

1  
2  
3  
4  
5  
6  
7  
8

**Second Draft**  
**Business Process Project Team**  
**Technical Specification Document**  
*Draft Version 2.0 6/23/00*

9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30

**Table of Contents**

➤ Document Introduction	2
➤ ebXML Business Process Metamodel	3
➤ Description of Metamodel Sub-groupings	5
➤ Illustrations of the Metamodel Sub-groupings	6
➤ Class Definitions	11
➤ Scenarios for Use of the ebXML Business Process Metamodel	16
➤ Auto Component Procurement Introduction and Example	22
➤ Section: Auto Example UML Use Cases	23
➤ Section: Auto Example UML Corresponding Collaborative Diagrams	28
➤ Section: Auto Supply Chain Procurement Practices Not Captured in Current Use Cases	35

31  
32 **The ebXML Business Process Metamodel Second Draft**  
33 **Business Process Project Team**  
34 **Technical Specification**  
35 ***Draft Version 2.0 6/23/00***  
36  
37

38 **The ebXML Business Process Metamodel**  
39

40 **Introduction**  
41

42 This document is a second draft technical specification for review by the ebXML Plenary.  
43 Comments are welcome. When registering your comment, please provide the following  
44 information:

- 45 ➤ Your name,
- 46 ➤ Your email address,
- 47 ➤ The document page and line number(s) associated with your comment,
- 48 ➤ Your comment,
- 49 ➤ Rationale for the comment, and,
- 50 ➤ Your recommended action for resolution of the issue or any recommended document  
51 add/change/delete modifications.

52  
53 Please e-mail comments to Marcia McLure, [marcia.mclure@mmiec.com](mailto:marcia.mclure@mmiec.com) within two weeks  
54 following the official posting date of June 23, 2000.  
55

56 This document includes the following sections:  
57

- 58  The ebXML Business Process Metamodel Class Diagram
- 59  Metamodel Sub-groupings
- 60  Descriptions of the Metamodel Sub-groupings
- 61  Metamodel Sub-grouping Class Diagrams
- 62  Class Definitions
- 63  Scenarios for the Use of the ebXML Business Process Metamodel
- 64  Automobile Component Procurement Example
- 65  Issues

66  
67  
68 Suggestions for document improvement are welcome. Thank you, in advance for your  
69 comments.  
70

71 ebXML Business Process Team

72  
73  
74  
75  
76  
77  
78  
79  
80  
81  
82  
83  
84  
85  
86  
87  
88  
89  
90  
91  
92  
93  
94  
95  
96  
97  
98  
99  
100  
101  
102  
103  
104  
105  
106  
107  
108  
109  
110  
111  
112  
113

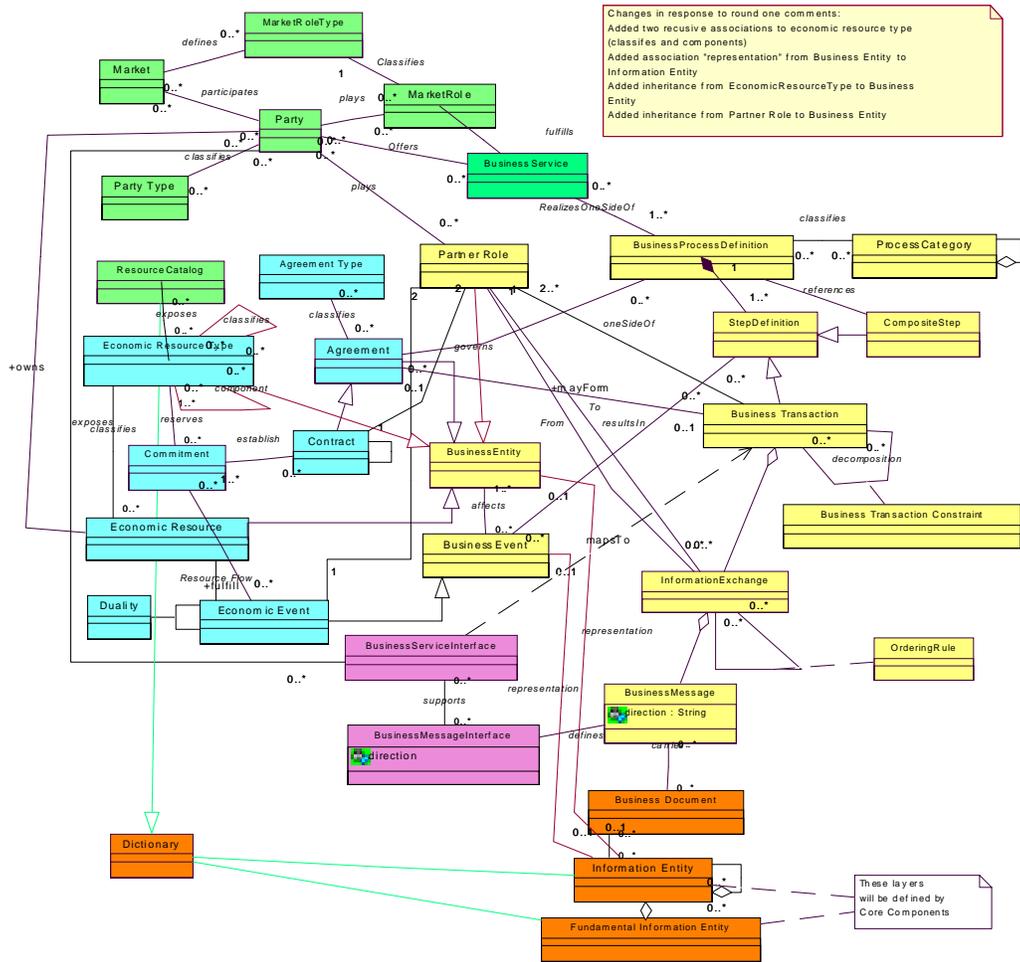
## **The ebXML Business Process Metamodel**

This is the ebXML business process metamodel that more fully defines the contract/commitment section. This ebXML business process metamodel also enables re-usability of process definitions. We refer to this state of the metamodel as Version 2.0.

The model consists of the following logical sub-groupings:

1. Resources and Contracts (color coded in blue),
2. Markets and Parties (color coded in green),
3. Business Processes and Rules (color coded in yellow),
4. Business Service Interfaces and Communication (color coded in purple), and the
5. Information Model (color coded in brown).

# The ebXML Business Process Metamodel



## Metamodel Sub-groupings

The metamodel consists of the following logical sub-groupings:

### 1. Resources and Contracts

This is a high level economic model, adapted from REA (Resources, Events, and Agents). It creates a very useful anchor point for the ebXML model, and establishes a pattern for how economic events should be transacted using this model.

### 2. Markets and Parties

This is the part of the model that allows organizations to register themselves relative to the markets they perform in and the types of services they offer. This aligns with the first four of the seven layers of the eCO framework. Once a number of organizations have registered themselves, other organizations can start discovering new business partners by navigating among the layers of the markets and parties sub-model.

### 3. Business Processes and Rules

This is the part of the model that describes the actual business processes that support the services offered by a given organization. It also describes the interactions required between the partners in order to obtain/perform the services offered.

### 4. Business Service Interfaces and Communication

This is the part of the model that describes the ‘interface’ that the partners expose, against which the ‘opposing’ partner can interact, typically by sending business signals consisting of business documents. Document is a broad term that covers both complete documents in the traditional sense, i.e. a sales order, but also descriptions of business events relevant to the service obtained/performed.

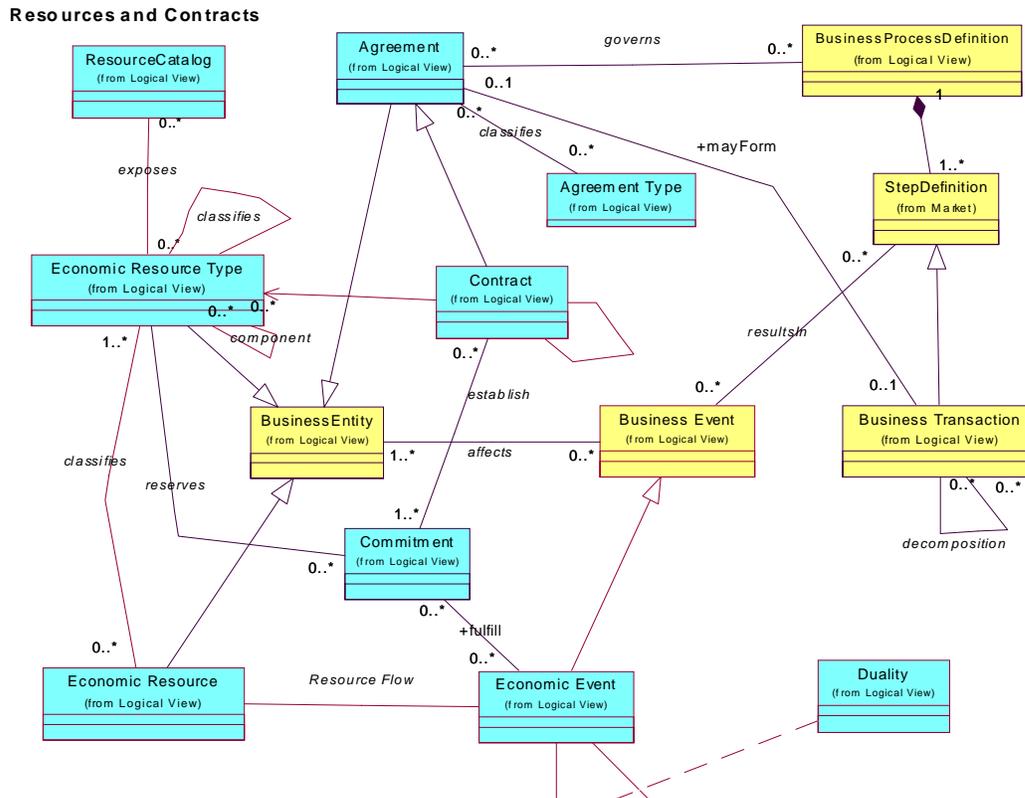
### 5. Information Model

151  
152  
153  
154  
155  
156  
157  
158

## Illustrations of the Metamodel Sub-groupings

The exact boundaries of each sub-grouping is subject to revision. The metamodel sub-groupings are as follows:

### 1. Resources and Contracts

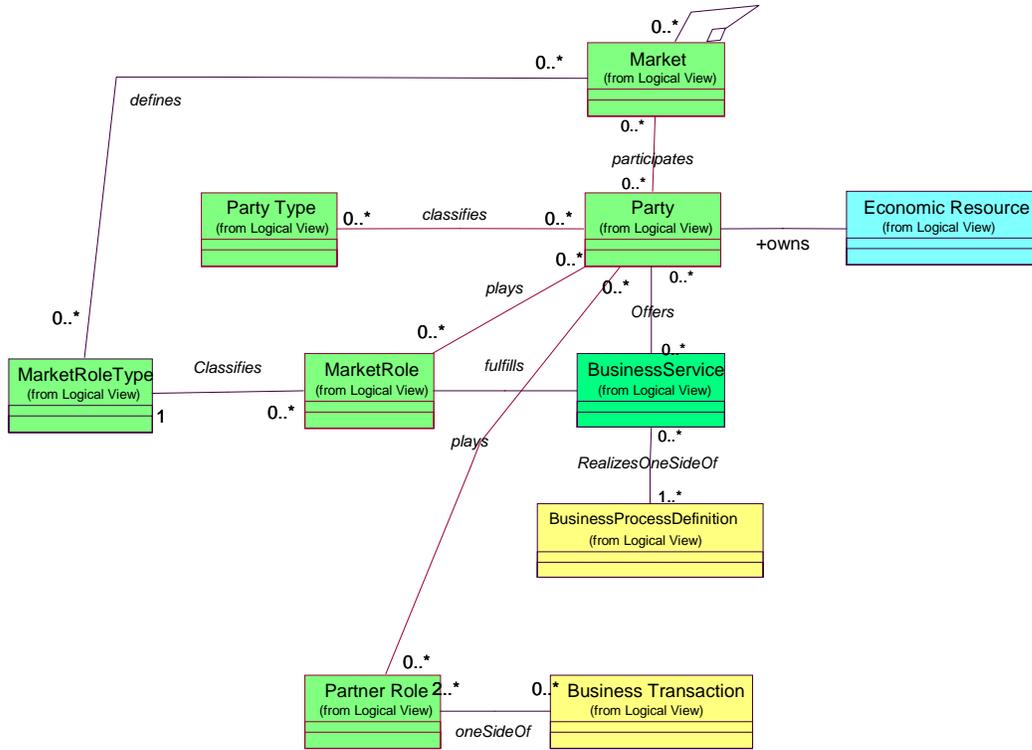


159

160  
161

## 2. Markets and Parties

Markets and Parties



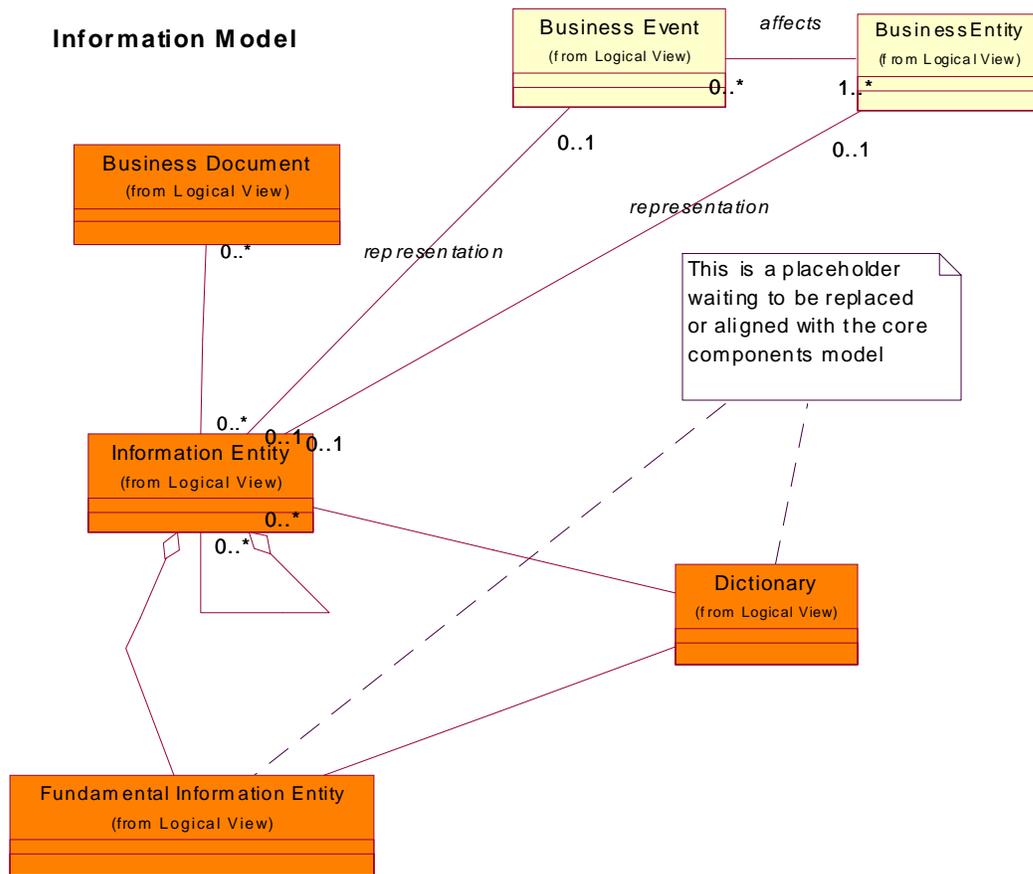
162  
163  
164  
165  
166  
167  
168  
169  
170  
171  
172  
173  
174  
175  
176  
177  
178  
179  
180  
181  
182  
183  
184  
185  
186





233  
234

## 5. Information Model



235  
236  
237  
238  
239  
240  
241  
242  
243  
244  
245  
246  
247  
248  
249  
250  
251  
252  
253  
254  
255  
256  
257

## Class Definitions

258  
259  
260  
261  
262  
263  
264  
265  
266  
267  
268  
269  
270  
271  
272  
273  
274  
275  
276  
277  
278  
279  
280  
281  
282  
283  
284  
285  
286  
287  
288  
289  
290  
291  
292  
293  
294  
295  
296  
297  
298  
299  
300  
301  
302  
303

Definitions of each of the classes are as follows:

### **Agreement**

An **agreement** is an arrangement between two parties that specifies in advance the conditions under which they will trade (terms of shipment, terms of payment, expectations of quotations and pricing, etc.) An agreement does not imply specific economic commitments.

### **Agreement Type**

An **agreement type** is the abstract classification of different types of agreements. Examples might include front-end agreements and yearly contracts.

### **Business Document.**

A **business document** is the description of a particular entity within a business, or the description of an agreement between organizations, or the description of a business event. The document is never the 'real' thing, just a description of it at a point in time. A business document is the central component of any information exchange among partner roles.

### **Business Entity**

**Business Entitys** are artifacts that are important in the execution of a company's business processes. Business Entities undergo changes that are reflected as Business Events.

### **Business Event**

A **business event** is a significant change in the state of one or more entities within a business, e.g. the taking of an order or the release of a shipment.

### **Business Message**

A **business message** is a message sent between two partners. A business message fulfills the requirements of an information exchange.

### **Business Message Interface**

A **business message interface** is the description of the protocol, both functional and technical, of business messages targeted to business services interfaces of a partner role engaged in an information exchange with another partner role.

### **Business Process Definition**

A business process is a collection of business transactions between business partners and/or internal activities within one business. These transactions and/or activities together support the objective of the business process. A **business process definition** specifies the choreography of business transactions needed to complete a business process. The internal activities that may also be needed, are outside the scope of the ebXML business process model.

### **Business Service**

**Business Services** are interfaces to a business process. Each Business Service offered by a Party provides the ability for a trading partner to interact with that Party in some way.

304  
305 A Party or "Service Provider" offers a Business Service to "Service Consumers". Any Party can  
306 be both a provider and a consumer of Business Services. Example Business Services offered by a  
307 company that makes widgets might be: "Examine my catalogue of widgets", "Buy a widget",  
308 "Submit engineering change order", "Become a VAR", "Find the cheapest price on this item and  
309 then apply for a loan to pay based on my credit rating and ability to establish a long term  
310 relationship", "Initiate manufacturing corrective action".

311  
312 **Business Service Interface.**

313 A **business service interface** is the definition of how to interact with one partner role in order to  
314 make him/her perform a desired service. For example, a partner role can expose a business  
315 process interface for 'quotation service'. It will describe precisely what kind of business  
316 messages you need to send, what you will get back, and what you may expect to have happen as  
317 a result of the exchange.

318  
319 **Business Transaction**

320 A **business transaction** is a logical unit of business conducted by two or more parties. The  
321 market, the partner roles, and the process, are all in a definable, and self-reliant state prior to the  
322 business transaction, and in a new definable, and self-reliant state after the business transaction.  
323 In other words if you are still 'waiting' for your business partner's response or reaction, the  
324 business transaction has not completed. A business transaction in our model is reflected as the  
325 required exchange or series of exchanges of information between two (or more) partner roles in  
326 order to complete the transaction. For example, the exchange could consist of a request for quote  
327 and the return either of the actual quote, or of the confirmation that the request had been  
328 received. It would not make sense to have the transaction (interaction) consist of the request  
329 only.

330  
331 **Business Transaction Constraint**

332 A **business transaction constraint** is a rule that guides and constrains the execution of business  
333 transactions within a business process.

334  
335 **Commitment**

336 A **commitment** is an obligation to perform an economic event at some future point in time.  
337 Commitments are fulfilled or executed by economic events. Order line items are examples of  
338 commitments.

339  
340 **Composite Step**

341 A composite step is a step composed of more than one logical step. This construct is used to  
342 provide greater flexibility in reusing step definitions.

343  
344 **Contract**

345 A **contract** is a mutual arrangement between parties that some actual economic exchanges will  
346 occur in the future. Contracts can have recursive relationships with other contracts, for example,  
347 yearly contracts with monthly releases and weekly or daily shipping schedules. Contracts are  
348 containers for collections of commitments. For example, a purchase order is a contract wherein  
349 the line items are commitments.

350

351 **Dictionary**

352 The **dictionary** should contain data types, re-usable components, and the templates (DTD's) of  
353 the business documents, but not the documents themselves.

354

355 **Duality.**

356 **Duality** is a relationship between Economic Events, where one is the legal or economic  
357 consideration of the other. Examples include a payment for a product or service.

358

359 **Economic Event**

360 An **economic event** is the transfer of control of an Economic Resource from one party to another  
361 party. Examples would include sale, cash-payment, shipment, and lease.

362

363 **Economic Resource**

364 An **economic resource** is a quantity of something of value that is under the control of an  
365 enterprise. Examples are cash, inventory, labor service and machine service.

366

367 **Economic Resource Type**

368 An **economic resource type** is the abstract classification or definition of an Economic Resource.  
369 For example, in an ERP system, ItemMaster or ProductMaster would represent the Economic  
370 Resource Type that abstractly defines an Inventory Item or Product. Economic Resource Types  
371 may have recursive relationships, so that for example broad classifications like "product" could  
372 group smaller classifications like "product family", which in turn could have as members the  
373 specific "product masters" with SKU numbers.

374

375 **Fundamental Information Entity.**

376 A **fundamental information entity** is in essence a data type. In business contexts we might need  
377 many more 'data types' with business semantics beyond the standard data types of 'int', 'float' etc.  
378 This class is a placeholder and will be further defined through discussions between the business  
379 process project team and the core components project team.

380

381 **Information Entity.**

382 An **information entity** is a primitive or complex data structure. We haven't defined this yet, but  
383 it may be that the difference between a data structure and an information entity is that the  
384 information entity also contains business rules about the data.

385 This class is a placeholder and will be further defined through discussions between the business  
386 process project team and the core components project team.

387

388 **Information Exchange.**

389 An **Information Exchange** is a set of business messages exchanged between two partners,  
390 related to a specific a specific business transaction.

391

392 **Market.**

393 A **market** is a 'meeting place' where organizations and individuals can exchange services or  
394 products. A market is defined in terms of the types of services and products that are likely to be  
395 exchanged. The "Yellow Pages" in a telephone book is an example of classifications of products  
396 and services, e.g. 'Legal Services', or 'Air condition products'. A person can then anticipate the  
397 existence of a 'Legal Services' market and an 'Air Conditioning' market.

398  
399  
400  
401  
402  
403  
404  
405  
406  
407  
408  
409  
410  
411  
412  
413  
414  
415  
416  
417  
418  
419  
420  
421  
422  
423  
424  
425  
426  
427  
428  
429  
430  
431  
432  
433  
434  
435  
436  
437  
438  
439  
440  
441  
442  
443  
444

**Market Role**

A **Market Role** defines a conceptual grouping of Business Services that a Party can provide in a Market.

**Market Role Type**

**Market Role Types** define and classify Market Roles.

**Ordering Rule.**

An **ordering rule** describes the interdependencies (i.e. ordering) of business messages within an information exchange.

**Partner Role**

A **partner role** is the role a party plays in a specific business process or business transaction.

**Party**

A **party** is any organization or individual that participates in exchanges of products or services in one or more markets. A party is established first as an absolute entity and then in terms of the roles it plays in a market and in terms of the role it plays in a business transaction.

**Party Type.**

A **party type** is a broad classification of the kind of organization or individual. Examples are 'University', 'Corporation', 'Individual', 'Government'.

**Process Category.**

A **process category** is a broad classification of business processes. At a macro level this classification could be like the "Yellow Pages" classification of services. At a finer level, processes could be classified to more functional groupings such as 'quotation', 'scheduling',  
The metamodel does not constrain the kinds of classification of processes.

**Resource Catalog**

A **resource catalog** is basically a navigable guide to offered products and services (Economic Resource Types). It is the market equivalence of a company's product catalog. It would be intended for narrowing down the particular kind of product or service you are looking for, hopefully leaving you with multiple possible sources for that product or service.

**Step Definition**

A **step definition** defines the steps in a business process. Step definition allows for the decomposition (and reuse) of steps and makes the interdependencies between steps explicit. Step interdependencies include predetermined step sequencing, and (implicit) business rules. A step definition always defines either an action taken by a single partner role or an interaction among partner roles.

445  
446  
447  
448  
449  
450  
451  
452  
453  
454  
455  
456  
457

458  
459  
460  
461  
462  
463  
464  
465  
466  
467  
468  
469  
470  
471  
472  
473  
474  
475  
476  
477  
478  
479  
480  
481  
482  
483  
484  
485  
486  
487  
488  
489  
490  
491  
492  
493  
494  
495  
496  
497  
498  
499  
500  
501  
502  
503  
504

## **Scenarios for Use of the ebXML Business Process Metamodel.**

The objective of ebXML is to “create a single global electronic market” that enables organizations to find each other and conduct business together through the exchange of information in the form of XML based business documents.

From this statement we can glean the following layers of importance to the Business Process Metamodel:

It must support the definition of a “market”, the definition of processes for “conducting business”, the definition of required “exchanges of information”, and the definition of the “business documents” themselves.

Therefore the following LAYERS of the business process must be supported by the metamodel:

- A. Market (for categorizing and organizing parties and their processes/services)
- B. Business Process (for conducting business)
- C. Information Exchange (in support of a business process)
- D. Business Document (for structuring information)

Note: There is an alignment of these layers to the packages of the metamodel.

The alignment is as follows:

- The market layer uses the ‘Markets and Parties’ package and the economic resource part of the “Resources and Contracts” package.
- The Business Process layer uses the “Business Process and Rules” package and the “Resources and Contracts” package.
- The “Information Exchange” layer uses the “Business Information Model ” and the “Business Service Interface” packages.
- The “Business Document” layer uses the business document and information entity part of the “Business Information Model” package.

We also divide the scenarios for usage of the metamodel into the following ‘STAGES’:

1. Designing/Describing markets, business processes, information exchanges and business documents.
2. Implementing system to execute in conformance with described business processes, information exchanges and business documents
3. Registering markets, business processes, information exchanges and business documents.
4. Discovering markets, business processes, information exchanges and business documents.
5. Actual execution of a business process through the exchange of business documents.

505  
 506 So we can organize the scenarios as follows:  
 507  
 508

	1. Design/Describe	2. Implementation	3. Register	4. Discover	5. Execution
A: Market	Market-Design	N/A	Market-Registration	Market-Discovery.	N/A
B: Business Process	Process-Design	Process-Implementation	Process-Registration	Process-Discovery	Process-Execution
C: Information Exchange	Exchange-Design	Exchange-Implementation	Exchange-Registration	Exchange-Discovery	Exchange-Execution
D: Business Document	Document-Design	Document-Implementation	Document-Registration	Document-Discovery	Document-Execution

509  
 510 In the following we describe, for each of the table entries above, how the user and/or a tool  
 511 provider will make use of the metamodel, and how each of the other pieces of the ebXML  
 512 architecture are related.

513  
 514 For ease of understanding, we divide this discussion into the following distinct types of  
 515 scenarios.

516 ‘From scratch design’ – An organization designing, implementing, registering a brand new  
 517 market and process.

518 ‘Conversion’ – An organization converting an existing market and process design, and adjusting  
 519 an existing implementation.

520 “Discovery and adaption” – An organization discovering an existing party and process and  
 521 adapting their existing implementation to interoperate.

522 “Actual communication” – Two organizations actually conducting business by exchanging  
 523 messages.

524  
 525 **Brand new business model.**

526  
 527 This scenario assumes for simplicity that none of the parts of the business model are yet in the  
 528 repository and that the organization(s) designing it are willing to retrofit their applications to fit  
 529 the new model.

530  
 531 The stages the organization would go through are:

- 532  
 533 1. Design: (For this stage the organization would either use established modeling tools and  
 534 convert the output to DTD/XML compliant with the ebXML metamodel, or they would  
 535 use newer lightweight ebXML front end tools to produce ebXML compliant DTD/XML  
 536 directly)
- 537 a) Market-Design: Determine and describe the market in terms of its domain and it’s  
 538 parties.
  - 539 b) Process-Design: Determine and describe the business process in terms of its  
 540 partner roles and business transactions
  - 541 c) Exchange-Design: Determine and describe each business transaction in terms of  
 542 its required messages exchanged.

- 543 d) Document-Design: Determine and describe each business document in terms of  
544 its attributes  
545
- 546 2. Implementation. (This may be accomplished using new lightweight adaptor tools to front-  
547 end their applications)
- 548 a) Market implementation is not relevant  
549 b) Process-Implementation: Design and implement a Business Service Interface that  
550 covers all the business transactions specified in 1.b. above.  
551 c) Exchange-Implementation: Design and implement Business Message Interfaces  
552 that cover all the Information Exchanges specified in 1.c. above.  
553 d) Document-Implementation: Design and implement mappings from the documents  
554 specified in 1.d. above.
- 555 When this is working they would register the market, party, partner-role, business  
556 process, information exchange and business documents and register themselves as  
557 capable of supporting this new model.  
558
- 559 3. Registration: Registration takes place by using a web-based front end to the ebXML  
560 repository and/or sending a model compliant xml file using the ebXML message  
561 exchange.
- 562 a) Market-Registration: Register each market and party specified in 1.a.  
563 b) Process-Registration: Register business process specified in 1.b. and its associated  
564 business transactions and business rules.  
565 c) Exchange-Registration: Register for each business transaction specified in 1.b. the  
566 required information exchanges as specified in 1.c.  
567 d) Document-Registration: Register each business document specified in 1.d. above.  
568

569 The process and site-implementation for this “brand new” business process is now ready for  
570 business, next step would be “discovery and adaptation” by potential business partners (see  
571 below)  
572

## 573 **Conversion**

574

575 This scenario assumes for simplicity that the company already has a complete model design  
576 described in some other format and protocol.  
577

578 The stages the organization would go through are:  
579

- 580 1. Design. (or in this case convert the existing explicit or implicit design)
- 581 a) Market-Design: Extract and convert from existing model the market in terms of  
582 its domain and it’s parties. This conversion should yield an ebXML metamodel  
583 compliant XML based model ready for registration.  
584 b) Process-Design: Extract and convert from existing model the business process in  
585 terms of its partner roles and business transactions. This conversion should yield  
586 an ebXML metamodel compliant XML based model ready for registration.

- 587 c) Exchange-Design: Extract and convert from existing model each business  
588 transaction in terms of its required messages exchanged. This conversion should  
589 yield an ebXML metamodel compliant XML based model ready for registration.  
590 d) Document-Design: Extract and convert from existing model each business  
591 document in terms of its attributes. Since many “libraries” of standard based  
592 document designs already exist, and since the metamodel here is very flexible, it  
593 is anticipated that little or no conversion be needed for standards based  
594 documents. Rather there would just be a qualification attribute of the exchange-  
595 design in 1.c. above as to which of several standards the documents involved  
596 belong to.  
597
- 598 2. Implementation. (This may be an activity of creating wrappers around the existing system  
599 to enable the sending and receiving of messages).  
600 a) Market implementation is not relevant  
601 b) Process-Implementation: Design and implement a Business Service Interface that  
602 covers all the business transactions specified in 1.b. above.  
603 c) Exchange-Implementation: Design and implement Business Message Interfaces  
604 that cover all the Information Exchanges specified in 1.c. above.  
605 d) Document-Implementation: Design and implement mappings from the documents  
606 specified in 1.d. above.  
607
- 608 3. Registration: Registration takes place by using a web-based front end to the ebXML  
609 repository and/or sending a model compliant xml file using the ebXML message  
610 exchange.  
611 a) Market-Registration: Register each market and party specified in 1.a.  
612 b) Process-Registration: Register business process specified in 1.b. and its associated  
613 business transactions and business rules.  
614 c) Exchange-Registration: Register for each business transaction specified in 1.b. the  
615 required information exchanges as specified in 1.c.  
616 d) Document-Registration: Register each business document specified in 1.d. above.  
617 Since your document may already be specified in another industry standard  
618 protocol, you may register just a hyper-link to where the specification is found in  
619 an ebXML compliant format.  
620

621 The process and site-implementation for this “converted” business process is now ready for  
622 business, next step would be “discovery and adaptation” by potential business partners (see  
623 below)  
624

## 625 **Discovery and adaption**

626

627 This scenario assumes for simplicity that an organization can find a partner with an appropriate  
628 process and only needs to make adjustments to its applications in order to ‘play’. In this scenario  
629 the discovery comes first (so we have changed the sequence, but left the numberings intact as a  
630 reference back to the matrix). Once discovery has yielded an acceptable, process, information  
631 exchange, and document structure, the organization has only to adapt its applications.  
632

633 The stages the organization would go through are:

634

635 4. Discovery: (This is done using web front ends to the ebXML repository, or by sending  
636 XML 'query' documents through the ebXML message facility).

637 a) Market-Discovery: Using appropriate keywords and wildcards find the market of  
638 interest. Starting from the market find possible parties who may be possible  
639 partners.

640 b) Process-Discovery: Starting from each possible party discover his/her role in  
641 various processes. Find a process that matches the business transactions you need  
642 to transact.

643 c) Exchange-Discovery: Starting from each business transaction discover if you are  
644 capable of producing and consuming the required information exchanges in the  
645 specified protocols.

646 d) Document-Discovery: Starting from each information exchange, discover if you  
647 are capable of mapping into and out of the specified business documents.

648

649 1. Design. Not applicable, in essence this organization is using a design already done by  
650 another organization.

651 a. Market design was discovered in 4.a. above

652 b. Process design was discovered in 4.b. above

653 c. Exchange design was discovered in 4.c. above

654 d. Document design was discovered in 4.d. above

655

656 2. Implementation. (This may be an activity of creating wrappers around the existing system  
657 to enable the sending and receiving of messages).

658 a. Market implementation is not relevant

659 b. Process-Implementation: Design and implement a Business Process Service that  
660 covers all the business transactions specified in 1.b. above.

661 c. Exchange-Implementation: Design and implement Business Message Interfaces  
662 that cover all the Information Exchanges specified in 1.c. above.

663 d. Document-Implementation: Design and implement mappings from the documents  
664 specified in 1.d. above.

665

666 3. Registration: Not required unless you want to establish a more formal 'trading partner  
667 agreement'

668 a. Market registration already done

669 b. Process registration already done

670 c. Exchange design may involve the registration of your business process interface  
671 to handle your end of the process. This may be validated against the business  
672 processes already registered for handling the other end.

673 d. Document registration may involve the registration of your document handler  
674 interfaces to handle the incoming and outgoing messages. At this point it may be  
675 possible to send a series of "test messages" that traverses the whole process and  
676 proves that the two parties can in fact live up to the implicit or explicit 'trading  
677 partner agreement'.

678 Note: The described kind of registration of business process interfaces and document  
679 handler interfaces may not initially be part of ebXML scope, rather – initially - an eCO  
680 style self-registration on your own site might be workable.  
681

682 This business partner is now ready to do business with the partner/process previously registered.  
683

### 684 **Actual communication**

685  
686 This scenario assumes that we have already designed, registered and implemented as per above.  
687

- 688 1. Design: Already done above
- 689 2. Implementation: Already done above
- 690 3. Discovery: Already done above
- 691 4. Registration: Already done above
- 692 5. Execution: The model drives the execution in the sense that the business transaction  
693 sequence within a process is (optionally) specified, and the message exchange sequence  
694 within a business transaction is (optionally) specified. So one could envision an  
695 implementation that actually accesses the ebXML repository to figure out what needs to  
696 happen next. More likely the parties implement their ebXML process compliant business  
697 process interfaces, and the exchanges happen directly between these business process  
698 interfaces, using message formats prescribed in the repository. These business process  
699 interfaces may themselves handle the mapping into or out of the organizations  
700 applications, or may interact with “wrappers” specifically designed for this purpose. In  
701 either case, the ebXML end of the mapping is prescribed by the registered documents.  
702  
703  
704  
705  
706

707  
708  
709  
710  
711  
712  
713  
714  
715  
716  
717  
718  
719  
720  
721  
722  
723  
724  
725  
726  
727  
728  
729  
730  
731  
732  
733  
734  
735

## Automobile Component Procurement Example

### Introduction

This is the first ebXML-BP metamodel example. More will come, including some that are much simpler than this one, which is deliberately complex in order to test the metamodel.

This example is not "final". The intention is for this example to develop along with the ebXML project until it is fully populated with functional test data, and also to be accompanied by several other examples illustrating different scenarios.

The reasons for starting with this particular process include:

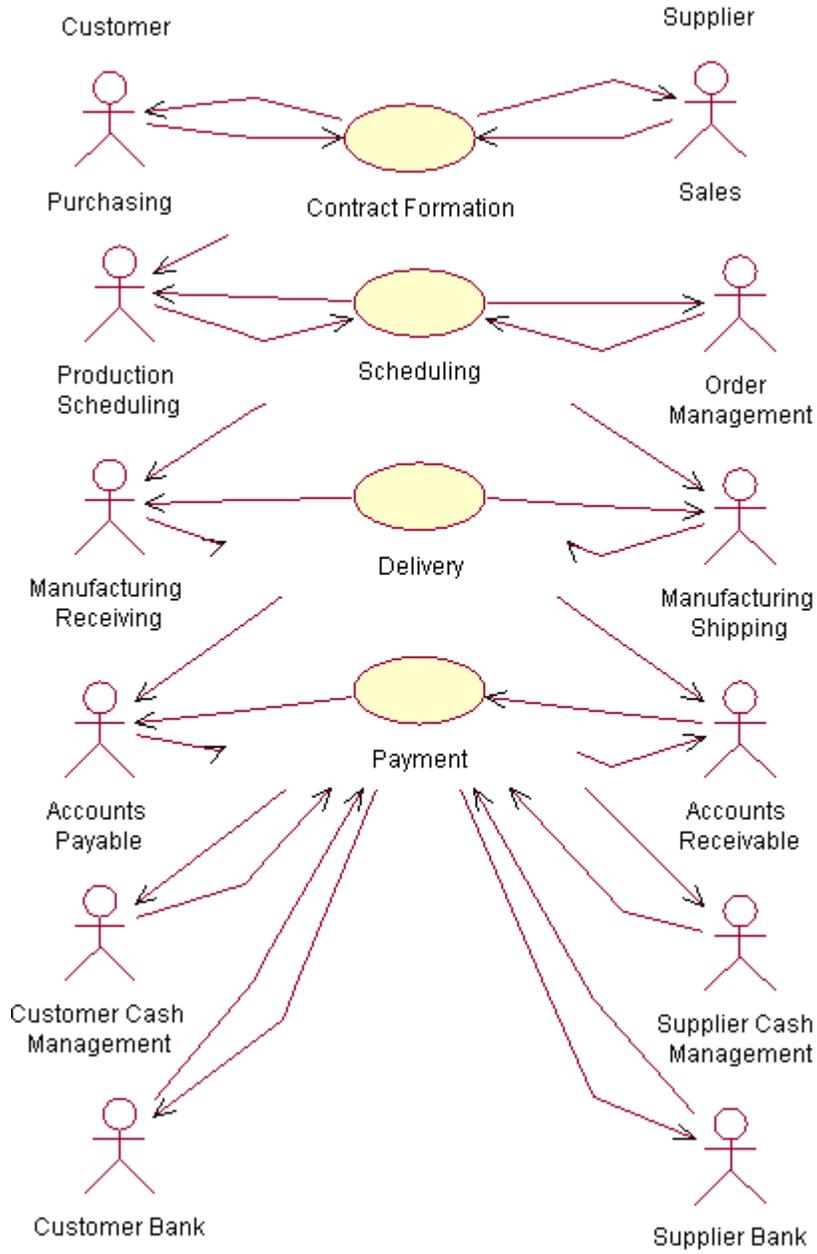
- it is a supply chain direct-component procurement example, instead of the usual office supply purchase;
- the business practices cover most of the metamodel;
- the business practices are well documented by an industry-wide group, AIAG (Automotive Industry Action Group);
- the business practices are similar to supply chain relationships in other industries, e.g. appliances and retail.

### Example Sections

1. **UML Use Cases**, with no reference to ebXML metamodel classes or technology.
2. **UML Collaboration Diagrams** mapping the use cases to the current ebXML metamodel classes. (Note: not every detail of the use cases is shown in collaboration diagrams. Some sections were omitted as being repetitive, with no new mappings.)
3. **Uncaptured auto supply chain procurement practices** - not yet included in the current use cases.

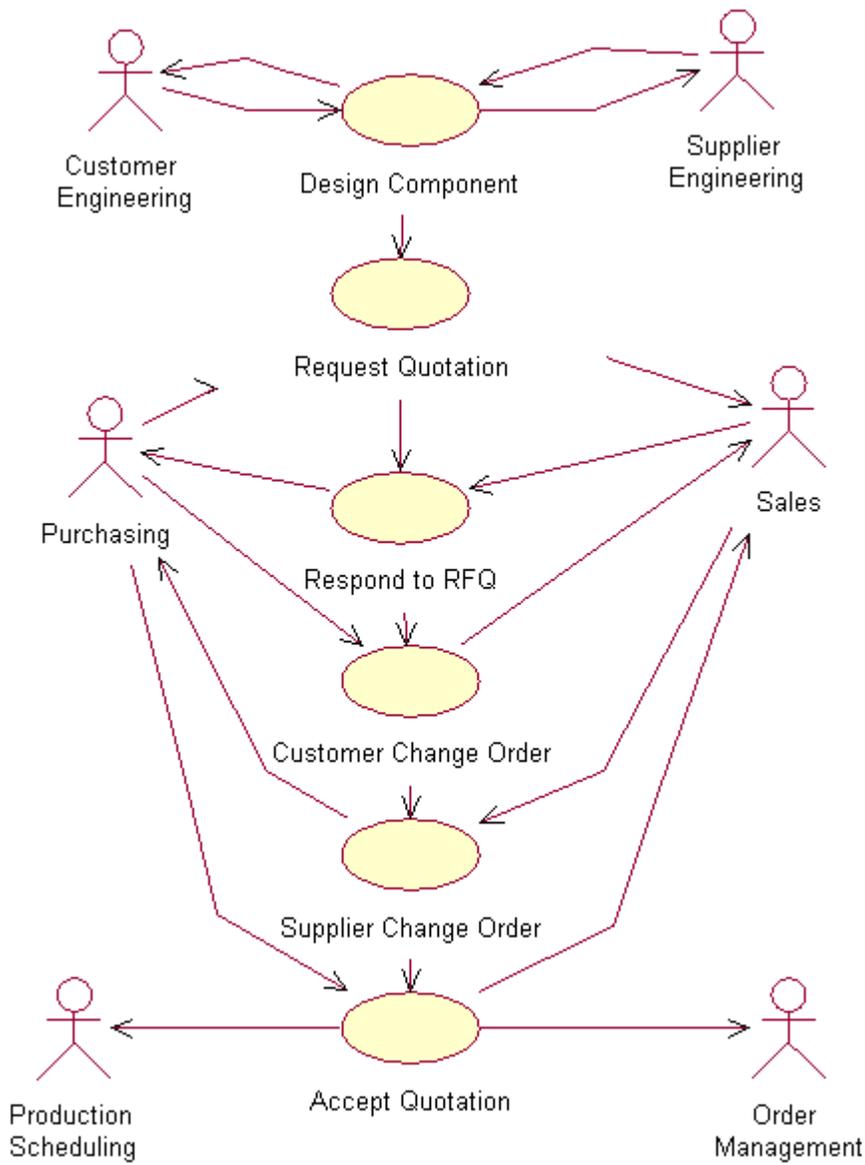
736  
737  
738

## Use Cases - Overview



739  
740

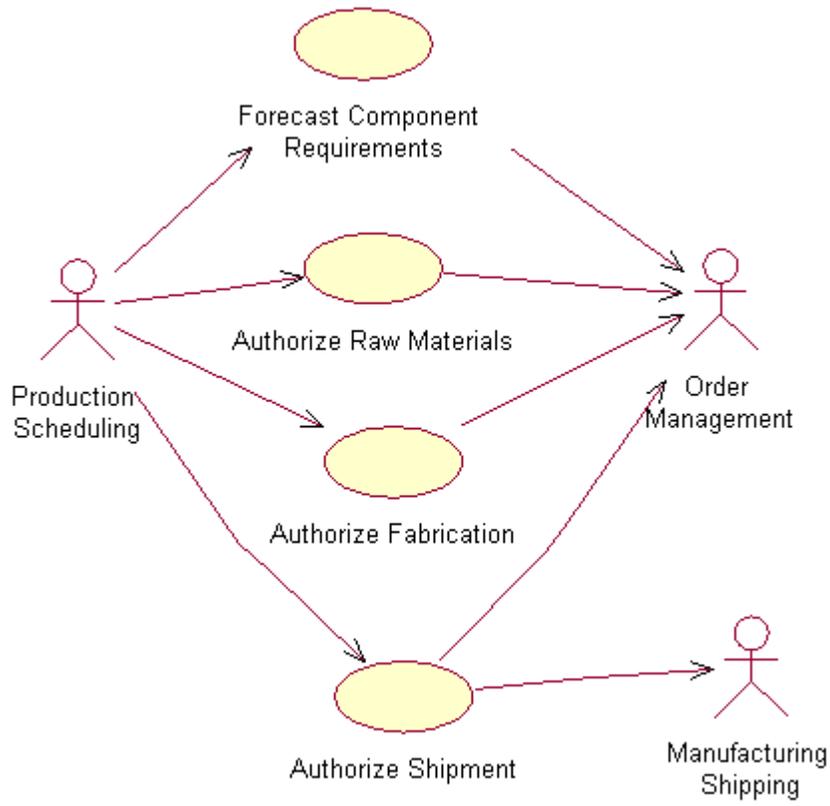
### Contract Formation



742  
743  
744  
745

746  
747

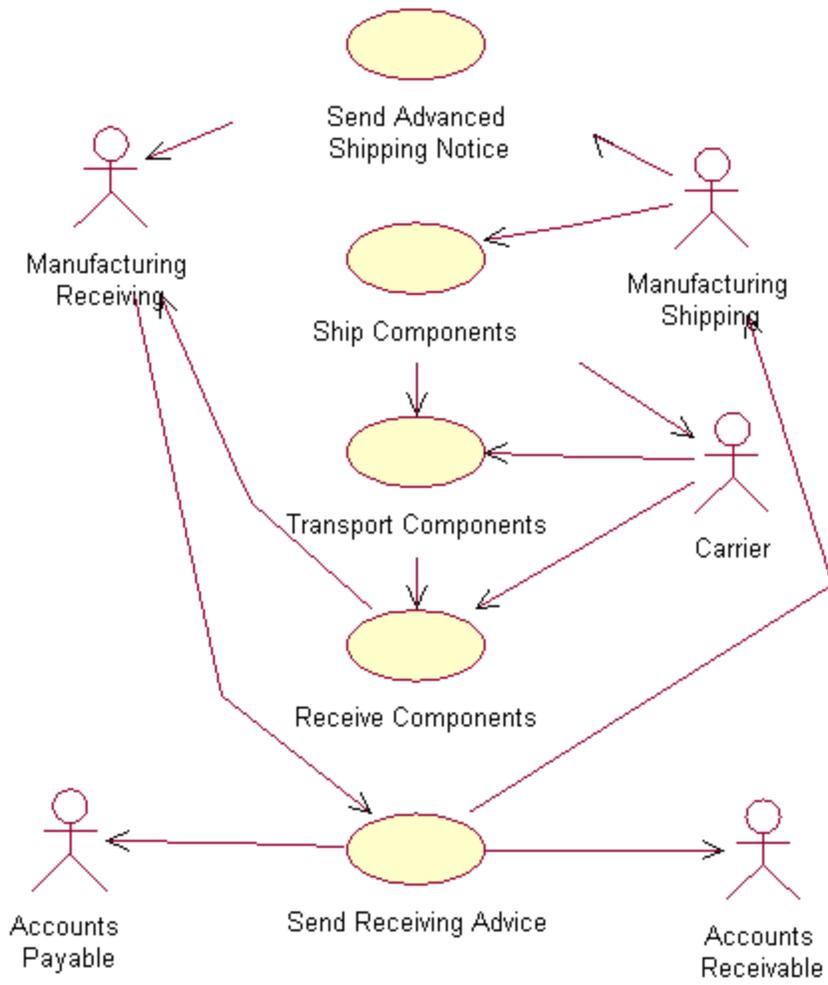
## Scheduling



748  
749

750  
751

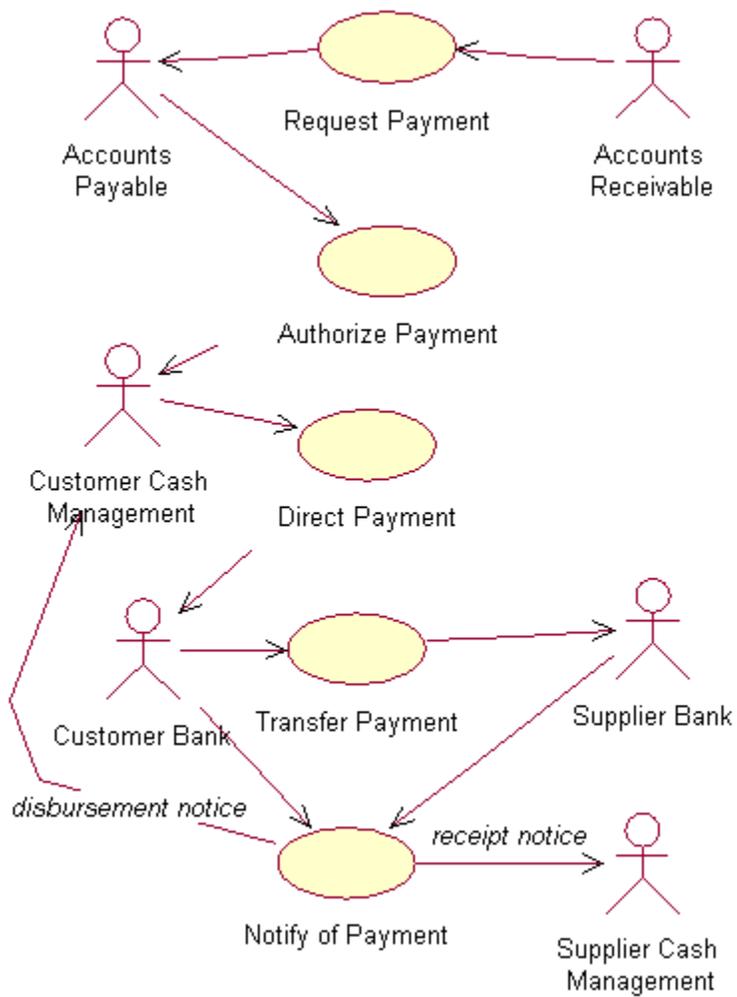
# Delivery



752  
753

754  
755

## Payment



756  
757

758 **Corresponding Collaboration Diagrams**

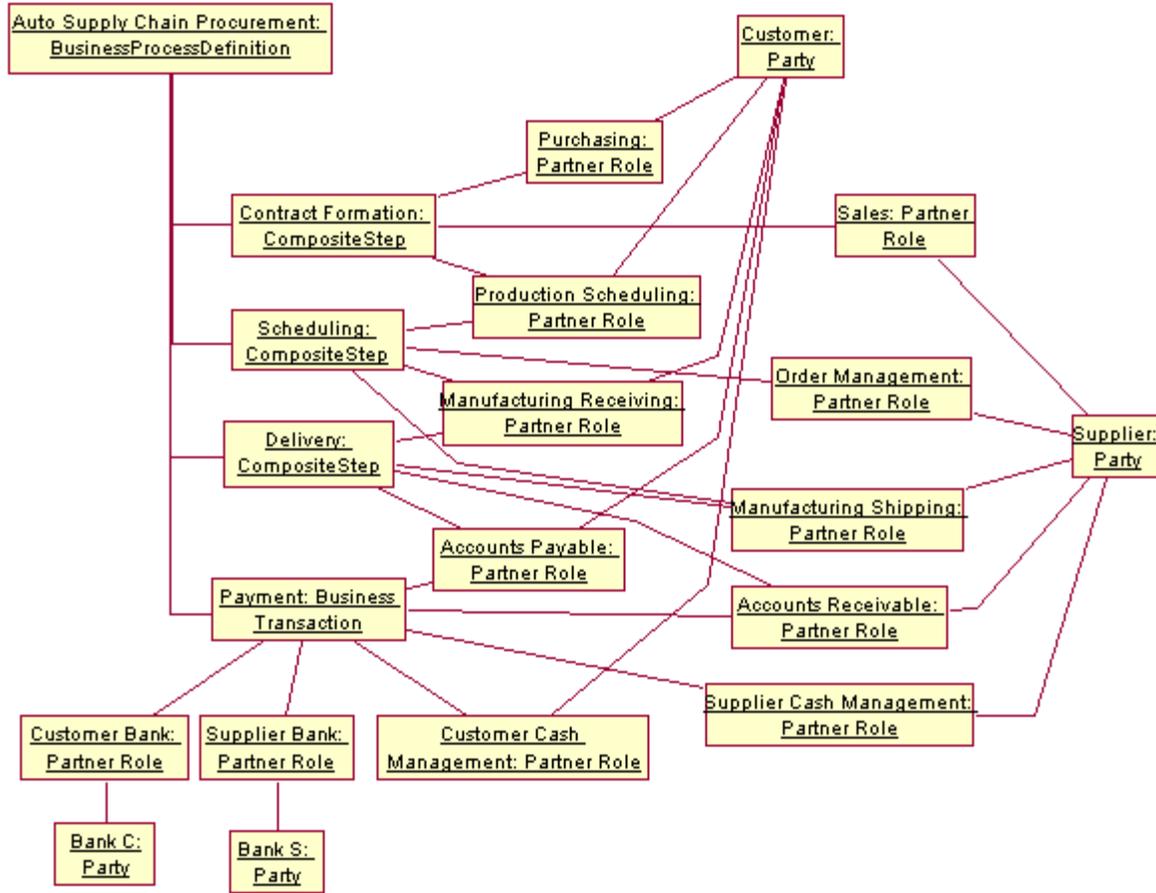
759

760 In each rectangle, an object name is followed by an ebXML metamodel class name, e.g. Object: Class.

761

762 **Overview**

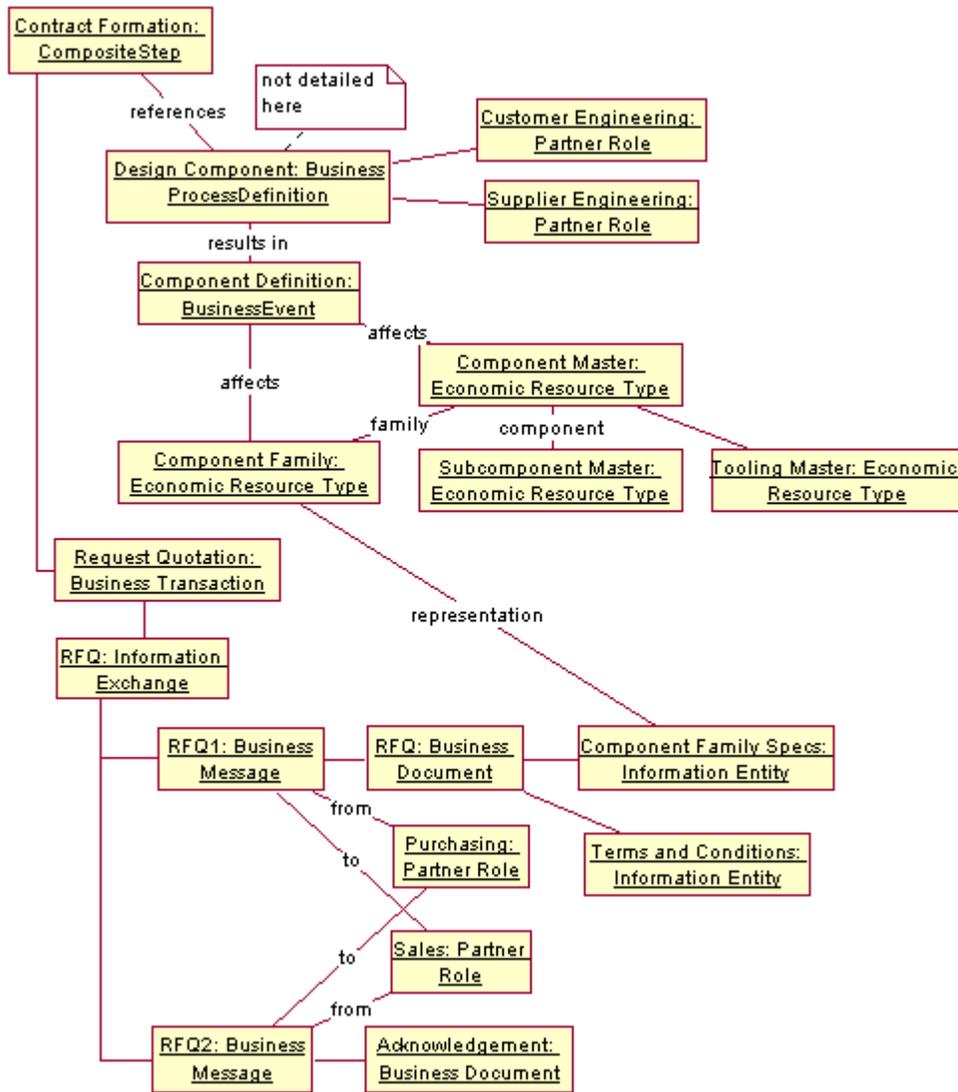
763



764

765  
766

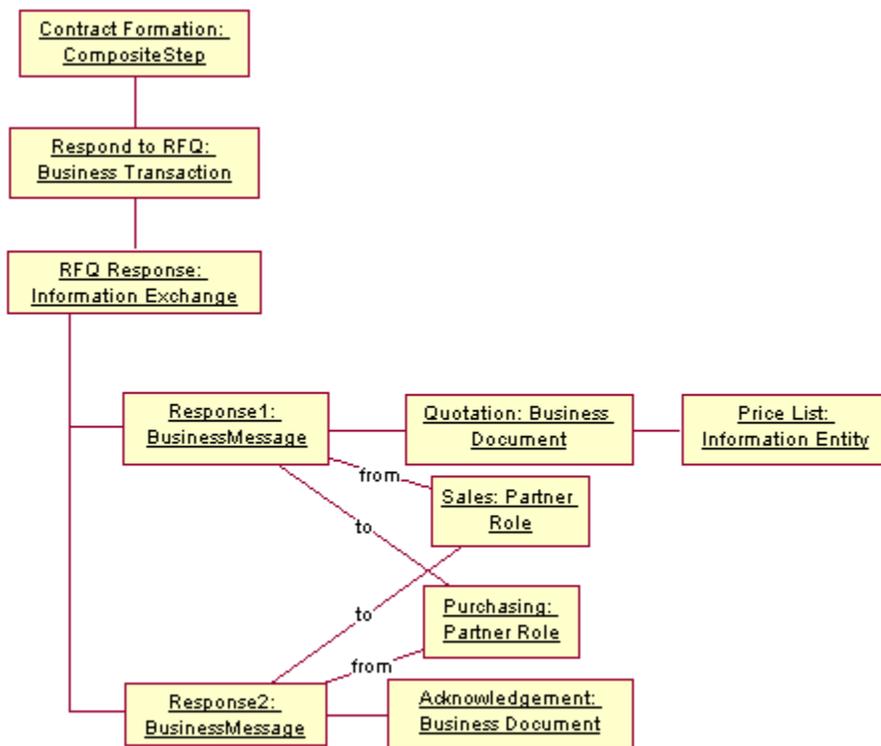
# Contract Formation Step 1



767  
768  
769  
770

771  
772

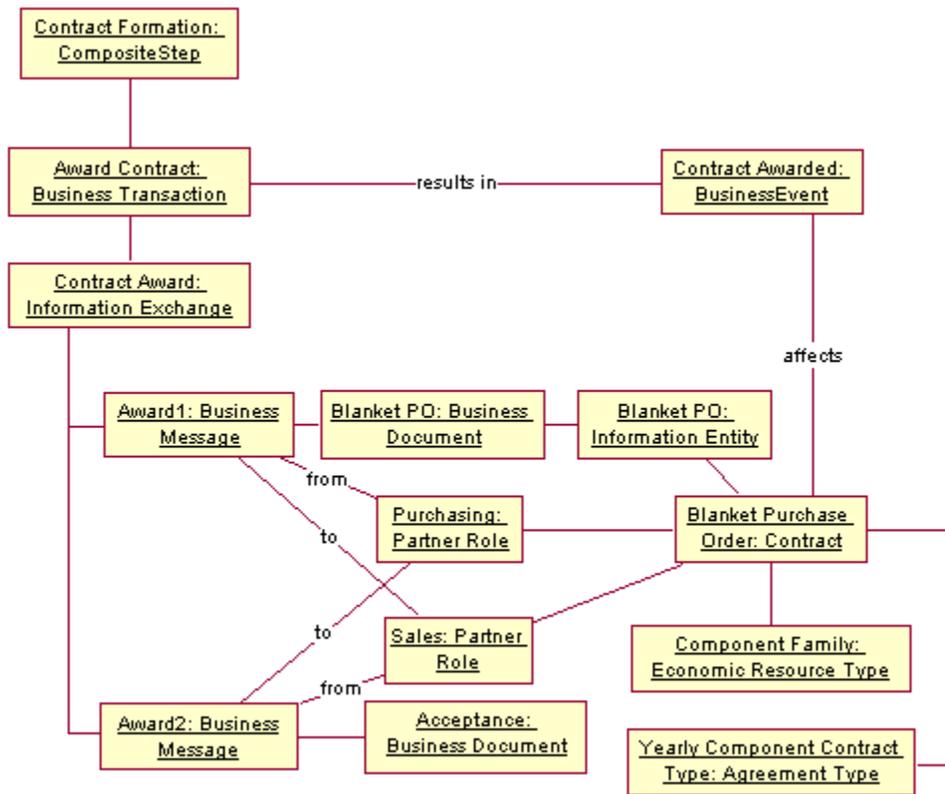
## Contract Formation Step 2



773  
774

775  
776

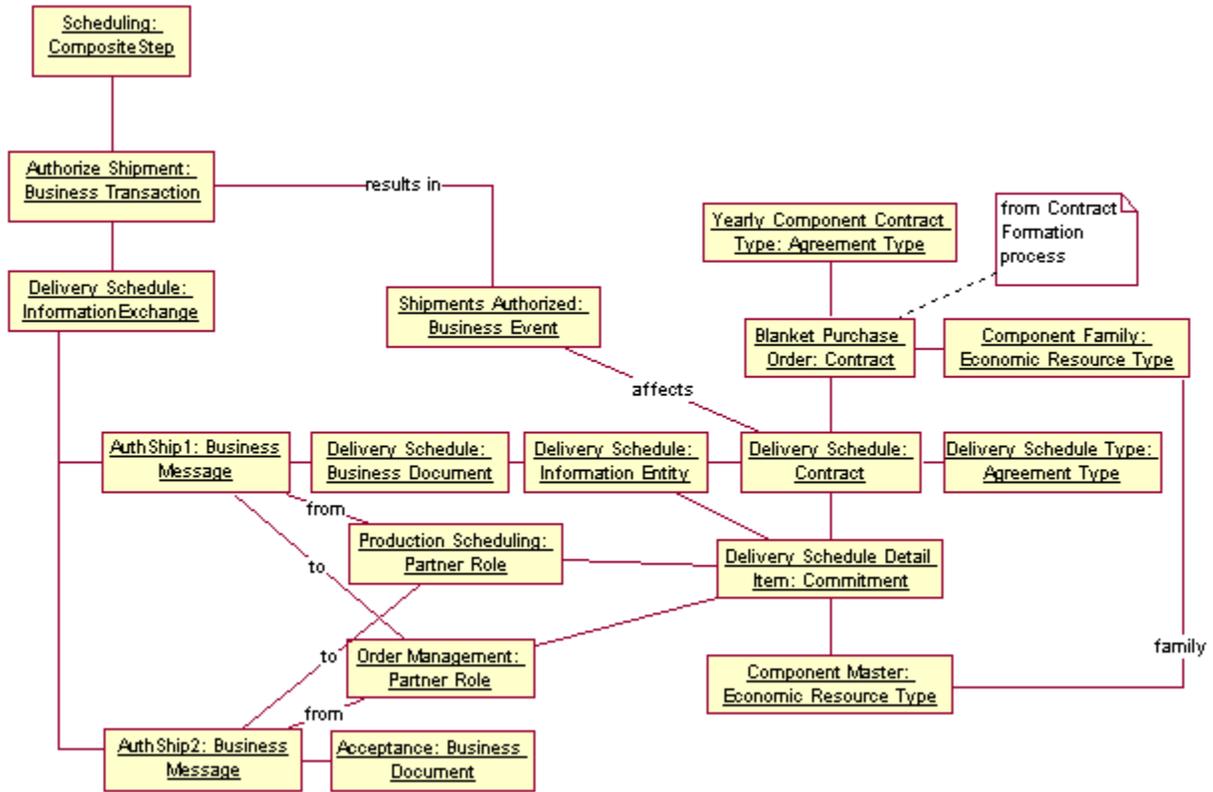
## Contract Formation Final Step



777

778  
779  
780

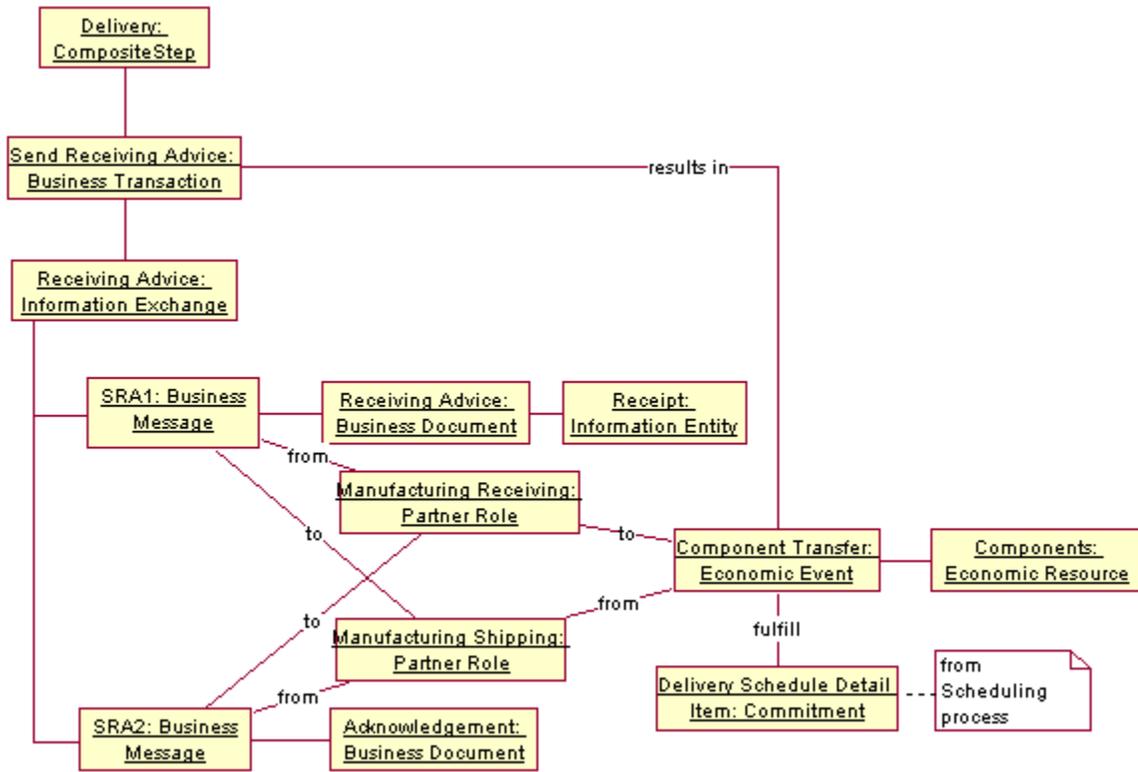
## Scheduling Final Step



781

782  
783

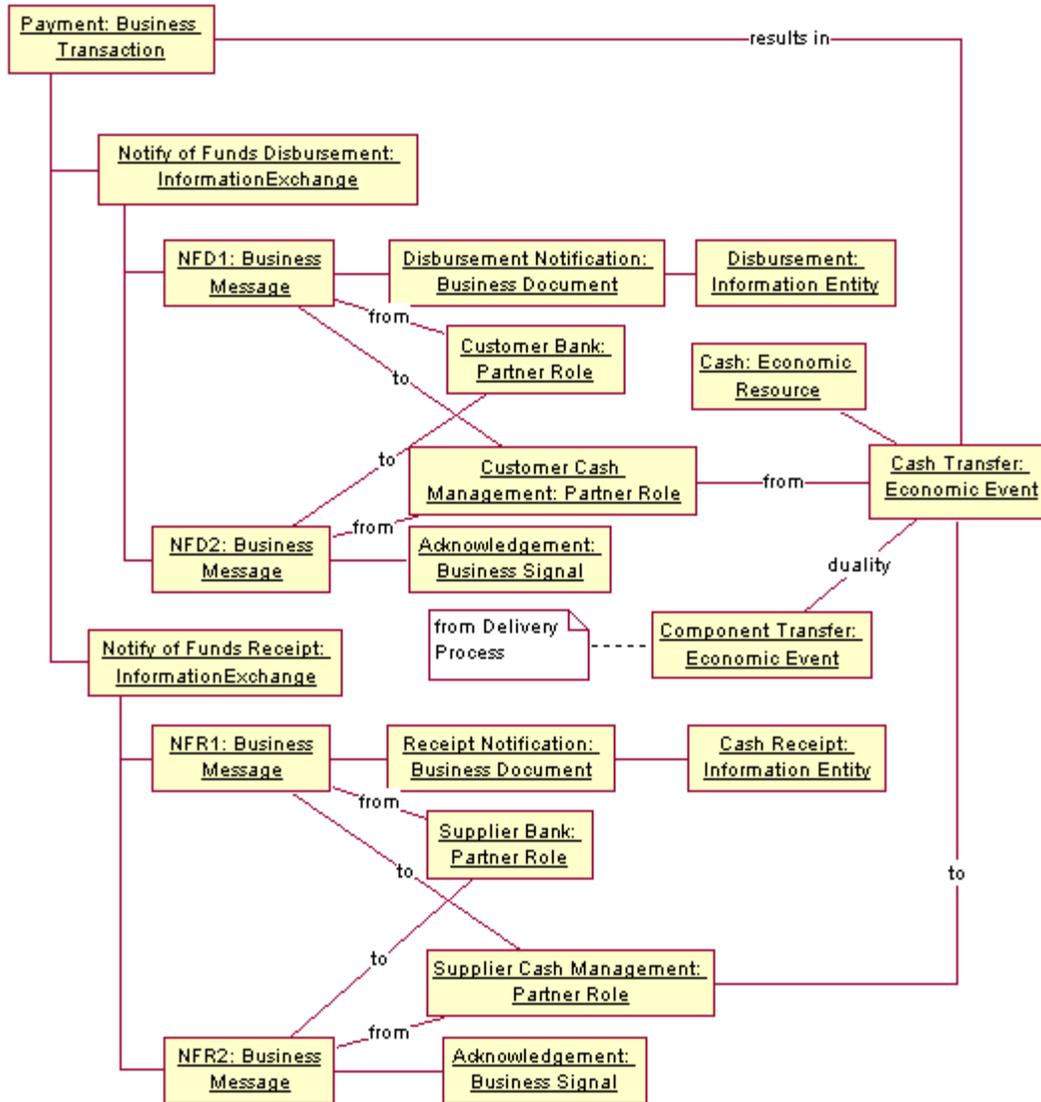
## Delivery Final Step



784

785  
786

## Payment Final Steps



787  
788  
789  
790  
791  
792

793 **Auto procurement practices not captured in current use cases**

794

- 795 1. **Preliminary trading partner agreements** may be formed before contracts are negotiated.  
796 These agreements may not carry any economic commitments. They would be mapped to the  
797 Agreement class in the ebXML metamodel.
- 798 2. **Intermediate consignees** may be used in the Delivery use case, to pool components before  
799 delivery to the point of production, and/or to perform outside services.
- 800 3. **Variations in delivery authorization** include regular purchase orders, delivery schedules,  
801 sequenced delivery schedules, and electronic Kanbans or JIT pull signals.
- 802 4. **Variations in payment authorization** include evaluated receipts settlement, pay on  
803 production, pay to the ASN, and invoices.
- 804 5. **Variations in payment** (to be researched).

805

806

807