Proposed Revisions to ebXML Technical Architecture Specification v1.0.4

ebXML Business Process Project Team

11 May 2001

Status of this Document

There are three categories of ebXML deliverables:
- o Technical Specifications conform to the ebXML Requirements document.
- o Technical Reports are either guidelines or catalogues.
- o White Papers constitute a snapshot of on-going work within a Project Team.

This White Paper represents a report that has been approved by the Business Process Project Team and has been accepted by the ebXML Steering Committee.

The material in this document constitutes a snapshot of on-going work within this Project Team.

Distribution of this document is unlimited.

This version:
    http://www.ebxml.org/specs/bpTAREV.pdf

34   6. Business Process and Information Analysis Methodology and Meta-model
35

36   Analysis teams will use methodologies and metamodels to specify the business processes in an
37   electronic business community. An analysis methodology prescribes the overall process and sub-
38   processes by which teams should proceed when defining business processes. The semantics of
39   the metamodel define the information that needs to be discovered and documented during the
40   analysis process. Methodologies often include patterns to expedite the "design" of the model and
41   help achieve common expression of similar concepts.
42

43   While business practices from one organization to another are highly variable, these practices can
44   be decomposed into *Business Processes, Business Collaborations, Business Transactions* and
45   their related *Business Information* (documents). This analysis through the modeling process will
46   identify *Business Process* and *Information Models* that are likely candidates for re-use and
47   standardization. The ebXML approach looks for standard reusable components at all levels in
48   business process and information models from which to further construct new models. This
49   approach facilitates interoperability through its reuse of well-understood models and sub-models.
50

51   ebXML recommends (but does not require) that analysis teams use the methodology specified by
52   the UN/CEFACT Modeling Methodology (UMM). If an alternative methodology is used, it is
53   highly recommended that it be compliant with the UMM so as to have the best opportunity of
54   creating business process models that are compatible with business process models created using
55   the UMM. The ebXML *Business Process and Business Information Analysis Overview*
56   document describes the process by which enterprises can analyze, identify, and define those
57   business processes and business documents necessary for the conduct of electronic business with
58   other enterprises, within the ebXML framework.
59

60   Compliance to ebXML Requirements requires that the business process and business information
61   artifacts generated as a result of the analysis effort be conformant to the semantics defined a
62   single consistent modeling language and methodology. The ebXML Business Process modeling
63   language and methodology of choice is the UML-based UMM and its supporting business
64   process metamodel (UMM Metamodel). This is necessary to give the best assurance of
65   compatibility between business process models and model sub-components. This semantic
66   conformance is necessary to meet the requirement that the models to be usable and re-usable, and
67   be capable of being compared and contrasted. With models that are UMM Metamodel
68   conformant, users and tools can generate runtime XML business process specification instances
69   (e.g. in ebXML Business Process Specification Schema format) and other alternative
70   representations that have the same semantics. Furthermore, the models can be freely shared
71   among ebXML-compliant modeling tools, including, but not limited, to UML tools.
72

73   6.1 UN/CEFACT Modeling Methodology Overview

74

75   The UMM, which primarily implements the Business Operational View of the Open-edi
76   Reference Model, ISO/IEC 14662, provides the prescription and precision required for predictive
77   results. The UMM is based on Business Modeling, Requirements, Analysis and Design
78   workflows needed to understand the business needs to produce business scenarios, business
79   objects and areas of business collaboration. The use and relationships of the methods, patterns
80   and model artifacts are defined within each workflow. For each workflow a method is applied to
81   a pattern using modelling elements with well-defined semantics. The deliverables of the UMM
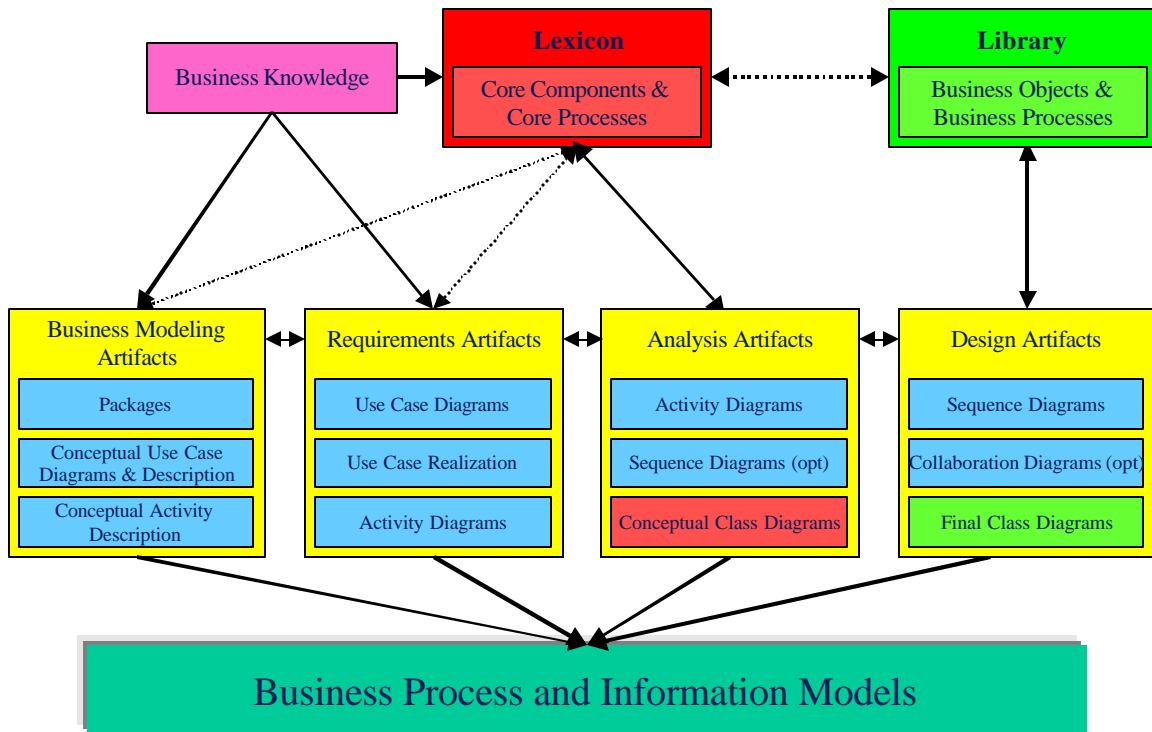82   workflows are shown as artifacts in Figure 2.

Figure 2 *UMM Workflows and Artifacts*

The UMM Metamodel is organized into the following views so that each process model can be viewed from a number of perspectives.

■ The Business Operations Map (BOM) metamodel – the partitioning of business processes into business areas and business categories.

■ The Business Requirements View (BRV) metamodel – the view of a business process model that captures the Use Case scenarios, inputs, outputs, constraints and system boundaries for commercial transactions and their interrelationships.

■ The Business Transaction View (BTV) metamodel - the view of a business process model that captures the semantics of business information entities and their flow of exchange between roles as they perform business activities.

■ The Business Service View (BSV) metamodel - the view of a business process model that specifies the network component services and agents and their message (information) exchange as interactions necessary to execute and validate a business process.

These perspectives support an incremental model construction methodology and provide levels of specification granularity that are suitable for communicating the model to business practitioners, business application integrators and network application solution providers.

6.2 ebXML Business Operation View

[Editor Note: Delete this section. The title of this section is confusing as the contents of this section provide what we have described in the *Business Process and Business Information Analysis* document (which is referenced above).]

6.3 ebXML Functional Service View

[Editor Note:
- Move this section to be between the current titles 7 and 7.1 and change title of section 7 to read "ebXML architecture overview". Also change figure number of Figure 4 to be Figure 3, as well as decrement all subsequent figure numbers.
- Add the following as the final sentence of the section: "The ebXML architecture corresponds to the Functional Service View of the Open-edi Reference Model, ISO/IEC 14662."]

121 **Changes to TA section 8.2 (this is a proposal to update/correct section 8.2 by item by**
122 **item changes, it is not a rewrite – to see actual changes, use Word's tools-> track**
123 **changes option:**
124
125 *The changes made are governed by the following rationale:*
126
127 **1. Current TA document makes inaccurate use of the word Meta Model.**
128 **Solution: Everywhere in the document change "ebXML Business Process and Information**
129 **Metamodel" to one of the following dependent on context of each occurrence:**
130
131 • **UMM metamodel**
132 • **Business Process and Information Model**
133 • **ebXML Business Process Specification**
134 • **Business Process**
135
136 **This applies to both text and figures.**
137
138 **2. Current TA document makes inaccurate use of the word Business Process.**
139 **Solution:**
140 **Everywhere where 'Business Process' refers to the document specifying it, change it,**
141 **dependent on context, to explicitly say ebXML Business Process Specification.**
142
143 **3. BP Analysis and BP metamodel teams have come to a new viewpoint on the relationship**
144 **between UMM methodology/metamodel and ebXML. This new viewpoint needs to be**
145 **reflected consistently in TA, BPSS, and Analysis documents.**
146 **Solution: State that UMM is not Mandatory but Recommended**
147 **3.a. Using a modeling methodology is optional, and even if you chose to model, UMM is**
148 **still optional**
149 **3.b. The only part of the UMM metamodel that is currently mandatory for BP is the**
150 **semantic subset represented by the ebXML Business Process Specification Schema.**
151 **3.c. As UN/CEFACT finalizes and evolved the UMM, it is anticipated that other parts of**
152 **the UMM metamodel may also become mandatory.**
153
154 **4. Current TA diagram has left out BPSS of the architecture overview diagram.**
155 **Solution: Amend figure 4 to show BPSS explicitly. This could be three boxes BPIM-**
156 **>ModelConversion to XML-->BusinessProcessSpecification.**
157
158 **5. Current TA diagram is unclear on storage format of BPSS. CPP/CPA requires it to be**
159 **XML**
160 **Solution: State that ebXML Business Process Specifications SHALL be expressed in XML,**
161 **(not just be expressable in XML)**
162
163 **6. Current TA is confusing in the discussion of UMM vs. UML. BP teams feel that TA**
164 **should concentrate on describing relationship to UMM rather than to UML>**
165 **Solution: Remove (or move) the requirement for UML**
166
167 **7. Current TA is not clear in its reference to ebXML Business Process Specification Schema**
168 **Solution: Fully spell out ebXML Business Process Specification Schema**
169
170 **8. BPSS issue # 118 needs to be addressed consistently in TA as well**

171 **Solution: Use phrase "UMM metamodel supports a set of business process viewpoints "**
172 **(rather than requirement/analysis/design viewpoints)**
173
174 **9. BP issue 41 points out inaccuracy in use of word message.**
175 **Solution: Replace Message with Business Document**
176
177

178 ## 8.2 Business Process and Information Modeling
179

180 ### 8.2.1 Introduction
181 The *UMM Metamodel* is a mechanism that allows *Trading Partners* to capture the details for a
182 specific business scenario using a consistent modeling methodology. A *Business Process*
183 describes in detail how *Trading Partners* take on roles, relationships and responsibilities to
184 facilitate interaction with other *Trading Partners* in shared collaborations. The interaction
185 between roles takes place as a choreographed set of business transactions. Each business
186 transaction is expressed as an exchange of electronic *Business Documents*. *Business Documents*
187 MAY be composed from re-useable *Business Information Objects* (see "Relationships to Core
188 Components" under 8.2.3 "*Interface*s" below). At a lower level, *Business Processes* can be
189 composed of re-useable *Core Processes*, and *Business Information Objects* can be composed of
190 re-useable *Core Components*.
191
192 The *UMM Metamodel* supports a set of business process viewpoints that provide a set of
193 semantics (vocabulary) for each viewpoint and forms the basis of specification of the artifacts that
194 are recommended to facilitate *Business Process* and information integration and interoperability.
195
196 An additional view of the UMM *Metamodel*, the ebXML *Business Process Specification Schema*
197 , is also provided to support the direct specification of the set of elements required to configure a
198 runtime system in order to execute a set of ebXML business transactions. By drawing out
199 modeling elements from several of the other views, the ebXML *Business Process Specification*
200 *Schema* forms a semantic subset of the *UMM Metamodel.* The ebXML*Business Process*
201 *Specification Schema* is available in two stand-alone representations, a *UML* version, and an
202 XML version.
203
204 The only part of the UMM metamodel that is currently mandatory for use in specifying ebXML
205 compliant software is the semantic subset represented by the ebXML Business Process
206 Specification Schema. As UN/CEFACT finalizes and evolves the UMM, it is anticipated that
207 other parts of the UMM metamodel may also become mandatory.
208
209 The relationship between the *UMM Metamodel* and the ebXML *Business Process Specification*
210 *Schema* can be shown as follows:
211
212
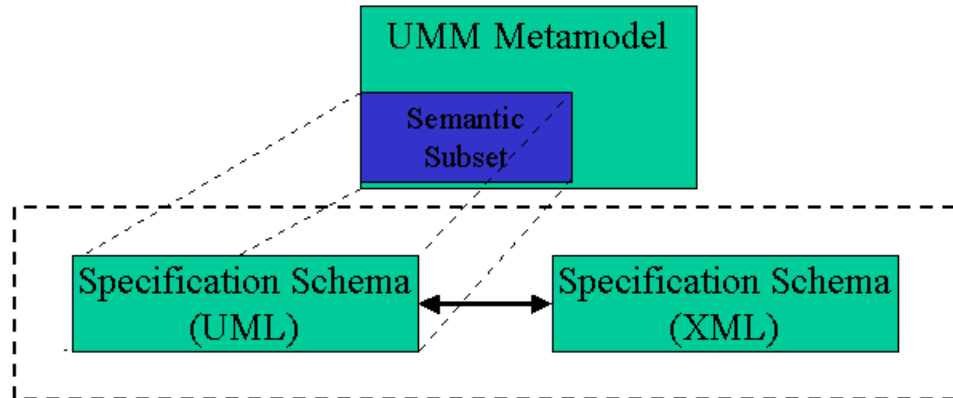
215 Using Figure 9 above as an illustration, instances of models and specifications would be created
216 as follows:

217   • A *Business Process and Information Model* is defined against the UMM Metamodel
218   • A *Business Process Specification* is defined against the *ebXML Business Process*
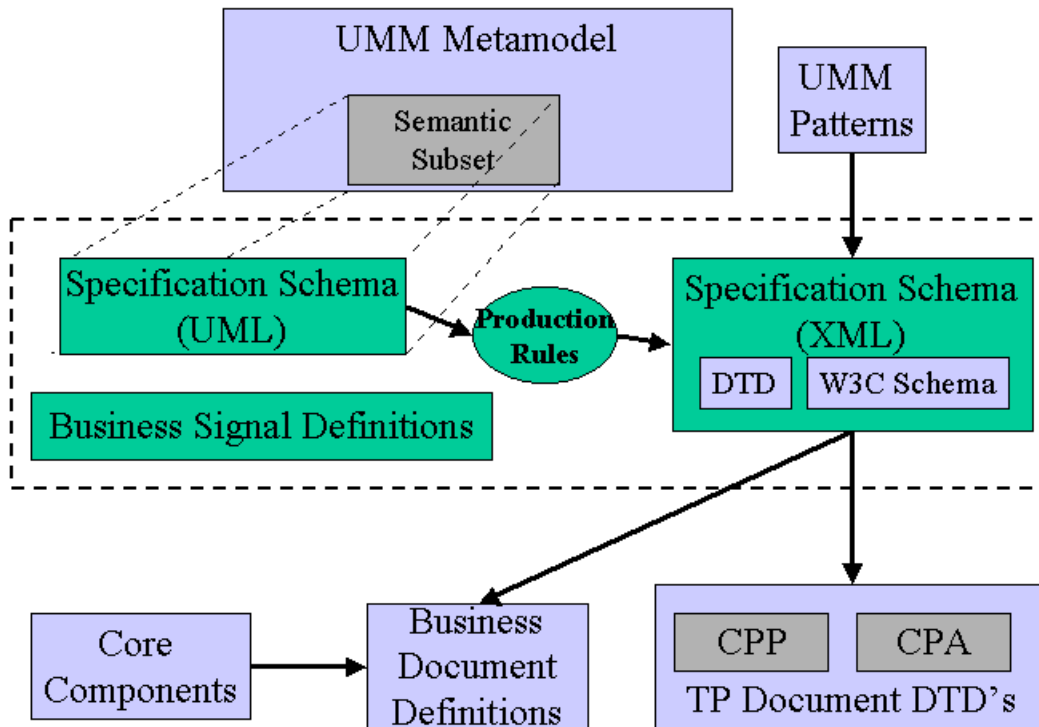219   *Specification Schema*

220

221 The ebXML *Business Process Specification Schema* supports the specification of business
222 transactions and the choreography of business transactions into *Business Collaborations*. Each
223 *Business Transaction* can be implemented using one of many available standard patterns. These
224 patterns determine the actual exchange of Business Document*s* and signals between *Trading*
225 *Partners* to achieve the required electronic transaction. To help specify the patterns the UMM
226 provides a set of standard patterns, and *the ebXML Business Process Specification Schema*
227 provides a set of modeling elements in support of those patterns. The ebXML specification of a
228 *Business Process* is referred to as a *Business Process Specification.*
229 The Business Process Specification serves as primary input for the formation of *Collaboration*
230 *Protocol Profiles* (*CPP's*) and *CPA's*.

231

232 This can be shown as follows:

233

Figure 10 *ebXML Business Process Specification Schema*

238 One of the key benefits of using a single consistent modeling methodology is that it is possible to
239 compare models to avoid duplication of existing *Business Processes*.

241 To further facilitate the creation of consistent *Business Process and information models*, ebXML
242 will define a common set of *Business Processes* in parallel with a *Core Library*. It is possible that
243 users of the ebXML infrastructure may wish to extend this set or use their own *Business*
244 *Processes.*

8.2.2 Formal Functionality

247 The representation of a *Business Process* Specification instance SHALL be in a form that will
248 allow both humans and applications to read the information. This is necessary to facilitate a
249 gradual transition to full automation of business interactions.

251 The *Business Process Specification* SHALL be storable and retrievable in a *Registry*
252 mechanism.*Business Process Specifications* MAY be registered in an ebXML *Registry* in order
253 to facilitate discovery and retrieval. To be understood by an application, a *Business Process*
254 *Specification* SHALL be expressed in *XML* syntax.

256 *Business Process Specifications* are capable of expressing the following types of information:

258   •   Choreography for the exchange of Business Document instances. (e.g. the choreography of
259   necessary *Business Document* exchanges between two Trading Partners executing a
260   "Purchasing" ebXML transaction.)
261   •   References to *Business Documents* (possibly *DTD's* or *Schemas*) that add structure to
262   business data.
263   •   Definition of the roles for each participant in a *Business Process.*
264
265 A *Business Process Specification*:
266   •   Provides the contextual constraints for using *Core Components*
267   •   Provides the framework for establishing *CPAs*
268   •   Specifies the domain owner of a *Business Process*, along with relevant contact information.
269   [NOTE: the above lists are not inclusive.]
270

8.2.3 Interfaces

Relationship to CPP and CPA

274 The *CPP* instance of a *Trading Partner* defines that partner's functional and technical capability
275 to support zero, one, or more roles in one or more *Business Process Specifications*.

277 The agreement between two *Trading Partners* defines the actual conditions under which the two
278 partners will conduct business transactions together. The *Interface* between a *Business Process*
279 and *Information Model,* and the *CPA* is the *Business Process* Specification. The the*Business*
280 *Process* Specification SHALL be instantiated as an *XML* document representing the business
281 transactional and collaboration layers of the *UMM Metamodel* according to the *ebXML Business*
282 *Process Specification Schema*. The expression of the sequence of commercial transactions in
283 *XML* is shared between the *Business Process Specification* and *Trading Partner CPP and CPA*
284 *documents.*

Relationship to Core Components

287 A *Business Process* Specification SHOULD specify the constraints for exchanging business data
288 with other *Trading Partners*. The business information MAY be comprised of components of the
289 ebXML *Core Library*. A *Business Process* Specification SHALL reference the appropriate
290 Business Documents (possibly DTD's or Schemas). The mechanism for interfacing with the *Core*
291 *Components* and *Core Library* SHALL be by way of a unique identifier for each component.

Relationship to ebXML Messaging

294 A *Business Process* Specification SHALL be capable of being transported from a *Registry*
295 *Service* to another *Registry Service* via an ebXML *Message.* It SHALL also be capable of being
296 transported between a *Registry* and a users application via the ebXML
297 *Messaging Service*.

Relationship to a Registry System

300 A *Business Process* Specification intended for use within the ebXML infrastructure SHALL be
301 retrievable through a *Registry* query, and therefore, each *Business Process Specification* SHALL
302 contain a unique identifier.

8.2.4 Non-Normative Implementation Details

305 The exact composition of *Business Information Objects* or a *Business Document* is guided by a
306 set of contexts derived from the *Business Process*. The modeling layer of the architecture is
307 highlighted in green in Figure 11 below.

308
309   ***Retain Current Figure 12 here, but renumber to Figure 11.***

310
311   *Business Process and Information Models* MAY be created following the recommended
312   UN/CEFACT *Modeling Methodology* (*UMM*), or MAY be arrived at in any other way. It is
313   recommended they comply with the *UMM Metamodel.*
314
315