business processes;
- within a process there may be several dialogs or collaborations;
- within a collaboration there may be sub collaborations;
- within a collaboration there may be multiple transactions;
- within a transaction there may be messages and signals being transmitted;
- within a message exchange there may be resending, reliability signals etc.

2322

2323

2324

2325

2326

Apart from behavioral and state scopes there are other types of governing  
 scopes and contexts in which an exchange of words, messages, documents or  
 business information may be conducted. Agreements and contracts provide legal  
 governance of information exchanges in order to satisfy the goals of business  
 relationships.

2327

2328

2329

2330

2331

2332

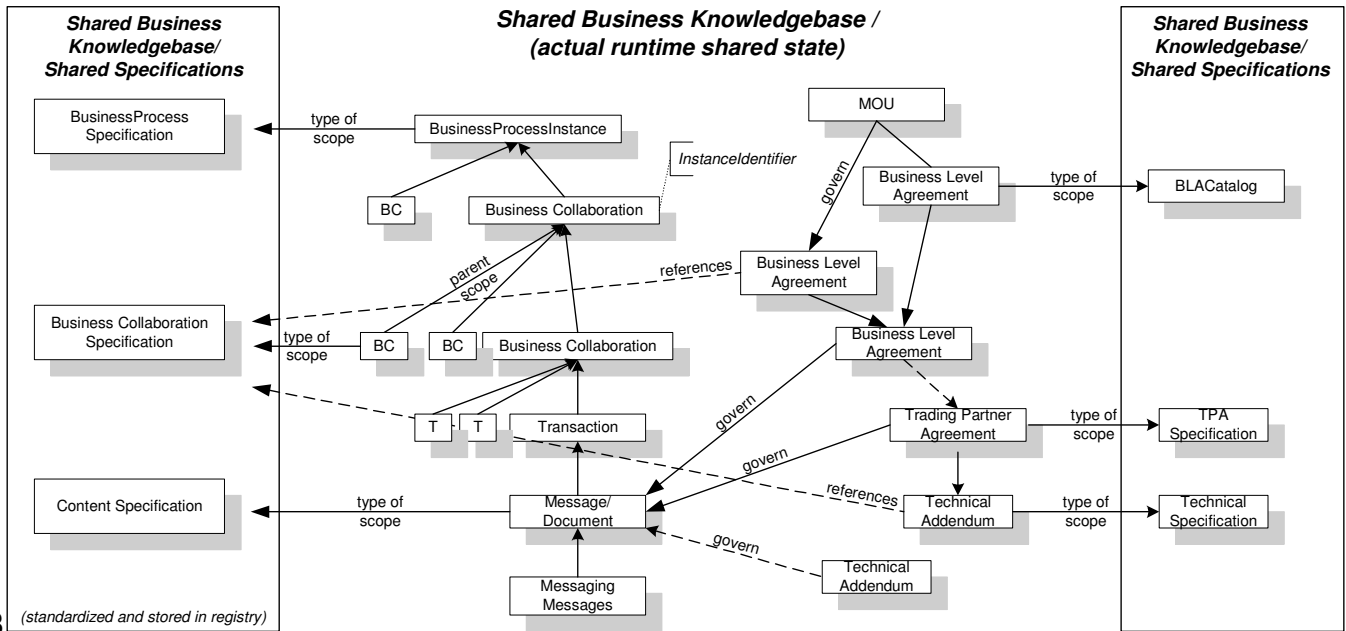
2333

An example: In order to fulfill a commitment to deliver goods, a business dialog  
 or collaboration must be defined and agreed upon. Since the parties already  
 have been engaged in electronic collaborations over an existing communication  
 channel. they may decide to reuse an existing Trading Partner Agreement, its  
 general provisions and technical details. Furthermore a generic business level  
 agreement may specify that all deliveries of a certain kind must or should be  
 made to a specific factory.

2334  
 2335  
 2336  
 2337  
 2338  
 2339  
 2340  
 2341  
 2342

It is unrealistic to prescribe that all governing details must be accessible from a single specification document, including all business and technical properties. This vision involves unnecessary bindings between the business perspective and technical details. If a delivery location is changed it should not cause a TPA to be renegotiated and agreed and vice versa.

Therefore a general and federated model based on dependencies is preferable.



2343  
 2344  
 2345  
 2346

Figure C.2. Shared Business Knowledgebase

2347 **C.1 The Commonly Occurring Perspectives of Business Scope**

2348  
 2349  
 2350  
 2351  
 2352  
 2353  
 2354  
 2355  
 2356  
 2357  
 2358  
 2359  
 2360  
 2361

There are 3 commonly occurring perspectives of scope and contexts:

1. Protocol:

When exchanging business information and documents, only the lowest level, smallest, innermost scope is needed or required. All upper level, governing parent scopes are accessed implicitly through knowledge of previously exchanged information and specifications. This view corresponds to a protocol stack where knowledge about upper layers should (must) not be required explicitly.

2. All scopes must be specified:

In order to successfully and deterministically process an exchange of business information all governing scopes must be available in every exchange.

2362

2363

**3. Interest based:**

2364

Only the scope information that the parties agree to or the parties deem interesting should be exchanged.

2365

2366

2367

Information about a particular perspective may be specified in a Profile. (see optional parts below)

2368

2369

**C.2 Meta model**

2370

2371

The meta model adds simple yet dynamic scoping to the header construct: The model specifies a directed acyclic graph (DAG) of governing scopes and contexts that covers a large set of frequently occurring business cases.

2372

2373

2374

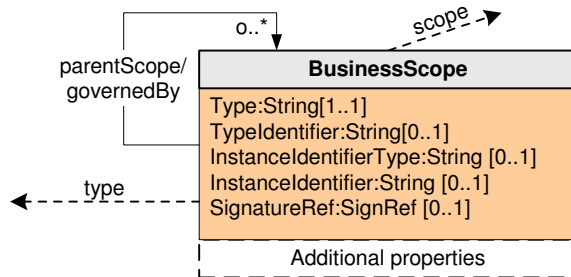
2375

This meta model of scope and context specification allows for great flexibility for business partners to use in ways we today cannot foresee. Yet it is predictable, composable and deterministic.

2376

2377

2378



2379

2380

2381

**BusinessScope** contains [1..1] [

2382

2383

**Scope** consists of [1..\*] [

2384

**ScopeType**:String [1..1] - type of scope:

2385

Examples are UN/CEFACT Transaction, BCF:BusinessCollaboration,

2386

BusinessProcess, ebXML:BusinessService, BusinessServiceAction,

2387

BCF:AuthorizedRole

2388

2389

**TypeIdentifier**: String[0..1] – optional unique identifier that references the

2390

type of governing scope (e.g. process model, document specification).

2391

Example; “bpss:dropship”

2392

2393

**InstanceIdentifierType**: String [0..1] – identifies the type of instance

2394

identifier. Examples: URL, GUID, ID, IdentifierString;

2395

2396

**InstanceIdentifier**: String [0..1] – unique identifier that references the

2397

instance of that scope (e.g. process execution instance, document instance)

2398

Example; “bpss:dropship:id-abcd123”

2399  
2400  
2401  
2402  
2403  
2404  
2405  
2406  
2407  
2408  
2409  
2410  
2411

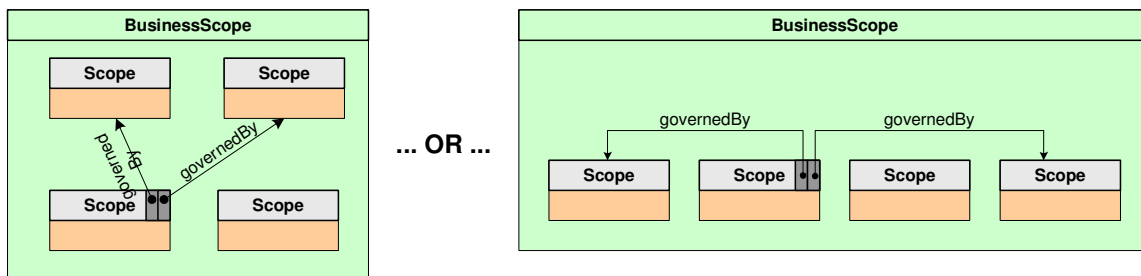
**ScopeSignatureReference: SignRef** [0..1] – a optional signature reference to the (governing) scope. In order to provide additional security a signature reference that point to governing scope may be defined.

**GovernedBy** contains [0..\*] [

**ParentInstanceIdentifier: String** [1..1] - optional Most of the time scopes forms a natural governance hierarchy and often a message exchange is governed by more than one parent agreement. This element references another Scopes InstanceIdentifier.

]

]



2412

### 2413 C.3 Wellformedness rules

2414

2415 [1] It is not mandatory to put all intermediary scopes in a generic header. Only  
2416 those that the parties agree to are needed. The following examples are all  
2417 relevant: [transaction], [transaction, business process], [business process],  
2418 [transaction, collaboration, collaboration, business process].

2419

2420 [2] A Profile may be used to group wf-rules together.

2421

2422 [3] The generic meta model specifies that cycles must not be present, i.e. by  
2423 following the GovernedBy relationship one must not return to the same scope.

### 2424 C.4 Optional parts

2425

2426 [1] An addition to above meta model: It is to possible add extra properties that  
2427 contain additional information about the scope and context. This information is  
2428 most likely to be redundant but may be used to control and verify state  
2429 synchronization. If the Scope is modeled using UML or similar modeling  
2430 language then additional properties may be captured in subclasses to Scope.

2431

2432 [2] It possible to add a Profile concept to Business Scope wellformdness rules so  
2433 that various combinations of mandatory ScopeType requirements may be

2434 grouped together. A profile is an expression of a particular perspective of  
2435 Business Scope.  
2436  
2437 [3] It is possibility to add an extra property to the Governance element which  
2438 specifies that the parent and child lifecycles are related and that when a parent  
2439 ends its lifecycle the child also end its lifecycle.  
2440  
2441 [4] It is possible to add an information element in the GovernedBy element in  
2442 order to indicate governance details. An example is an element that defines  
2443 superiority rules regulating overlapping rules in child scope versus parent  
2444 governing scope.  
2445  
2446 [5] The generic meta model specifies that cycles must not be present, i.e. by  
2447 following the GovernedBy relationship one must not return to the same scope.  
2448 This restriction may be relaxed by adding above Superiority rule and allowing  
2449 cycles.

## 2450 **C.5 NOTES**

2451  
2452 [1] The parent child relationship between scopes is not the same as a lifecycle  
2453 relationship. When a parent scope ends the child scope may still be active.  
2454 However in many use cases the scope relationship is linked to lifecycles but in  
2455 this generic meta model this dependency is implicit.  
2456  
2457 [2] Several methods may be use to identify scopes: Global identifiers (GUID, ... )  
2458 , relative identifiers (role name sequence number, local name, ..)  
2459  
2460 [3] In many type of specifications, business rules in a parent scope determine  
2461 processing rules of child scopes. Dynamic composition of specification and the  
2462 usage of business context such as in Core Component make it difficult to extract  
2463 information from one source, one specification document in order to determine  
2464 the final set of processing rules.  
2465  
2466 [4] In the future TPA, Contracts and technical agreements should be added as  
2467 governing scopes when defined within UBAC project.  
2468  
2469 [5] It is also possible have a Role-Party as a scope type. Could be used to  
2470 indicate role reversal.  
2471  
2472 [6] Business processes are important to organization but most business systems  
2473 don't keep track of them explicitly.  
2474  
2475 [7] Processing nodes between the sender and receiver may add and remove  
2476 scopes at the lowest lever without disturbing higher level governing parent  
2477 scopes. An example is a communication service that adds transport specific

2478 scopes before forwarding messages to lower lever transports and removes it  
2479 when forwarding messages to upper lever business data receiver application.



## Appendix D Relationship Between the SBDH and Other Standards

Cross-Section of Areas of Potential Interest between SBDH and other UN/CEFACT and ebXML Standards

	<b>Boundaries</b>	<b>Integration Points</b>	<b>Dependencies</b>
<b>AS2</b>	<b>Near-Term</b> SBDH supplements AS2 technology	SBDH integrates only at the Communication Software Application level	AS2 utilizes the SBDH for routing
	<b>Long-Term</b> SBDH will continue to supplement AS2 technology	SBDH will continue to integrate at the Communication Software Application level	AS2 will continue to utilize the SBDH for routing
<b>ATG NDR</b>	<b>Near-Term</b> SBDH follows its own syntax Naming and Design Rules	--	--
	<b>Long-Term</b> SBDH syntax will be subsumed by ATG Naming and Design Rules	--	SBDH is dependent upon the ATG Naming and Design Rules for interoperability
<b>BPSS</b>	<b>Near-Term</b> SBDH will supplement BPSS technology	SBDH integrates at the Parser/Translator or Middleware level	BPSS is not dependent upon some generic header technology but may optionally use it
	<b>Long-Term</b> SBDH will supplement or be subsumed by BPSS technology	SBDH will continue to integrate at the Parser/Translator or Middleware level	BPSS is not dependent upon some generic header technology but may optionally use it
<b>ebMS</b>	<b>Near-Term</b> SBDH may supplement ebMS technology	SBDH integrates at the Communication Software Application level	ebMS is not dependent upon some generic header technology but may optionally use it
	<b>Long-Term</b> SBDH may supplement ebMS technology	SBDH will integrate at the Communication Software Application level	

<b>EDI</b>	<b>Near-Term</b>	SBDH will supplement EDI technology	SBDH will integrate at the Parser/Translator or Middleware Application level	EDI is not dependent upon some generic header technology but may optionally use it, especially for Service Information and Correlation Information
	<b>Long-Term</b>	SBDH will supplement EDI technology	SBDH will continue to integrate at the Parser/Translator or Middleware Application level	EDI is not dependent upon some generic header technology but may optionally use it, especially for Service Information and Correlation Information
<b>UBAC</b>	<b>Near-Term</b>	SBDH will supplement or be subsumed by UBAC specifications	SBDH will integrate at the Business Transaction View and Business Service View levels	UBAC is dependent upon some generic header technology such as the SBDH
	<b>Long-Term</b>	To be determined	To be determined	If another technology becomes available, UBAC could use the new technology or the SBDH
<b>UMM</b>	<b>Near-Term</b>	SBDH will supplement UMM	SBDH will integrate at the Business Transaction View and Business Service View levels	SBDH is dependent upon the UMM meta-model
	<b>Long-Term</b>	To be determined	To be determined	SBDH continues to be is dependent upon the UMM meta-model