

	<b>Table of Contents</b>	<b>Page</b>
1		
2		
3	9 Metamodel.....	1
4	9.1 Business Metamodel.....	1
5	9.1.1 Model Abstract Syntax .....	1
6	9.1.2 Model Semantics .....	4
7	9.1.3 Model Management Abstract Syntax .....	5
8	9.1.4 Model Management Semantics .....	6
9	9.2 Requirements Metamodel.....	8
10	9.2.1 Model Abstract Syntax .....	8
11	9.2.2 Model Semantics .....	17
12	9.2.3 Model Management Abstract Syntax .....	18
13	9.2.4 Model Management Semantics .....	19
14	9.3 Analysis Metamodel.....	21
15	9.3.1 Model Abstract Syntax .....	21
16	9.3.2 Model Semantics .....	31
17	9.3.3 Model Management Abstract Syntax .....	38
18	9.3.4 Model Management Semantics .....	39
19	9.4 Design Metamodel.....	41
20	9.4.1 Model Abstract Syntax .....	41
21	9.4.2 Model Semantics .....	51
22	9.4.3 Model Management Abstract Syntax & Semantics .....	55
23	9.5 Business Information Structure Design Metamodel.....	57
24	9.5.1 Business Information Model Abstract Syntax .....	57
25	9.5.2 Model Semantic.....	60
26		

## 27 9 Metamodel

### 28 9.1 *Business Metamodel*

29

30 This section specifies the abstract syntax and semantics of a Business Operations Map  
31 (BOM) model and model management packages. The abstract syntax of models is defined  
32 using stereotypes and tagged values. The semantics of models are specified using the truth  
33 semantics of well-formed-formula expressed with OCL expressions and with natural  
34 language.

35

#### 36 9.1.1 *Model Abstract Syntax*

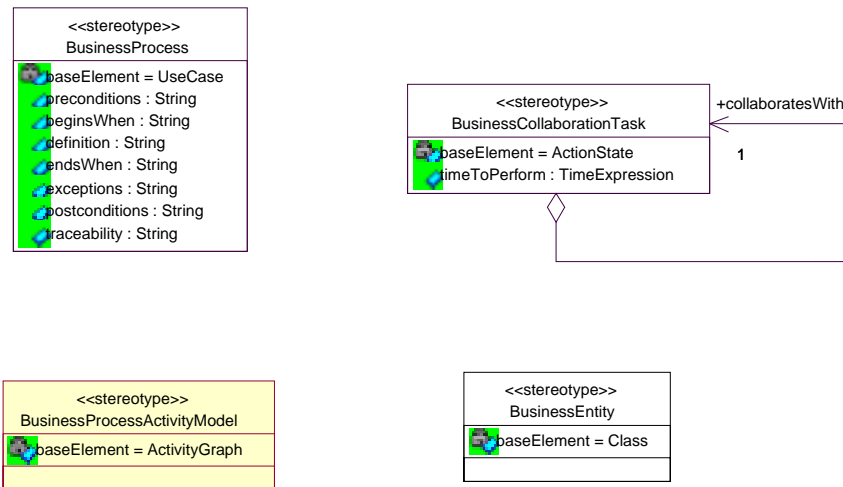
37

##### 38 9.1.1.1 *Stereotypes and Tagged Values*

39

40 Figure 9-1 specifies the modeling elements, and their inter-relationships that are used to  
41 express the structure and behavior of objects in a BOM model. Each element and  
42 interrelationship permitted in a BOM is defined in the metamodel specified in this section  
43 of the document.

## Business Operations Map Abstract Syntax



**Figure 9-1 Business Operations Map Abstract Syntax**

### *BusinessProcess*<sup>1</sup>

A business process is a use case that is used to gather requirements about business processes. Inputs to the business process must be specified in the preconditions and outputs from the business process must be specified in the post-conditions.

#### **Tagged Values:**

- preconditions.* Preconditions are constraints that must be satisfied starting the use case.
- beginsWhen.* Describe the initial event from the actor that starts a use case.
- definition.* A set of simple sentences that state the actions performed as part of the use case. Include references to use cases at extension points.
- endsWhen.* Describe the condition or event that causes normal completion of the use case.

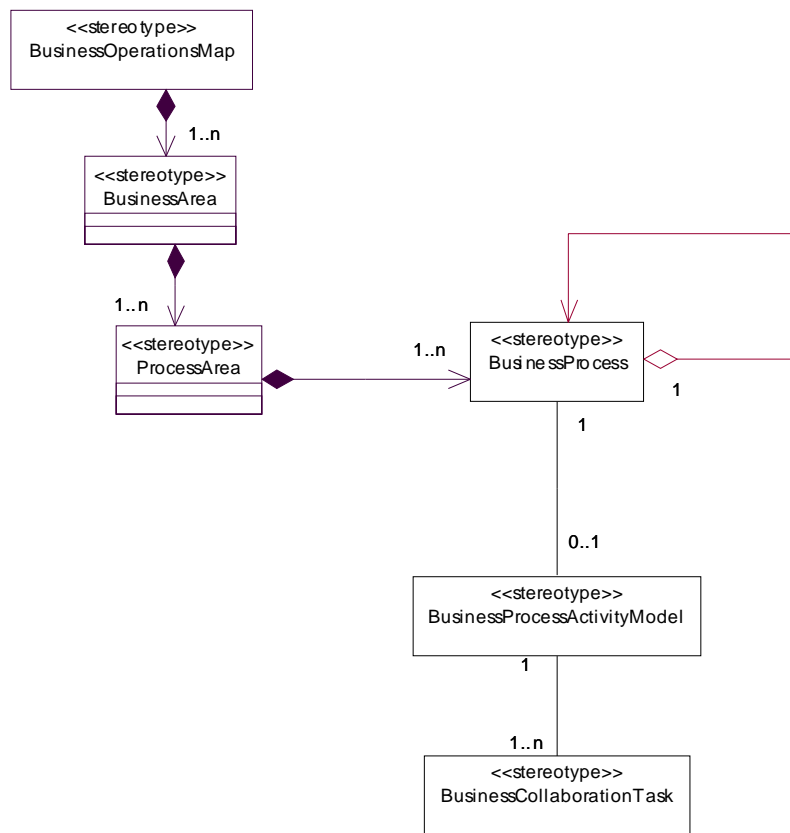
<sup>1</sup> Use cases should consider the inclusion of measure, metric and meter parameters for a business process. Measures are quantifiable properties; a metric is an expression of some performance calculation and a meter is a comparison of the metric to a benchmark.



### 9.1.2 Model Semantics

The semantics of each element of the BOM metamodel is defined in this section. Figure 9-2 illustrates the interrelationships between the BOM modeling elements.

#### Business Operations Map Semantics



**Figure 9-2 Business Operations Map Semantics**

A business process is a sequence of business tasks performed by one business partner alone and business interface tasks performed two or more business partners. A business process activity model should only contain activity states that are either business interface task specifications or that are interpreted as business tasks.

Each task can be further decomposed into activities. Business process can be decomposed into sub-processes using the «include» association stereotype defined in the UML.

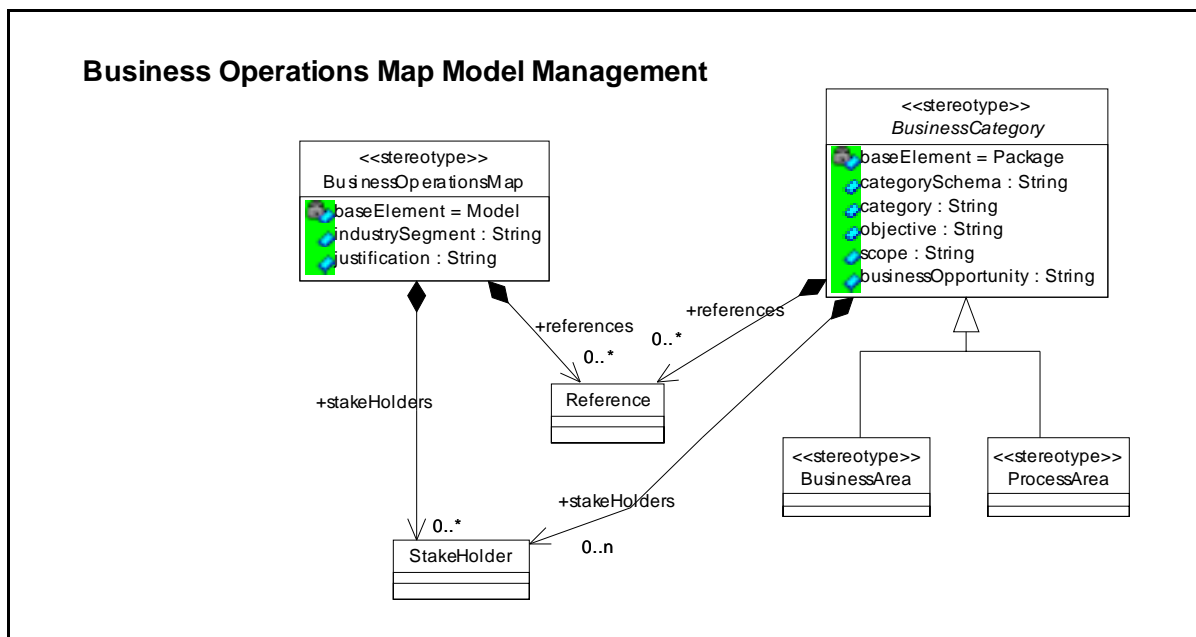
A transition relationship specifies a change in state of a business process that is triggered by the completion of some part of the business process. A transition relates a source business process and a target business process. The direction of the transition is from the source to the target.

### 9.1.3 Model Management Abstract Syntax

The BOM model management organizes business process use cases and business process activity models into a framework of business areas and process areas. These modelling elements are organized as logical, business area and sub-process categories arranged in a framework for understanding their interrelationships. The framework is termed a Business Operations Map (BOM).

#### 9.1.3.1 Stereotypes and Tagged Values

Figure 9-3 shows the metamodel for managing the BOM model. The modelling elements used to manage and organize these three specifications are defined in this section.



**Figure 9-3 Business Operations Map Model Management Abstract Syntax**

The following stereotypes and tagged values are contained in the BOM management metamodel.

#### *BusinessOperationsMap*

A Business Operations Map is a framework for understanding business area sub-process interrelationships. This framework is termed a Business Operations Map (BOM).

#### **Tagged Values:**

146 *industrySegment. ?????? define*  
 147 *justification. ?????? define*  
 148  
 149 *BusinessArea*  
 150 A business area is a category of decomposable business process  
 151 areas. A business area collates business processes areas.

152 *BusinessCategory*  
 153 A business category is an abstraction category for reusing tag-  
 154 values. A business category collates sub-categories.

155 **Tagged Values:**  
 156 *categorySchema.* The name of the categorization schema  
 157 used to reference use cases.  
 158 *category.* The category identifier used to reference a  
 159 business area or business process set of  
 160 use cases.  
 161 *Objective. define*  
 162 *Scope. define*  
 163 *businessOpportunity define*  
 164

165 *ProcessArea*  
 166 A process area is a category of business processes and business  
 167 transactions. A process area collates business processes and  
 168 business transactions.

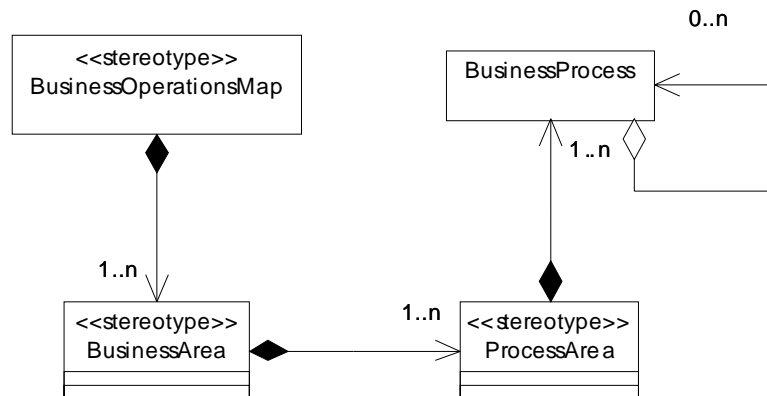
169 **9.1.3.2 Well-formedness Rules**  
 170  
 171 The following well-formedness rules apply to the business operational map metamodel  
 172 package.

- 173 • A BOM must contain at least one Business Area.
- 174 • A Business Area must contain at least one Process Area.
- 175 • A Process Area must contain at least one Business Process.

176

177 **9.1.4 Model Management Semantics**  
 178  
 179 The semantics of each element of the BOM model management metamodel is  
 180 defined in this section.  
 181 Figure 9-4 illustrates the interrelationships between the BOM model management  
 182 elements.

## Business Operations Map Model Management Semantics



**Figure 9-4 Business Operations Map Model Management Illustration**

A business operations map comprises business areas. The [Supply Chain Council](#) defines plan, source, make and deliver business areas in their Supply Chain Operations Reference (SCOR) model. The model describes business processes in the Discrete and Continuous Goods Supply Chain. The [Telecommunications Management Forum](#) defines fulfill, assure and bill business areas in their Telecom Operations Map (TOM). The map describes business processes in the services industry.

Business areas comprise process areas. A process area is a sequence of business processes that implements a particular business model. Business areas such as “Deliver stocked product” and “Deliver make-to-order products” are two different business models that use many of the same business processes.

197

## 198 **9.2 Requirements Metamodel**

199

200 The Business Requirements View (BRV) of a process model specifies the use case  
201 scenarios, input and output triggers, constraints and system boundaries for business  
202 transactions (BTs), business collaboration protocols (BCPs) and their interrelationships.  
203 This section specifies the abstract syntax and semantics of the BRV of a BT and BCP model  
204 and model management packages. The abstract syntax of models is specified using  
205 stereotypes and tagged values. The semantics of models are specified using the truth  
206 semantics of well-formed-formula expressed with OCL expressions and with natural  
207 language.

208

### 209 **9.2.1 Model Abstract Syntax**

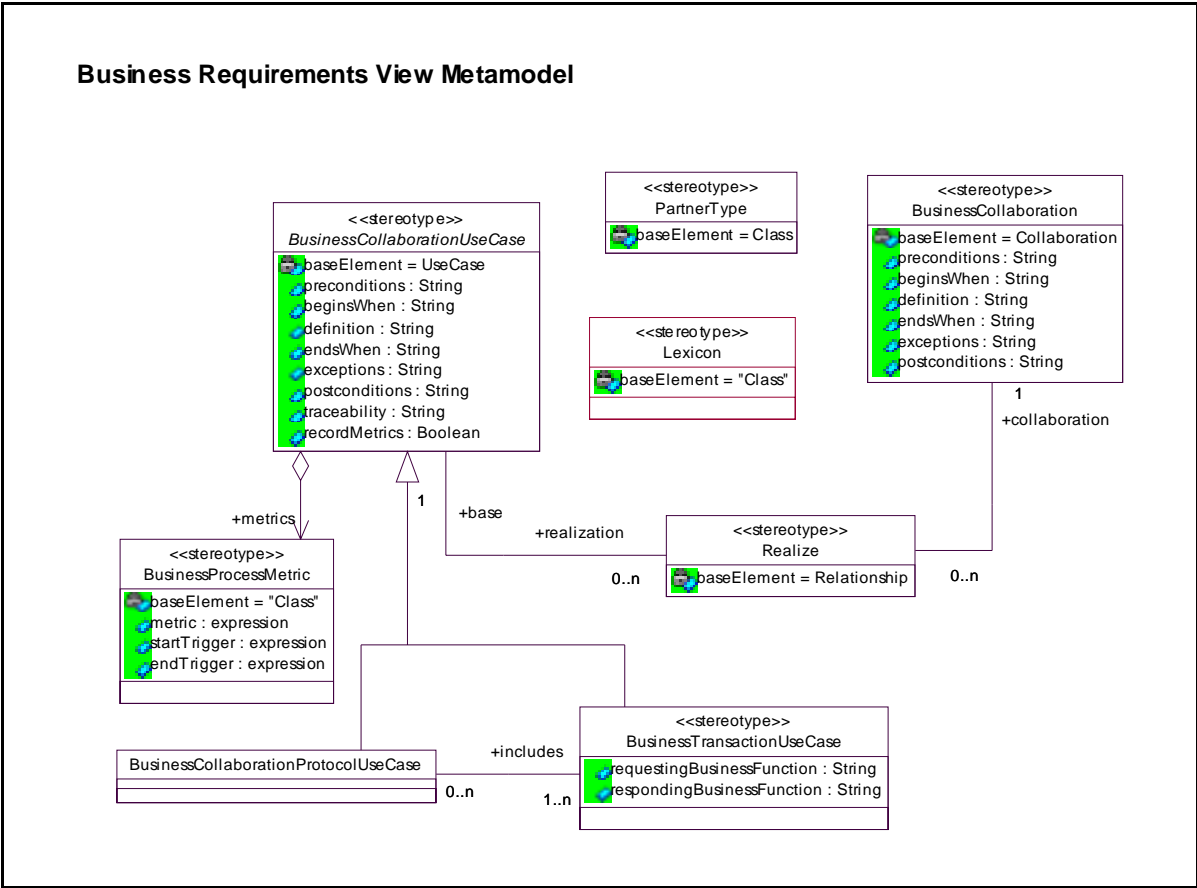
210

#### 211 **9.2.1.1 Stereotypes and Tagged Values**

212

213 Figure 9-5 specifies the modeling elements and their interrelationships  
214 that are used to express the structure and behavior of objects in the BRV  
215 of a BT and BCP model. Each element and interrelationship permitted in  
216 a BRV is defined in the metamodel specified in this section of the  
217 document.





**Figure 9-5 BRV Abstract Syntax**

*BusinessCollaborationProtocolUseCase*

A business collaboration protocol use case is used to gather requirements for e-business collaboration protocol specifications.

*BusinessCollaboration*

A business collaboration model specifies the input and output relationships between business collaboration use cases and Agents. Agents provide input triggers to use cases and business collaboration use cases can provide input triggers and output triggers to and from other business collaboration use cases.

A business collaboration model captures business information constraints imposed by a specific partner type collaboration. For example, sending a business document to a US Government agency requires a Standard Industry Classification (SIC) code to be included with the business information.

**Tagged Values:**

*preconditions*. Conditions that must be true before starting the use case.

237 *beginsWhen.* Describe the initial event from the actor that  
 238 starts a use case.

239 *definition.* A set of simple sentences that state the  
 240 actions performed as part of the use case.  
 241 Include references to use cases at  
 242 extension points.

243 *endsWhen.* Describe the condition or event that causes  
 244 normal completion of the use case.

245 *exceptions.* List all exception conditions that will cause  
 246 the use case to terminate before its normal  
 247 completion.

248 *postconditions.* Conditions that must be true before ending  
 249 the use case.

## 250 *BusinessCollaborationUseCase*

251 A business collaboration use case is an abstraction for a business  
 252 collaboration protocol use case and a business transaction use  
 253 case. The abstraction permits the reuse of the business  
 254 collaboration realization relationship.

255 A completed use case assumes that some one “thing” of  
 256 “measurable value” be created either as a service performed or a  
 257 product created. Four appropriate classes of measure that can be  
 258 applied to use case performance are: quantity measure, quality  
 259 measure, time of performance measure and resource usage or  
 260 consumption measure. Each use case should have an identified  
 261 set of appropriate measures. At a minimum, at least one quantity  
 262 measure should be employed.

### 263 **Tagged Values:**

264 *preconditions.* Conditions that must be true before starting  
 265 the use case.

266 *beginsWhen.* Describe the initial event from the actor that  
 267 starts a use case.

268 *definition.* A set of simple sentences that state the  
 269 actions performed as part of the use case.  
 270 Include references to use cases at  
 271 extension points.

272 *endsWhen.* Describe the condition or event that causes  
 273 normal completion of the use case.

274 *exceptions.* List all exception conditions that will cause  
 275 the use case to terminate before its normal  
 276 completion.

277 *postconditions.* Conditions that must be true before ending  
 278 the use case.

279 *traceability*. An explicit list of requirements, identified by  
280 category, that are either partially or  
281 completely satisfied by this use case.

282

283 **Associations:**

*realization.* A business collaboration is a realization of a  
business collaboration use case.

286

287 *BusinessTransactionUseCase*

288 A business transaction use case is used to gather requirements  
289 for business transaction specifications.

290 Tagged Values:

291                    *requestingBusinessFunction*. The business function that is  
292                    implemented by the requesting business  
293                    partner which is performing a role with  
294                    respect to the use case e.g. procurement.

295                    *respondingBusinessFunction*. The business function that is  
296                    implemented by the responding business  
297                    partner which is performing a role with  
298                    respect to the use case e.g. fulfillment.

299 *Realize*

300 A relationship between a business collaboration Use Case and the  
301 realization of a Use Case.

302 **Associations:**

303 base. The base use case for the collaboration in  
304 the realization relationship.

collaboration. The collaboration realization for the base  
use case.

307 *PartnerType*

308 A partner type is an actor in a business collaboration use case.  
309 Partner types are manufacturer, distributor, retailer, end user,  
310 carrier and financier.

311

Figure 9-6 specifies the modeling elements and their interrelationships that are used to express the structure and behavior of objects in the BRV of an economic model depicting resources, events and agents (REA). Each element and interrelationship permitted in the REA model is defined in the metamodel specified in this section of the document.

## Economic Modeling Elements

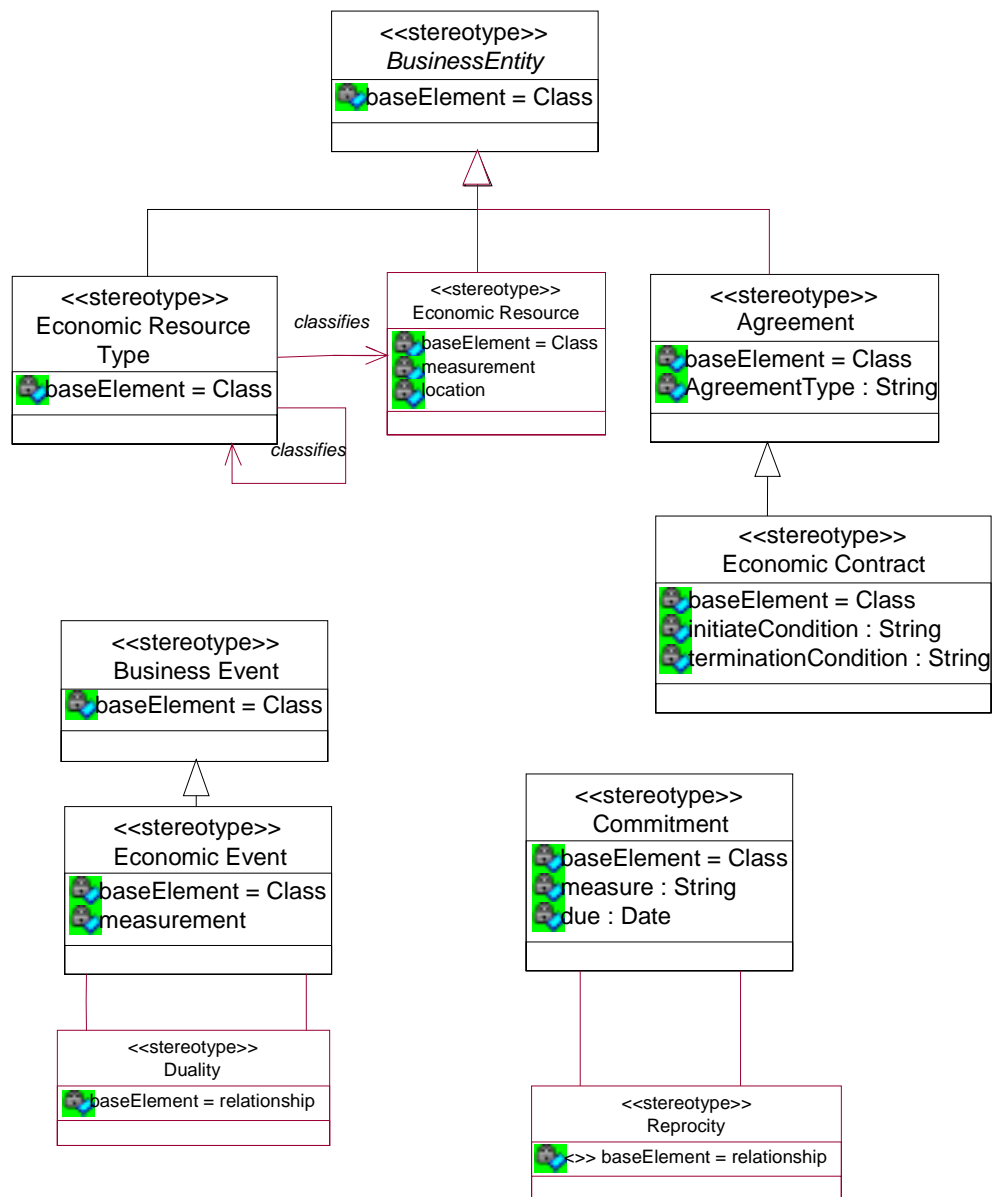


Figure 9-6 Economic Elements Abstract Syntax

319

## *BusinessEntity*

320

Business Entity is an abstraction for any artifact that is important in the execution of a business collaboration.

321

322

## *Agreement*

323

An agreement is an arrangement between two partner types that specifies in advance the conditions under which they will trade (terms of shipment, terms of payment, collaboration protocols, etc.) An agreement does not imply specific economic commitments.

324

325

326

327

### **Tagged Values:**

328

*AgreementType*. *AgreementTypes* classify and structure agreements. For example, an *AgreementType* might specify the kinds of terms and conditions that must be agreed upon for any instance of an agreement of the particular type. Examples of agreement types might include trading partner agreements and yearly economic contracts.

329

330

331

332

333

334

335

336

### **Associations:**

337

*governs*. One agreement may govern another agreement, recursively.

338

339

*participation*. Partner types participate in agreements.

340

## *EconomicContract*

341

A contract is subtype of agreement between partner types that some actual economic exchanges will occur in the future. Contracts can have recursive relationships with other contracts, for example, yearly contracts with monthly releases and weekly or daily shipping schedules. Contracts are containers for collections of commitments. For example, a purchase order is a contract wherein the line items are commitments.

342

343

344

345

346

347

348

### **Tagged Values:**

349

*initiateCondition*. An economic contract term of effect is determined by the *initiateCondition*. This is an OCL constraint and may be defined by measurable elements such as a date, event or system metric.

350

351

352

353

354

*TerminationCondition*. An economic contract is no longer in effect if the *terminationCondition* has been true after the qualification of the *initiateCondition*. This is an OCL constraint and may be defined by measurable elements such as a date, event or system metric.

355

356

357

358

359

360

361                                   **Associations:**  
362                                   *establishes.*   An economic contract establishes two or  
363                                   more commitments.

364                   *Commitment*

365                   An economic commitment is an obligation to perform an economic event  
366                   (that is, transfer ownership of a specified quantity of a specified economic  
367                   resource type) at some future point in time. Order line items are examples of  
368                   commitments.  
369

370                   **Tagged Values:**

371                                   *measure.*       The measurement of an economic resource  
372                                   of the specified type to be transferred.

373                                   *due.*           The condition that determines when the  
374                                   transfer of ownership is promised to occur.  
375                                   This is an OCL constraint and may be  
376                                   defined by elements such as a date, event  
377                                   or system metrics.

378                   **Associations:**

379                                   *fulfills.*       Commitments may be fulfilled by economic  
380                                   events.

381                                   *from.*          A commitment is an obligation from one  
382                                   partner type.

383                                   *to.*            A commitment is an obligation to another  
384                                   partner type.

385                                   *reciprocal.*    A commitment always has reciprocity  
386                                   relationships with one or more other  
387                                   commitments.

388                                   *specifies.*     Commitments specify economic resource  
389                                   types.

390                   *Reciprocity*

391                   Reciprocity is a mandatory relationship between two or more  
392                   commitments. Business contracts require reciprocal commitments,  
393                   called “consideration”.

394                   *EconomicResourceType*

395                   An economic resource type is the abstract classification or definition of an  
396                   economic resource. For example, in an ERP system, ItemMaster or  
397                   ProductMaster would represent the Economic Resource Type that abstractly  
398                   defines an Inventory item or product. Forms of payment are also defined by  
399                   economic resource types, e.g. currency.

400                   **Associations:**

401 *classifies.* Economic resource types classify economic  
402 resources .  
403 *classifies.* Economic Resource Types may have  
404 recursive relationships, so that for example  
405 broad classifications like "product" could  
406 group smaller classifications like "product  
407 family", which in turn could have as  
408 members the specific "product masters"  
409 with SKU numbers.  
410 *specifies.* Commitments specify economic resource  
411 types.

## 412 *EconomicResource*

413 An economic resource is a quantity of something of value that is under the  
414 control of an enterprise, which is transferred from one partner type to  
415 another in economic events. Examples are cash, inventory, labor service and  
416 machine service.

### 417 **Tagged Values:**

418 *measurement.* The number and unit of the economic  
419 resource. Unit may be a unit of measure for  
420 products, a unit of time for services, or a  
421 currency for cash.

422 *location.* The location where the economic resource  
423 currently resides or is available.

### 424 **Associations:**

425 *classifies.* Economic resources are classified by  
426 economic resource types.

427 *resourceFlow.* Economic resources flow from one partner  
428 type to another via economic events.

## 429 *BusinessEvent*

430 A business event is a significant change in the state of one or more entities  
431 within a business, e.g. the taking of an order or a price change.

## 432 *EconomicEvent*

433 An economic event is the transfer of control of an economic resource from  
434 one partner type to another partner type. Examples would include sale, cash-  
435 payment, shipment, and lease.

### 437 **Tagged Values:**

438 *measurement.* The number and unit of the economic  
439 resource. that is being transferred.

### 440 **Associations:**

441	<i>duality.</i>	Duality is a relationship between economic
442		events, where one is the legal or economic
443		consideration of the other. Examples
444		include a payment for a product or service.
445		If one economic event occurs, but its dual or
446		expected consideration has not occurred,
447		the giving partner type has an imputed claim
448		against the taking partner type for the value
449		of the economic resources transferred.
450	<i>fulfills.</i>	An economic event may fulfill a prior
451		commitment.
452	<i>participation.</i>	At least two partner types must participate
453		in an economic event, one to give the
454		economic resources, the other to take them.
455	<i>resourceFlow.</i>	Economic resources flow from one partner
456		type to another via economic events.

## 457 *Duality*

458 Duality is a relationship between Economic Events, where one is the legal or  
459 economic consideration of the other. Examples include a payment for a  
460 product or service. Duality relationships occur between two or more  
461 economic events.

### 463 **9.2.1.2 Well-formedness Rules**

464  
465 The following well-formedness rules apply to the business requirements view metamodel  
466 package.

- 467 • All associations between partner types and business use cases must specify the
- 468 partner type as the source of the association and the source association end must
- 469 have a name that is the role of the partner type with respect to the business
- 470 transaction use case to which it interfaces.
- 471 • A business transaction use case may not be the target of an «extend» association.
- 472 • Business transaction use cases may not be the source of an «include» association.
- 473 • Compliant models must have all use cases stereotyped as
- 474 «BusinessCollaborationProtocolUseCase», to at least be either the source of an
- 475 «include» association or the target of an «extend» association.
- 476 • The name of the association between a partner type and a use case must be the
- 477 name of input/output triggers of the use case.
- 478 • All partner types in the model (classes stereotyped «PartnerType») must be
- 479 defined as partner types e.g. manufacturer, distributor, retailer, carrier, financier
- 480 and end user.
- 481 • Economic contracts must have at least two partner types as participants.
- 482 • Each economic contract must establish at least two commitments.
- 483 • Each commitment must have a reciprocity relationship with at least one other
- 484 commitment.
- 485 • If an economic event fulfills a prior commitment, the economic resource type of the
- 486 economic resource transferred by the economic event must be compatible to the

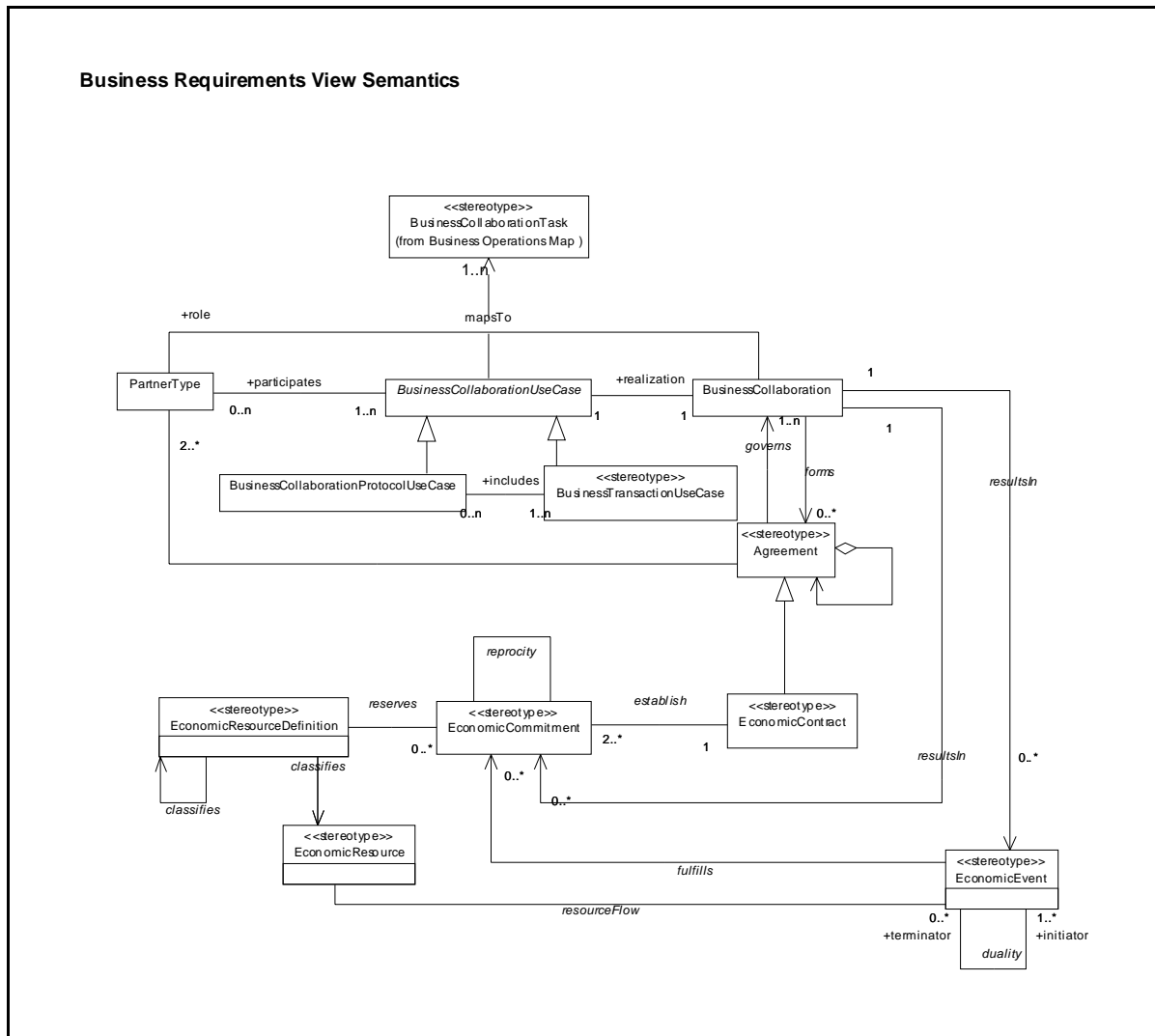


economic resource type promised in the commitment. “Compatible” means either the same type or a subtype of the type of the commitment.

## 9.2.2 Model Semantics

The semantics of each element of the BRV metamodel is defined in this section.

Figure 9-7 illustrates the interrelationships between the BRV modeling elements.



**Figure 9-7 BRV Illustration**

A business collaboration use case maps to two business interface tasks specified in a Business Operations Map. One task is the originator of a business contract and the other is a responder to the business contract. The business collaboration use case can either be a business collaboration protocol specification or a business transaction specification.

501 A business transaction specifies an initiating business partner starting the  
502 contract formation process by communicating a business document request to a  
503 responding business partner. A responding partner accepts the conditions of the  
504 business contract in zero or more returning business signals (e.g. an  
505 acknowledgement of receipt) followed by an optional responding business  
506 document (e.g. an acknowledgement of acceptance)<sup>2</sup>.

507 A business collaboration protocol choreographs business transactions when the  
508 contract formation process requires a number of requesting and responding  
509 business document exchanges. For example the creation of a purchase order  
510 request can be specified as a business collaboration protocol that choreographs  
511 both a purchase order and notification of acceptance business transactions. In  
512 these instances the responding business partner does not accept the entire  
513 purchase order offer in a response to the initial business transaction request.  
514 Instead the partner communicates line item acceptance of the purchase order  
515 using many notifications of acceptance over an agreed period. The contract is  
516 formed when the initiating business partner is able to reconcile all the  
517 notifications of acceptance with the original purchase order request.

518 A partner type performs a specific role in business collaboration. The partner  
519 roles are not employee or organization titles.

520 A business requirements use case should capture both the requirements for  
521 forming business contracts and the requirements for auditing the formation of  
522 business contracts. A business transaction models the start and end of a  
523 business contract formation process. This is not always sufficient to capture the  
524 start and end of an auditable business formation process. For example, an offer  
525 and acceptance contract is formed once an originating partner receives the  
526 agreed "acceptance document". The fact that the sending partner does not  
527 receive a verification of proper receipt for an acceptance business document is  
528 immaterial to the formation of the contract. It may be important, however, if the  
529 sending partner wishes to retain an audit trail of the process for a receiving party  
530 to verify proper receipt of the business document.

531 Economic contracts carry two or more reciprocal commitments, which are  
532 promises that future economic events will occur, specifying particular economic  
533 resource types. Business contracts require reciprocal commitments, called  
534 "considerations". Subsequently, the promised economic events may fulfill the  
535 commitments, transferring ownership of actual economic resources of the  
536 committed types from one partner type to another. For example, a purchase  
537 order is an economic contract, typically committing one partner type to deliver a  
538 product or service of a specified type, and the other partner type to pay for it.  
539 The delivery of the product or service might be the first economic event (fulfilling  
540 one commitment) and obligating (by the duality relationship) the reciprocal  
541 partner type to pay the committed price.

### 542 **9.2.3 Model Management Abstract Syntax**

543  
544 The BRV model can be a business collaboration protocol use case model or a  
545 business transaction use case model, as well as business collaborations.

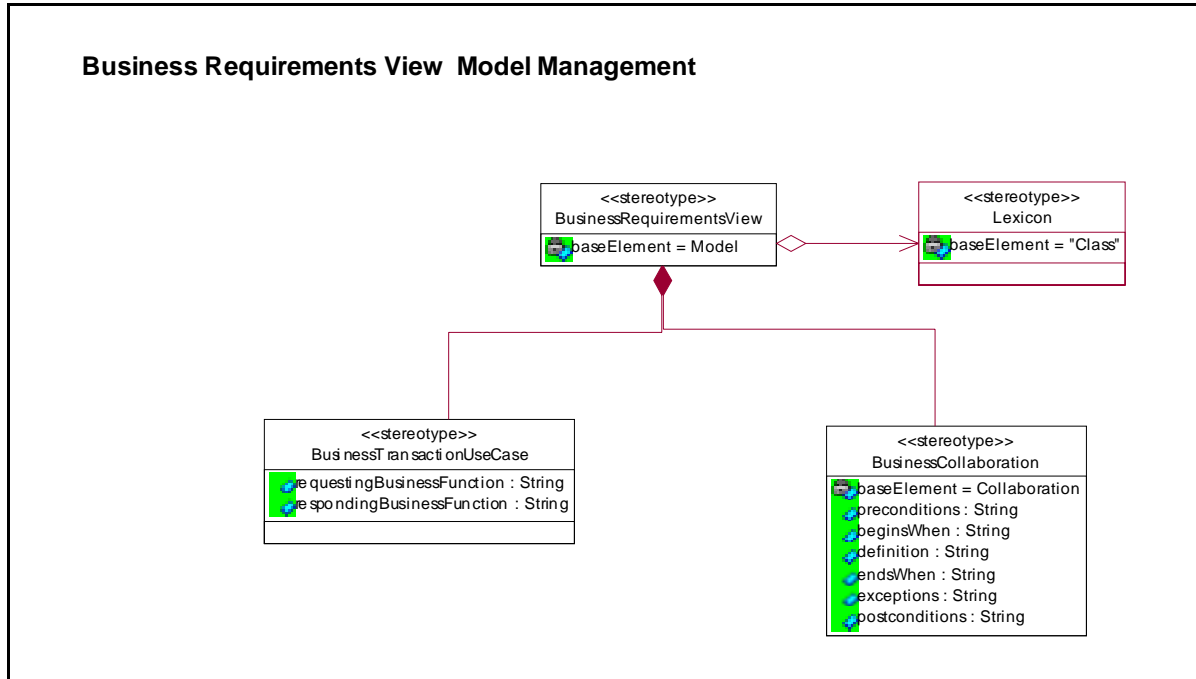
546

---

<sup>2</sup> Business Collaboration Protocol = ( Request Signal\*, Response? ) +

### 9.2.3.1 Stereotypes and Tagged Values

Figure 9-8 shows the metamodel for managing the BRV model. The modeling elements used to manage and organize these modeling elements are defined in this section.



**Figure 9-8 BRV Model Management Abstract Syntax**

The following stereotypes and tagged values are contained in the BRV model management metamodel.

#### *BusinessRequirementsView*

The Business Requirements View specifies the requirements for one or more business collaborations.

### 9.2.3.2 Well-formedness Rules

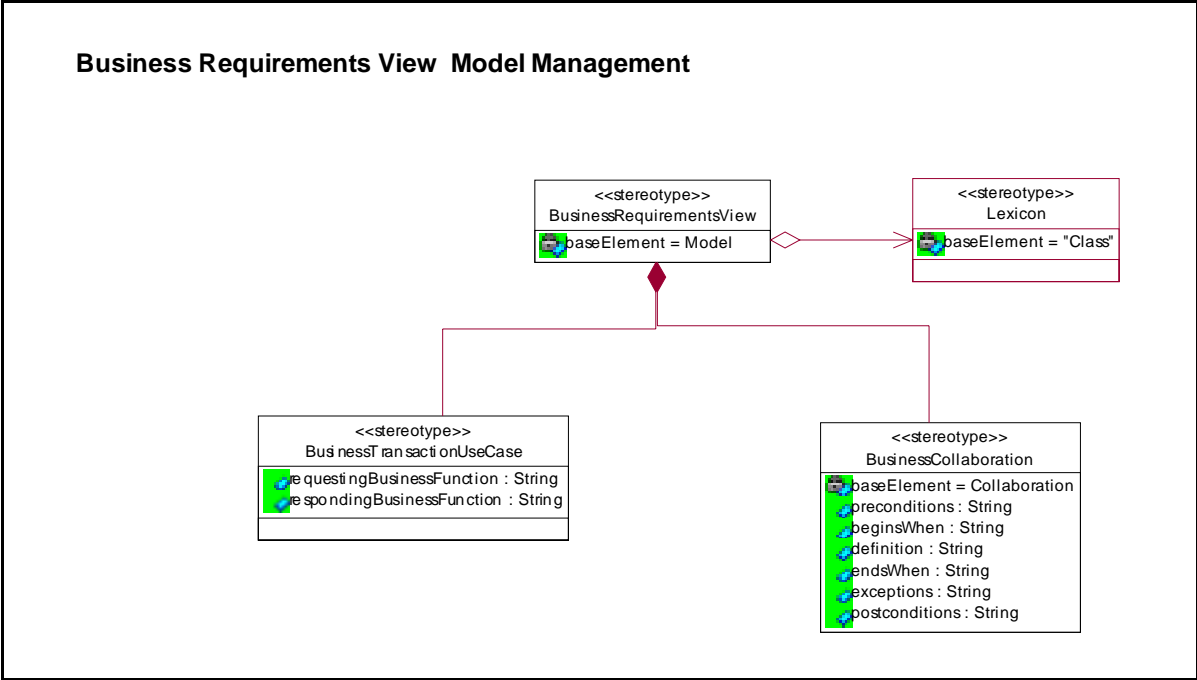
The following well-formedness rules apply to the business requirements view metamodel package.

- A business requirements view model contains one or more Business Transaction Use Case.
- Each Business Transaction Use Case is realized by a Business Collaboration.

### 9.2.4 Model Management Semantics

The semantics of each element of the BRV model management metamodel is defined in this section.

Figure 9-9 illustrates the interrelationships between the BRV model management and model elements.



**Figure 9-9 BRV Model Management Illustration**

A business requirements view is a model of the requirements of a single business collaboration Use Case and its realizations as business collaborations.

577

## 578 **9.3 Analysis Metamodel**

579

580 The Business Transaction View (BTV) of a process model specifies the flow of business  
581 information<sup>3</sup> between business roles as they perform business activities. The business  
582 process specification can be formal as in the formation of offer/acceptance business  
583 contracts as well as informal as in the announcement of new products.

584 This section specifies the abstract syntax and semantics of the BTV of a business  
585 transaction (BT) and business collaboration protocol (BCP) model and model  
586 management packages. The abstract syntax of models is specified using stereotypes  
587 and tagged values. The semantics of models are specified using the truth semantics of  
588 well-formed-formula expressed with OCL expressions and with natural language.

589

### 590 **9.3.1 Model Abstract Syntax**

591

592 The syntax of e-business collaboration models is comprised of stereotypes and  
593 tagged values. The semantics of e-business collaboration models are specified  
594 using the truth semantics of well-formed-formula (specified as OCL expressions)  
595 and with language.

596

#### 597 **9.3.1.1 Stereotypes and Tagged Values**

598

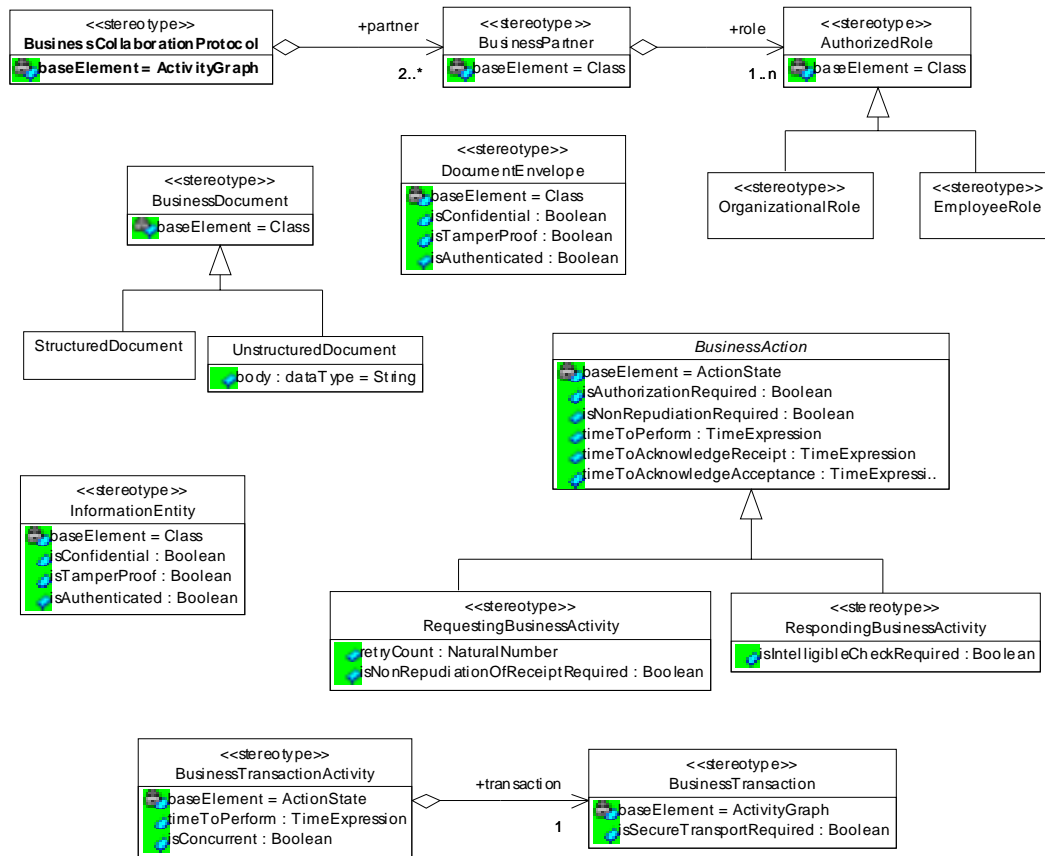
599 Figure 9-10 specifies the modelling elements and their interrelationships  
600 that are used to express the structure and behaviour of objects in the BTV  
601 of a BT and BCP model. Each element and interrelationship permitted in  
602 a BTV is defined in the metamodel specified in this section of the  
603 document.

604

---

<sup>3</sup> The use the term “business information” is intentional as the BRV of a business process must capture the semantics of business information exchanged and not the data format or storage format of the information that is specified in the BSV.

## Business Transaction View Metamodel



**Figure 9-10 BTV Abstract Syntax**

### *BusinessActivity*<sup>4</sup>

The business activity is the state of a business action executed by a partner role during a business transaction. This is an abstract class that is not a stereotype.

### Tagged Values:

*IsAuthorizationRequired.* If a partner role needs authorization to request a business action or

<sup>4</sup> A business activity is derived from the UML Action State model element. This enables multiple exit and entry transitions for the requesting and responding activity states. A business activity is *not* derived from the UML Call State model element that typically models the behavior of an operation. An Activity state does not have an internal transition, exit action or a do activity. The entry action of a Call State is a single call action.

614 to respond to a business action then the  
615 sending partner role must sign the business  
616 document exchanged and the receiving  
617 partner role must validate this business  
618 control and approve the authorizer. A  
619 responding partner must signal an  
620 authorization exception if the sending  
621 partner role is not authorized to perform the  
622 business activity. A sending partner must  
623 send notification of failed authorization if a  
624 responding partner is not authorized to  
625 perform the responding business activity.

626 *isNonRepudiationRequired.* If non-repudiation of origin and  
627 content is required then the business  
628 activity must store the business document in  
629 its original form for the duration mutually  
630 agreed to in a trading partner agreement. A  
631 responding partner must signal a business  
632 control exception if the sending partner role  
633 has not properly delivered their business  
634 document. A requesting partner must send  
635 notification of failed business control if a  
636 responding partner has not properly  
637 delivered their business document.

638  
639 This property provides the following audit  
640 controls:  
641 **Verify sending role identity**  
642 **(authenticate)** – Verify the identity of the  
643 sending role (employee or organization).  
644 For example, a driver's license or passport  
645 document with a picture is used to verify an  
646 individual's identity by comparing the  
647 individual against the picture.  
648 **Verify content integrity** – Verify the  
649 integrity of the original content sent from a  
650 partner role i.e. check that the content has  
651 not been altered by a 3<sup>rd</sup> party while the  
652 content was exchanged between partners.

653 *timeToPerform.* Both partners agree to perform a business  
654 transaction within a specific duration. A  
655 responding partner must exit the transaction  
656 if they are not able to respond to a business  
657 document request within the agreed timeout  
658 period. A sending partner must retry a  
659 business transaction if necessary or must  
660 send notification of failed business control  
661 (possibly revoking a contractual offer) if a  
662 responding partner does not deliver their  
663 business document within the agreed time

664 period. The time to perform is the duration  
665 from the time a business document request  
666 is sent by a requesting partner role until the  
667 time a responding business document is  
668 “properly received” by the requesting  
669 partner role. Both partners agree that the  
670 business signal document or business  
671 action document specified as the document  
672 to return within the time to perform is the  
673 “Acceptance Document” in an on-line  
674 offer/acceptance contract formation  
675 process.

676 *TimeToAcknowledgeReceipt.* Both partners agree to  
677 mutually verify receipt of a requesting  
678 business document within specific time  
679 duration. A responding partner must exit the  
680 transaction if they are not able to verify the  
681 proper receipt of a business document  
682 request within the agree timeout period. A  
683 sending partner must retry a business  
684 transaction if necessary or must send  
685 notification of failed business control  
686 (possibly revoking a contractual offer) if a  
687 responding partner does not verify properly  
688 receipt of a business document request  
689 within the agreed time period. The time to  
690 acknowledge receipt is the duration from the  
691 time a business document request is sent  
692 by a requesting partner until the time a  
693 verification of receipt is “properly received”  
694 by the requesting business partner. This  
695 verification of receipt is an audit-able  
696 business signal and is instrumental in  
697 contractual obligation transfer during a  
698 contract formation process (e.g.  
699 offer/accept).

700 *timeToAcknowledgeAcceptance.* Both partners agree to  
701 the need for a business acceptance  
702 document to be returned by a responding  
703 partner after the requesting business  
704 document passes a set of business rules.  
705 The time to acknowledge business  
706 acceptance of a requesting business  
707 document is the duration from the time a  
708 requesting partner sends a business  
709 document until the time an  
710 acknowledgement of acceptance is  
711 “properly received” by the requesting  
712 partner. A responding partner must exit the  
713 transaction if they are not able to



714 acknowledge business acceptance of a  
715 business document request within the  
716 agreed timeout period. A sending partner  
717 must retry a business transaction if  
718 necessary or must send notification of failed  
719 business control (possibly revoking a  
720 contractual offer) if a responding partner  
721 does not acknowledge acceptance of a  
722 business document within the agreed time  
723 period.

724 *RequestingBusinessActivity*

725 A requesting business activity is a business activity that is  
726 performed by a partner role requesting business service from  
727 another business partner role.

728 **Tagged Values:**

729 *isNonRepudiationOfReceiptRequired.* Both partners agree  
730 to mutually verify receipt of a requesting  
731 business document and that the receipt  
732 must be non-reputable. A receiving partner  
733 must send notification of failed business  
734 control (possibly revoking a contractual  
735 offer) if a responding partner has not  
736 properly delivered their business document.  
737

738 Non-repudiation of receipt provides the  
739 following audit controls.

740 **Verify responding role identity**  
741 (authenticate) – Verify the identity of the  
742 responding role (individual or organization)  
743 that received the requesting business  
744 document.

745 **Verify content integrity** – Verify the  
746 integrity of the original content of the  
747 business document request.

748 *retryCount.* Both partners agree to the number of times  
749 to retry a transaction when a time-out-  
750 exception condition is signaled. This  
751 parameter only applies to time-out signals  
752 and not business process controls or  
753 document content exceptions.

754 *RespondingBusinessActivity*

755 A responding business activity is a business activity that is  
756 performed by a partner role responding to another business  
757 partner role's request for business service.

758 **Tagged Values:**

759 *isIntelligibleCheckRequired*. Both partners agree that a  
760 responding partner role must check that a  
761 requesting document is not garbled  
762 (unreadable, unintelligible) before  
763 verification of proper receipt is returned to  
764 the requesting partner. Verification of  
765 receipt must be returned when a document  
766 is “accessible” but it is preferable to also  
767 check for garbled transmissions at the same  
768 time in a point-to-point synchronous  
769 business network where partners interact  
770 without going through an asynchronous  
771 service provider.

## 772 *InformationEntity*

773 An information entity realizes structured business information that  
774 is exchanged by partner roles performing activities in a business  
775 transaction. Information entities include or reference other  
776 information entities through associations.

777 A secure information entity is an information entity with security  
778 controls. Security controls must be specified when information  
779 must be secured within an enterprise until it is accessed by an  
780 authorized partner role.

781 These parameters on this model element must be specified in a  
782 manner that ensures document integrity by maintaining a “chain-  
783 of-custody” from the sender to the intended recipient of the  
784 business information.

### 785 **Tagged Values:**

786 *isConfidential*. The information entity is encrypted so that  
787 unauthorized parties cannot view the  
788 information.

789 *isTamperProof*. The information entity has an encrypted  
790 message digest that can be used to check if  
791 the message has been tampered with. This  
792 requires a digital signature (sender’s digital  
793 certificate and encrypted message digest)  
794 associated with the document entity.

795 *isAuthenticated*. There is a digital certificate associated  
796 with the document entity. This provides  
797 proof of the signer’s identity.

## 798 *StructuredDocument*

799 A structured document is a information entity container.

800                    *UnstructuredDocument*

801                    An unstructured document is any document that is not comprised

802                    of document entities.

803                    **Tagged Values:**

804                           *dataType.*        This property specifies the document type. It

805                                                   is recommended that a registered MIME

806                                                   type be used for this property (refer to

807                                                   <http://www.iana.org>) for registered MIME

808                                                   types. Partners can agree to use their own

809                                                   experimental MIME types.

810                    *OrganizationalRole*

811                           Only an organization performs a particular role in an e-business

812                           collaboration. An employee does not perform these activities.

813                    *FunctionalRole*

814                           A partner role is a functional role, an employee role or an

815                           organizational role. Either an employee role or an organizational

816                           role can perform a functional role.

817                           An organizational role must be performed by a conforming

818                           business service.

819                    *EmployeeRole*

820                           An employee for business/legal reasons can only perform an

821                           employee role. Usually the details of the employee must be

822                           captured and stored/transmitted to another partner for

823                           auditing/liability purposes when the two partner roles are not in the

824                           same organization.

825                    *BusinessTransaction*

826                           A business transaction is a set of business information and

827                           business signal exchanges between two business partners that

828                           must occur in an agreed format, sequence and time period. If any

829                           of the agreements are violated then the transaction is terminated

830                           and all business information and business signal exchanges must

831                           be discarded. Business transactions can be formal as in the

832                           formation of on-line offer/acceptance business contracts and

833                           informal as in the distribution of product announcements.

834                           Business transactions can be comprised of sub-transactions.

835                    **Tagged Values:**

836                           *isSecureTransportRequired.* Both partners must agree to

837                                                   exchange business information using a

838                                                   secure transport channel. The following

839                                                   security controls ensure that business

840                                                   document content is protected against

841 unauthorized disclosure or modification and  
842 that business services are protected against  
843 unauthorized access. This is a point-to-point  
844 security requirement. Note that this  
845 requirement does not protect business  
846 information once it is off the network and  
847 inside an enterprise. The following are  
848 requirements for secure transport channels.

849 **Authenticate sending role identity –**  
850 Verify the identity of the sending role  
851 (employee or organization) that is initiating  
852 the role interaction (authenticate). For  
853 example, a driver's license or passport  
854 document with a picture is used to verify an  
855 individual's identity by comparing the  
856 individual against the picture.

857 **Authenticate receiving role identity –**  
858 Verify the identity of the receiving role  
859 (employee or organization) that is receiving  
860 the role interaction.

861 **Verify content integrity –** Verify the  
862 integrity of the content exchanged during  
863 the role interaction i.e. check that the  
864 content has not been altered by a 3<sup>rd</sup> party.

865 **Maintain content confidentiality –**  
866 Confidentiality ensures that only the  
867 intended, receiving role can read the  
868 content of the role interaction. Information  
869 exchanged during role interaction must be  
870 encrypted when sent and decrypted when  
871 received. For example, you seal envelopes  
872 so that only the recipient can read the  
873 content.

#### 874 *BusinessCollaborationProtocol*

875 A business collaboration protocol choreographs one or more  
876 business transaction activities. A business collaboration protocol  
877 is not a transaction and should be used in cases where  
878 transaction rollback is inappropriate. For example, a buying  
879 partner may request a purchase order by a selling partner. The  
880 selling partner may partially accept the purchase order and thus  
881 complete the transaction but may only return shipping information  
882 on part of the order. The buying partner is sent any number of  
883 later notifications regarding the outstanding portions of the order  
884 until the order is completely reconciled.

885 *partner.* The partners that collaborate are  
886 enumerated so that they can be associated

887 with the roles that they provide in each of  
888 the business transaction activities.

889 *BusinessPartner*

890 The business partners that participate in business collaborations  
891 are enumerated for each business collaboration protocol. Partners  
892 provide the initiating and responding roles in the protocol.

893 *role.* The roles provided by each of the partners  
894 in the business collaboration protocol. A  
895 partner provides each initiating and  
896 responding role in a business transaction  
897 activity.

898 *BusinessTransactionActivity*

899 A business transaction activity is a business collaboration protocol  
900 activity that executes a specified business transaction. The  
901 business transaction activity can be executed more than once if  
902 the *isConcurrent* property is *true*.

903 **Tagged Values:**

904 *timeToPerform.* Both partners agree to perform a business  
905 transaction activity within a specific  
906 duration. The initiating partner must send a  
907 failure notification to a responding partner  
908 on timeout. A responding partner simple  
909 terminates its activity. The time to perform is  
910 the duration from the time a business  
911 transaction activity initiates the first  
912 business transaction until there is a  
913 transition back to the initiating business  
914 transaction activity. Both partners agree that  
915 the business signal document or business  
916 action document specified as the document  
917 to return within the time to perform is the  
918 "Acceptance Document" in an on-line  
919 offer/acceptance contract formation  
920 process.

921 *transaction.* This property relates a specific business  
922 transaction to a business transaction  
923 activity. The business transaction activity  
924 executes the business transaction.

925 *isConcurrent.* If the business transaction activity is  
926 concurrent then more than one business  
927 transaction can be open at one time. If the  
928 business transaction activity is not  
929 concurrent then only one business  
930 transaction activity can be open at one time.

931

### *DocumentEnvelope*

932

A document envelope is a container for structured and  
unstructured business documents.

933

### 934 **9.3.1.2 Well-formedness Rules**

935

936

The following well-formedness rules apply to the Business Transaction  
View metamodel package.

937

938

### *BusinessActivity*

939

- If non-repudiation is required then the input or returned business document must be a tamper proofed entity.

940

941

- If authorization is required then the input business document and business signal must be an authenticated or a tamper proofed secure entity.

942

943

- The time to acknowledge receipt must be less than the time to acknowledge acceptance if both properties have values.

944

945

946

$\text{timeToAcknowledgeReceipt} < \text{timeToAcknowledgeAcceptance}$

947

- If the time to acknowledge acceptance is null then the time to perform an activity must either be equal to or greater than the time to acknowledge receipt.

948

949

- The time to perform a transaction cannot be null if either the time to acknowledge receipt or the time to acknowledge acceptance is not null.

950

951

- If non-repudiation of receipt is required then the time to acknowledge receipt cannot be null.

952

953

- The time to acknowledge receipt, time to acknowledge acceptance and time to perform cannot be zero.

954

955

- If non-repudiation is required at the requesting business activity, then there must be a responding business document.

956

957

- The time to acknowledge receipt, time to acknowledge acceptance and time to perform properties must be specified for both the requesting and responding business activities and they must be equal.

958

959

960

### *RequestingBusinessActivity*

961

- There must be one input transition whose source state vertex is an initial pseudo state.

962

963

- There must be one output transition whose target state vertex is a final state specifying the state of the machine when the activity is successfully performed.

964

965

- There must be one output transition whose target state vertex is a final state specifying the state of the machine when the activity is not successfully performed.

966

967

- There must be one output transition to an object state that in turn has one output transition to a responding business activity.

968

969

- There must be zero or one input transition from an object state that in turn has one input transition from a responding business activity.

970

971                                    *RespondingBusinessActivity*

972                    •There must be one input transition from an object state that in turn has one input

973                    transition from a requesting business activity.

974                    •There must be zero or one output transition to an object state that in turn has an

975                    output transition to a requesting business activity.

976                                    *Object Flow State*

977                    •The source and target of an object flow must not be the same business activity.

978                    •The source and target of the requesting object flow must be opposite to the source

979                    and target of the responding object flow.

980                                    *Information Entity*

981                    •The associations on an information entity must be aggregation relationships with

982                    other information entities to form a partonomy, a hierarchical decomposable

983                    arrangement of business document parts.

984                    •The information entity associations only must be navigable from a containing entity

985                    to an element entity (has-part relationship).

986                    •Constraints on an information entity association must be specified on the role of the

987                    part (supplier) with respect to the whole (client).

988                    •The client and supplier of an entity association must not be the same entity.

989                                    *Business Collaboration Protocol*

990                    •A business partner cannot provide both the initiating and responding roles of the

991                    same business transaction activity.

992

993    **9.3.2    *Model Semantics***

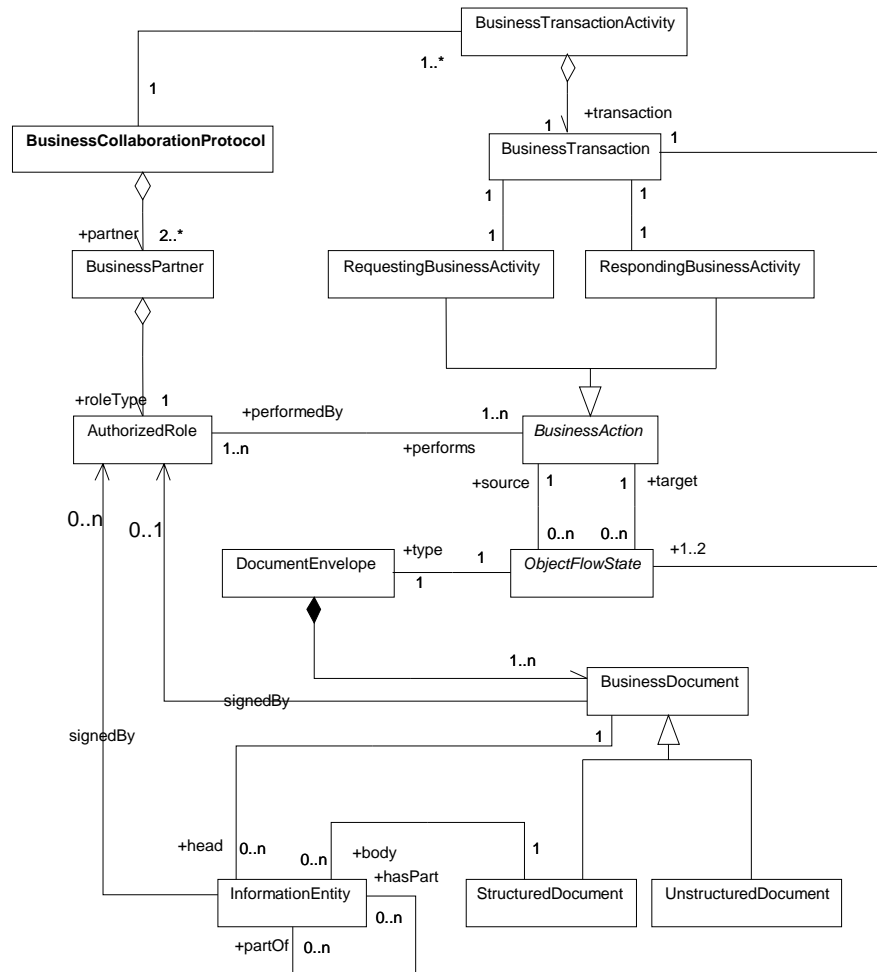
994

995                    The semantics of each element of the BTV model metamodel is defined in this

996                    section.

997                    Figure 9-11 illustrates the interrelationships between the BTV model elements.

## Business Transaction View Metamodel Semantics



**Figure 9-11      BTV Illustration**

#### 9.3.2.1 Business Activities

A business activity is an activity performed by a partner role participating in a business transaction. There are two business activities in a business transaction each of which is performed by one of two partners engaged in a business endeavor. A business partner that is initiating the business transaction performs a requesting business activity. A business partner that is responding to a request to engage in a business transaction performs a responding business activity.

A business transaction specifies either a synchronous or asynchronous flow of control between two activities. The business transaction is a unit of



1011 work. All of the interactions in a business transaction must succeed or the  
 1012 transaction must be rolled back to a defined state before the transaction  
 1013 was initiated.

1014 There are two business signals that can be asynchronously returned to  
 1015 the initiator of the business transaction: a business signal to verify proper  
 1016 receipt of a business document request and a business signal to non-  
 1017 substantively confirm the acceptance of a requesting business document  
 1018 for business processing.

1019 If any of the time out parameters are exceeded, a time out exception must  
 1020 be thrown. If the *retryCount* property on the responding business activity  
 1021 is greater than zero then the business transaction must be re-initiated (or  
 1022 a notification of failed business control – possibly revoking a contractual  
 1023 offer – must be sent). All business signals and business documents  
 1024 returned after the transaction was initiated and up until the time out  
 1025 exception must be discarded. The recurrence property specifies the  
 1026 number of times a business transaction must be initiated. If the recurrent  
 1027 property value is 3 then the business transaction can be initiated a total of  
 1028 4 times (the first initiation plus 3 retries). The time to perform property  
 1029 specifies the time to perform a single business transaction.

1030 A responding partner simply terminates if a timeout is thrown. This  
 1031 prevents responding business transactions from hanging indefinitely.

1032 A partner role that initiates an asynchronous business transaction does  
 1033 not need to receive any business signals. A partner role that initiates a  
 1034 synchronous business transaction must be able to receive business  
 1035 signals and must block until the flow of control is returned. This should not  
 1036 preclude the initiation and execution of multiple concurrent business  
 1037 transactions, however.

1038 If any business exceptions (includes negative receipt and acceptance  
 1039 acknowledgements) are signaled then the business transaction must  
 1040 terminate. The business transaction must not be re-initiated even if the  
 1041 *retryCount* parameter is not zero. Business transactions must only be  
 1042 retried if a timeout exception is thrown.

1043 There are two business signals that are used for on-line business contract  
 1044 formation and auditing:

1045       Acknowledge receipt business signal. The UN/EDIFACT model  
 1046       Trading Partner Agreement (TPA) suggests that partners  
 1047       should agree on the point at which a message can be "said" to  
 1048       be properly received and this point should be when a receiving  
 1049       partner can "read" a message. They suggest this should be the  
 1050       point after which a message passes a structure/ schema  
 1051       validity check. Note that this is not a necessary condition for  
 1052       verifying proper receipt, only accessibility is. The property  
 1053       *isIntelligibleCheckRequired* allows partners to agree that a  
 1054       message should be "readable" before its receipt is verified<sup>5</sup>.

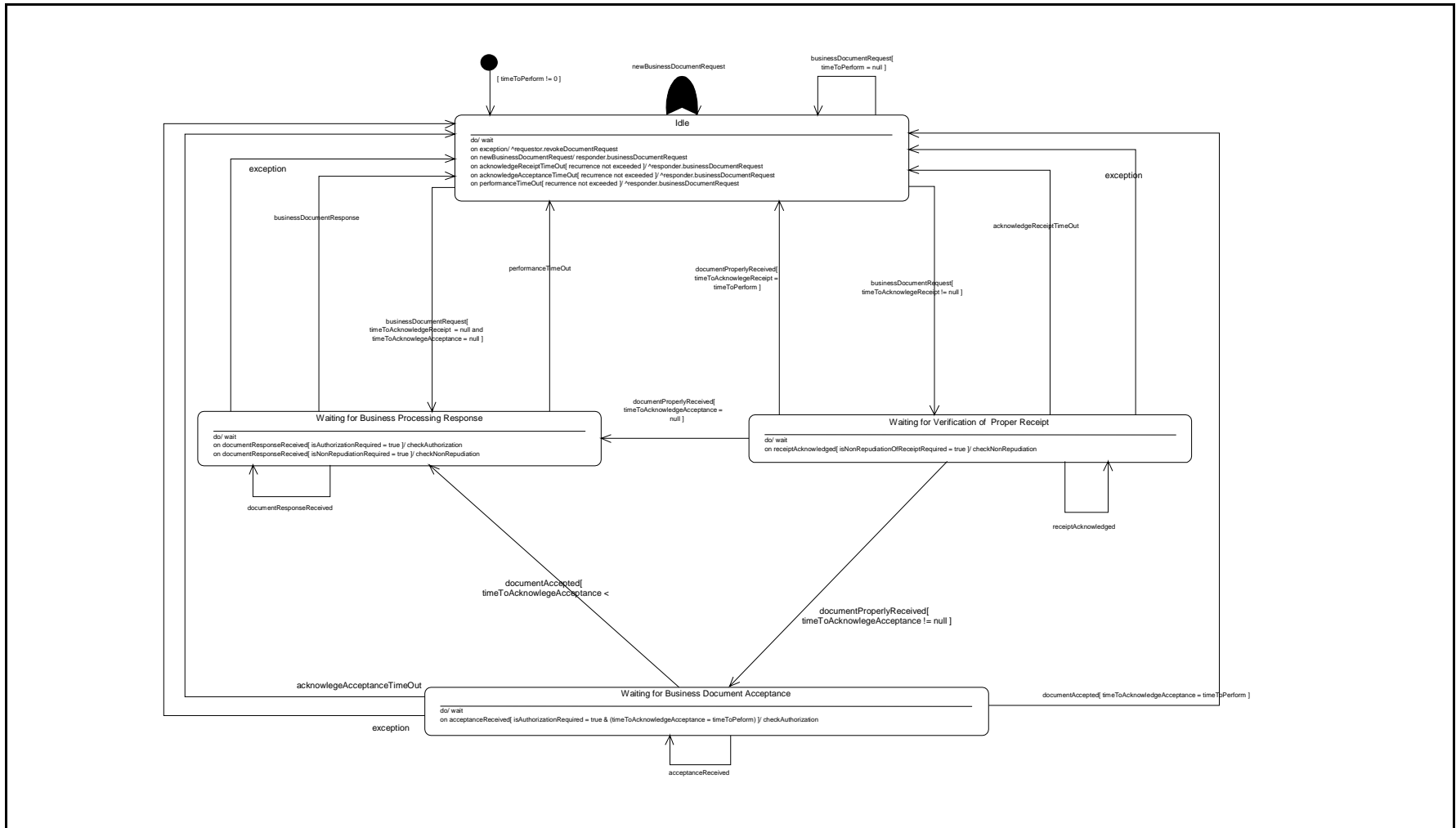
---

<sup>5</sup> This is the convention specified for RosettaNet commercial transactions.

1055 Acknowledge acceptance business signal. The UN/EDIFACT model  
1056 TPA suggests that partners should agree on the point at which  
1057 a message can be "said" to be accepted for business  
1058 processing and this point should be after the contents of a  
1059 business document have passed a business rule validity check.  
1060 For example, if 100,000,000,000 copies of a single book from  
1061 Amazon are ordered, it can be assumed the order will fail some  
1062 business rule check. These business rules are often found in  
1063 trading contracts.

1064 Figure 9-12 and Figure 9-13 show the valid activity states for requesting  
1065 and responding partner roles respectively. The behavior of each role is  
1066 determined by the values specified for each business activity.



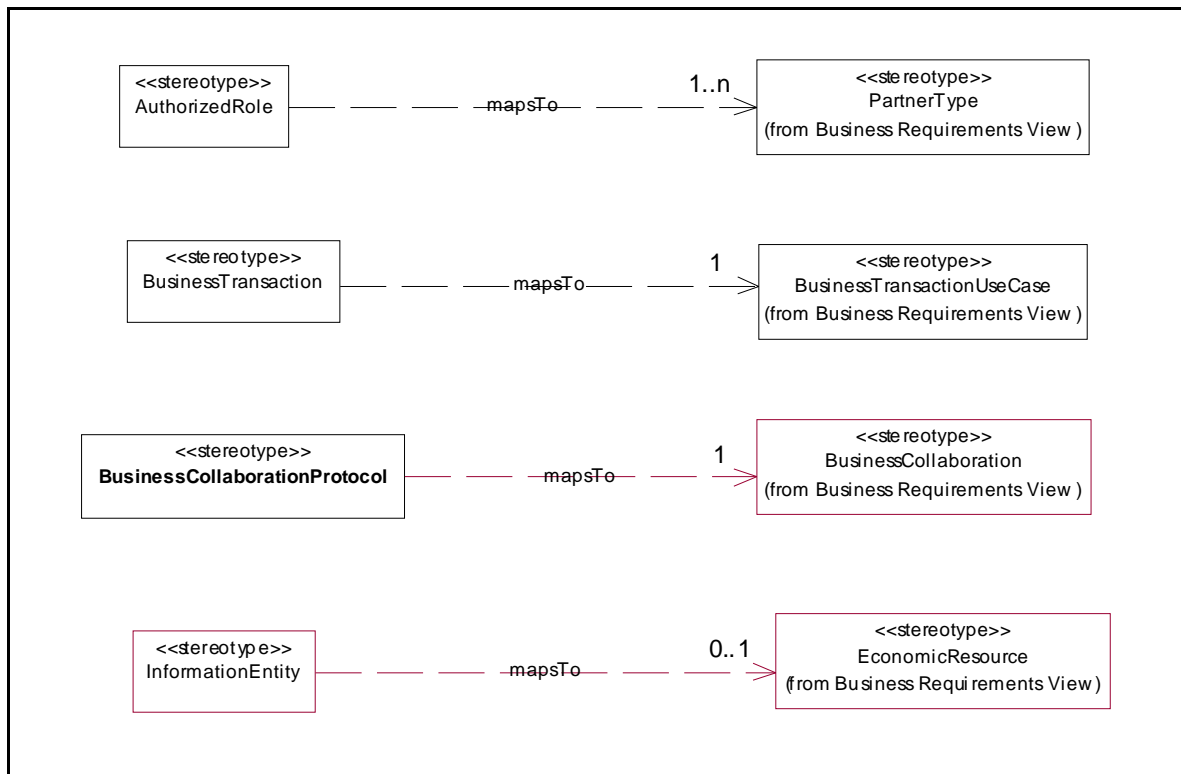


**Figure 9-12 Requesting Business Activity States**



### 9.3.2.2 BTV-to-BRV Mapping

A BTV model is the Business Transaction View of a business process that meets the requirements of a business process as described in a BRV model. Figure 9-14 illustrates the elements of the BTV metamodel that map to elements of the BRV metamodel.



**Figure 9-14 BTV-to-BRV Syntax Map**

A functional role in the BTV is a refinement of a partner type performing a particular role as described in a business transaction Use Case. A business transaction is an activity graph that is a refinement of a business transaction Use Case. Partner roles are modeled in a business transaction activity graph and partner types and their roles are modeled in a Use Case model. The conditional constraints on business information that are described in BRV collaborations are described using business information entity constraints and business information constraints.

A business collaboration protocol activity graph is a refinement of a business collaboration protocol Use Case.

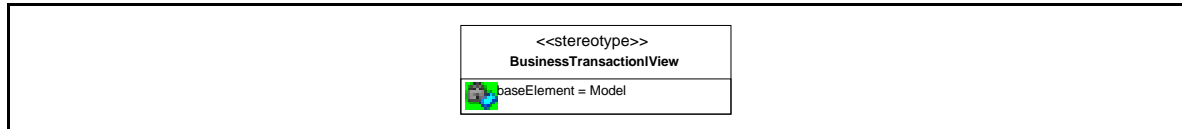
### 9.3.3 Model Management Abstract Syntax

Business process models specify business process participants interacting while executing a business process. A complete business process model must comprise the following modeling elements: business process information, business process participants, and business process flow. The modeling

elements used to manage and organize these three modeling elements are described in this section.

### 9.3.3.1 Stereotypes and Tagged Values

Figure 9-15 shows the metamodel for managing business process models. The modeling elements used to manage and organize these three specifications are described in this section.



**Figure 9-15 BTV Model Management Abstract Syntax**

The following stereotypes and tagged values are contained in the Business Transaction View management metamodel.

#### *BusinessTransactionView*

The business transaction view of an e-business collaboration model comprises diagrams and specifications that show the flow of business data entities between roles as they perform business activities.

### 9.3.3.2 Well-formedness Rules

The following well-formedness rules apply to the Business Transaction View metamodel package.

#### *BusinessTransactionView*

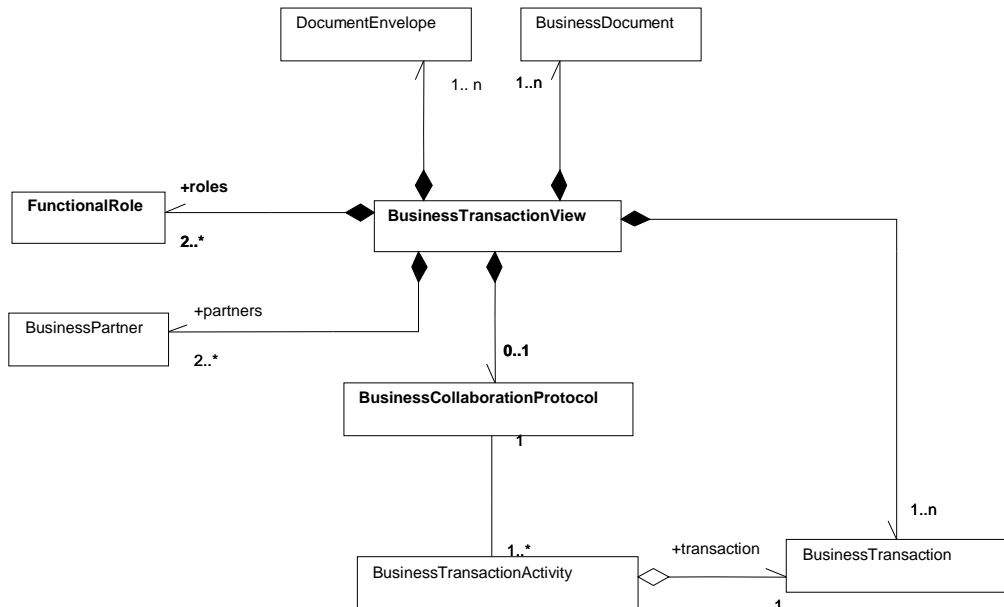
- [1] A business transaction view must comprise one business transaction or business collaboration protocol state machine.

### 9.3.4 Model Management Semantics

The semantics of each element of the BTV model management metamodel is defined in this section.

Figure 9-16 illustrates the interrelationships between the BTV model management and model elements.

## Business Transaction View Model Management Semantics



**Figure 9-16 BTV Model Management Illustration**

The Business Transaction View contains all the objects and activity graphs in the BTV model. A BTV model can comprise zero or one business collaboration protocol specification and can comprise one or more business transaction specifications.



## **9.4 Design Metamodel**

The Business Service View (BSV) Metamodel captures the syntax and semantics of business actions and their exchange between network components that provide business services. The BSV's metamodel specifies the elements of an execution process (Service Collaboration) that comprises business transaction exchange between network component business services as a result of the execution of business activities. The functional service model is a reification of the Business Transaction View model.

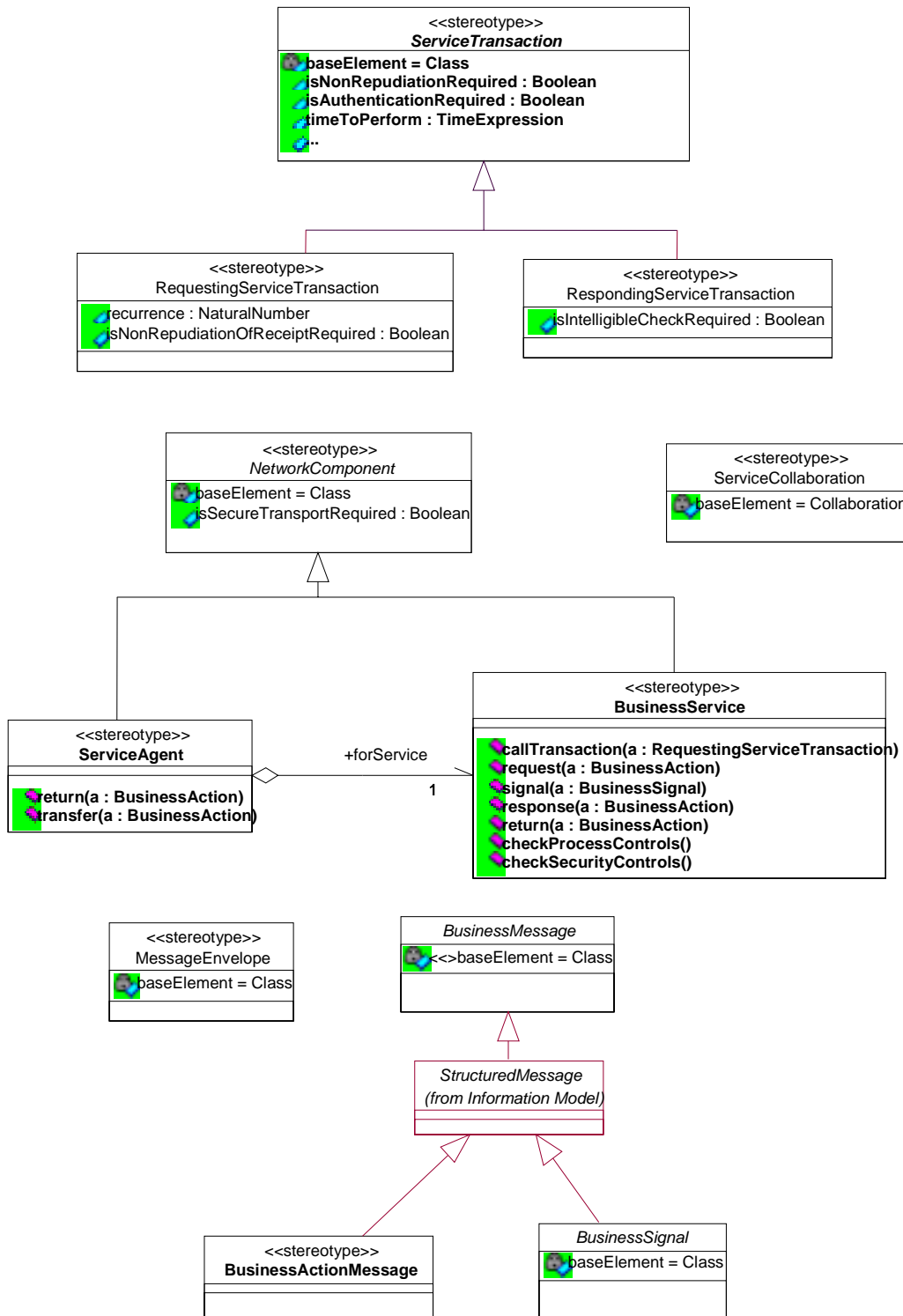
The first part of this section specifies the syntax and semantics of execution processes. The second part of this section specifies the organizational management elements of these execution process models.

### **9.4.1 Model Abstract Syntax**

#### **9.4.1.1 Stereotypes and Tagged Values**

Figure 9-17 specifies the modeling elements and their interrelationships that are used to express the structure and behavior of objects in the BSV of a Business Transaction and Business Collaboration Protocol model. Each element and interrelationship permitted in a BSV is defined in the metamodel specified in this section of the document.

## Business Service View Metamodel (Collaboration Elements)



**Figure 9-17      BSV Abstract Syntax**

### *ServiceAgent*

A *ServiceAgent* is a business communications component that must implement protocols up to the agent layer of the e-business application communications model.

#### **Associations:**

*forService.*      An *ServiceAgent* acts on behalf of a service.

#### **Operations:**

*return(a:BusinessActionMessage).* Return a business action message to this *ServiceAgent*. This *ServiceAgent* becomes the owner of the business action. The argument may not be null.

*transfer(a: BusinessActionMessage).* Transfer a business action message to this agent. This *ServiceAgent* becomes the owner of the business action. The argument may not be null.

### *BusinessService*

A business service is a network component that responds to business transaction requests initiated by other services.

#### **Operations:**

*callTransaction(a: RequestingServiceTransaction).*

*response(a:BusinessAction).* Response to a timed (synchronous) business action request.

*request(a:BusinessAction).* Request to perform a business action. This request can be timed or asynchronous.

*signal(a:BusinessAction).* Asynchronous signal returned for security, auditing and execution control.

*return(a:BusinessAction).* Return a business transaction from an enterprise component after a business action has been performed.

*checkProcessControls().* Requests the Business Service to validate the current state of the current business transaction.

*checkSecurityControls()*.Requests the Business Service to validate the security controls of the current business transaction.

**Associations:**

*transactions*. The *ServiceTransactions* that support this *BusinessService*.

*ServiceTransaction*

A *ServiceTransaction* is a mutually binding interaction between an initiating service and a responding service.

**Tagged Values:**

*isNonRepudiationRequired*. If non-repudiation of origin and content is required then the business activity must store the business document in its original form for the duration mutually agreed to in a trading partner agreement. A responding partner must signal a business control exception if the sending partner role has not properly delivered their business document. A requesting partner must send notification of failed business control if a responding partner has not properly delivered their business document.

This property provides the following audit controls:

**Verify sending role identity (authenticate)**<sup>6</sup> – Verify the identity of the sending role (employee or organization). For example, a driver's license or passport document with a picture is used to verify an individual's identity by comparing the individual against the picture.

**Verify content integrity** – Verify the integrity of the original content sent from a partner role i.e. check that the content has not been altered by a 3<sup>rd</sup> party while the content was exchanged between partners.

*timeToPerform*. Both partners agree to perform a business transaction within a specific duration. A responding partner must exit the transaction if they are not able to respond to a business document request within the agreed timeout period. A

---

<sup>6</sup> The BCF specifies digital signatures for partner-to-partner non-repudiation of origin and content.

sending partner must retry a business transaction if necessary or must send notification of failed business control (possibly revoking a contractual offer) if a responding partner does not deliver their business document within the agreed time period. The time to perform is the duration from the time a business document request is sent by a requesting partner role until the time a responding business document is “properly received” by the requesting partner role. Both partners agree that the business signal document or business action document specified as the document to return within the time to perform is the “Acceptance Document” in an on-line offer/acceptance contract formation process.

*timeToAcknowledgeReceipt.* Both partners agree to mutually verify receipt of a requesting business document within specific time duration. A responding partner must exit the transaction if they are not able to verify the proper receipt of a business document request within the agreed timeout period. A sending partner must retry a business transaction if necessary or must send notification of failed business control (possibly revoking a contractual offer) if a responding partner does not verify properly receipt of a business document request within the agreed time period. The time to acknowledge receipt is the duration from the time a business document request is sent by a requesting partner until the time a verification of receipt is “properly received” by the requesting business partner. This verification of receipt is an audit-able business signal and is instrumental in contractual obligation transfer during a contract formation process (e.g. offer/accept).

*timeToAcknowledgeAcceptance.* Both partners agree to the need for a business acceptance document to be returned by a responding partner after the requesting business document passes a set of business rules. The time to acknowledge business acceptance of a requesting business document is the duration from the time a requesting partner sends a business

document until the time an acknowledgement of acceptance is “properly received” by the requesting partner. A responding partner must exit the transaction if they are not able to acknowledge business acceptance of a business document request within the agreed timeout period. A sending partner must retry a business transaction if necessary or must send notification of failed business control (possibly revoking a contractual offer) if a responding partner does not acknowledge acceptance of a business document within the agreed time period.

#### **Associations:**

*requestingAction.* The *BusinessActionMessage* that initiates this *ServiceTransaction*.

*respondingAction.* The *BusinessActionMessage* that is the response to *theRequestingAction*. Not all requesting actions require a response message. In this case a ‘non-substantive’ acknowledgement is sufficient.

*receiptAcknowledgement.* A *BusinessSignalMessage* that affirms receipt of a *BusinessActionMessage*.

*exceptions.* *BusinessSignalMessages* that report control or process exceptions.

*acceptanceAcknowledgement.* An acceptanceAcknowledgement is a *BusinessSignalMessage* that affirms the acceptance of a action request. This business signal is an acceptance from a legal viewpoint. Through this acceptance mechanism, responsibility for the transaction is transferred to the responding business service.

#### *NetworkComponent*

A network component is a logical computing component in a distributed network environment.

#### **Tagged Values:**

*isSecuredTransportRequired*. Both partners must agree to exchange business information using a secure transport channel. The security controls ensure that business document content is protected against unauthorized disclosure or modification and that business services are protected against unauthorized access. This value is derived from the *isSecuredTransportRequired* property of the *BusinessTransaction* in the BTV.

### *BusinessMessage*

A *BusinessMessage* is a document or information that is exchange between business processes.

#### **Associations:**

<i>header</i> .	Message header that contains security, signature and dictionary reference information.
-----------------	--

### *MessageEnvelope*

A *MessageEnvelope* is container used to route *BusinessActionMessages*.

#### **Associations:**

<i>header</i> .	Message header that contains security, signature and dictionary reference information.
<i>body</i> .	One or more business messages that are carried with this envelope.
<i>prototype</i> .	Identification of the message envelope prototype.

### *BusinessActionMessage*

A *BusinessActionMessage* is a specialized *StructuredMessage* used to convey *BusinessDocuments* (from BTV) between two business processes via a network component.

### *BusinessSignalMessage*

A *BusinessSignalMessage* is used to convey control and exception conditions between two business processes.

#### **Associations:**

<i>forAction</i> .	References the <i>BusinessActionMessage</i> that this <i>BusinessSignalMessage</i> correlates to. Signals are returned to an initiating service by a responding service.
--------------------	--

### *RequestingServiceTransaction*

A *RequestingServiceTransaction* is the initial business transaction within a *CommercialTransaction*.

#### **Tagged Values:**

*recurrence*. Specifies the number of attempts a *RequestingServiceTransaction* may be sent in response to a control exception. Control exceptions are those which were generated as a result of a control failure (e.g. TimeOut, Authentication, ect)

*isNonRepudiationOfReceiptRequired*. The *isNonRepudiationOfReceiptRequired* is derived from the *RequestingBusinessActivity(BTV)* and indicates that both partners agree to mutually verify receipt of a requesting business document and that the receipt must be non-reputable.

### *RespondingServiceTransaction*

A *RespondingServiceTransaction* is the responding business transaction within a *BusinessTransaction* to a particular *RequestingServiceTransaction*.

#### **Tagged Values:**

*isIntelligibleCheckRequired*. Both partners agree that a responding partner role must check that a requesting document is not garbled (unreadable, unintelligible) before verification of properly receipt is returned to the requesting partner.

### *ServiceCollaboration*

A *ServiceCollaboration* comprises a set of interactions (service request) between network components, which comprises one business collaboration (from BTV).

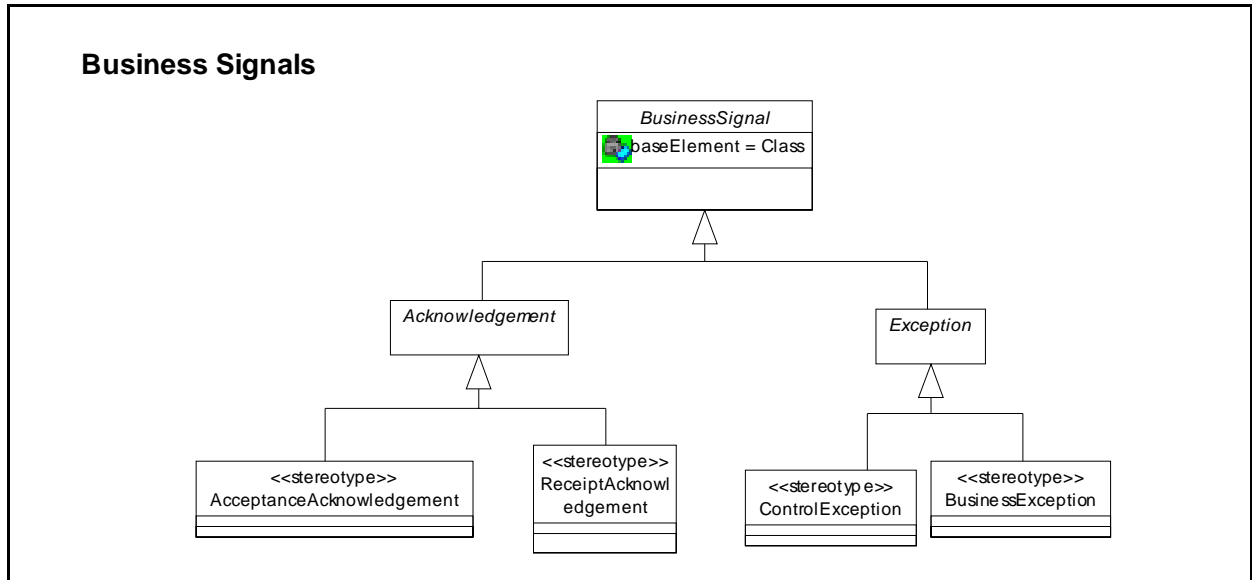
#### **Associations:**

*components*. References the *NetworkComponent* that participates in this collaboration.

*interactions*. References the *BusinessTransactions* that are exchanged between the *NetworkComponents*.



Figure 9-18 specifies the modeling elements and their interrelationships that are used to express the structure and behavior of objects in the Business Signal model. Each element and interrelationship permitted in a Business Signal is defined in the metamodel specified in this section of the document.



**Figure 9-18 BSV Abstract Syntax (Business Signals)**

#### *Acknowledgement*

An acknowledgement is an asynchronous business signal that acknowledges some aspect of a received business action message (request). The acknowledgement is sent to the service from which the business action message was received.

#### *AcceptanceAcknowledgement*

An acceptance acknowledgement business signal is returned to the initiating service if the business action message (request) content is valid with respect to the responding services business rules and the responding service is willing to perform further processing activities with this content. The initiating service must not assume that the responding service will act on a request that has not been accepted by the responding service. A trading partner agreement must agree that a receiving service has “legally” accepted a business action request (*BusinessActionMessage*) when the *BusinessActionMessage* has been “accepted” by the receiving service. At this point there is transference of legal responsibility for the fulfillment of this request by the receiving service. This signal is required if the correlating ServiceTransaction has the *timeToAcknowledgeAcceptance* attribute set to a duration greater than zero.

### *BusinessSignal*

A business signal is an object that is transmitted asynchronously back to an activity that initiated the transfer of business process execution control.

### *ControlException*

A *ControlException* signals an error condition in the management of a *ServiceTransaction* within a *ServiceCollaboration*. This signal is asynchronously returned to the initiating service that originated the request. This exception must terminate the *ServiceCollaboration*. These errors deal with the mechanisms of message exchange such as verification, validation, authentication and authorization and will occur up to message acceptance. Typically the rules and constraints applied to the message will have only dealt with structure, syntax and message element values.

### *ProcessException*

A *ProcessException* signals an error condition in a business activity. This signal is asynchronously returned to the initiating service that originated the request. This exception must terminate the *ServiceCollaboration*. These errors deal with the mechanisms that process the *ServiceTransaction* and will occur after message verification and validation. Typically the rules and constraints applied to the message will deal the semantics of message elements and the validity of the request itself and the content is not valid with respect to a responding service's business rules. This type of exception is usually generated after an *AcceptanceAcknowledgement* has been returned.

### *ReceiptAcknowledgement*

Acknowledges the receipt of a *BusinessActionMessage*. This business signal is returned by the responding service to acknowledge the receipt of a *BusinessActionMessage* if it is syntactically and structurally valid. A trading partner agreement must agree that a receiving service has "legally" received a business action request (*BusinessActionMessage*) when the *BusinessActionMessage* can be "read" by the receiving service. This signal is required if the correlating has the *timeToAcknowledgeReceipt* attribute set to a duration greater than zero.

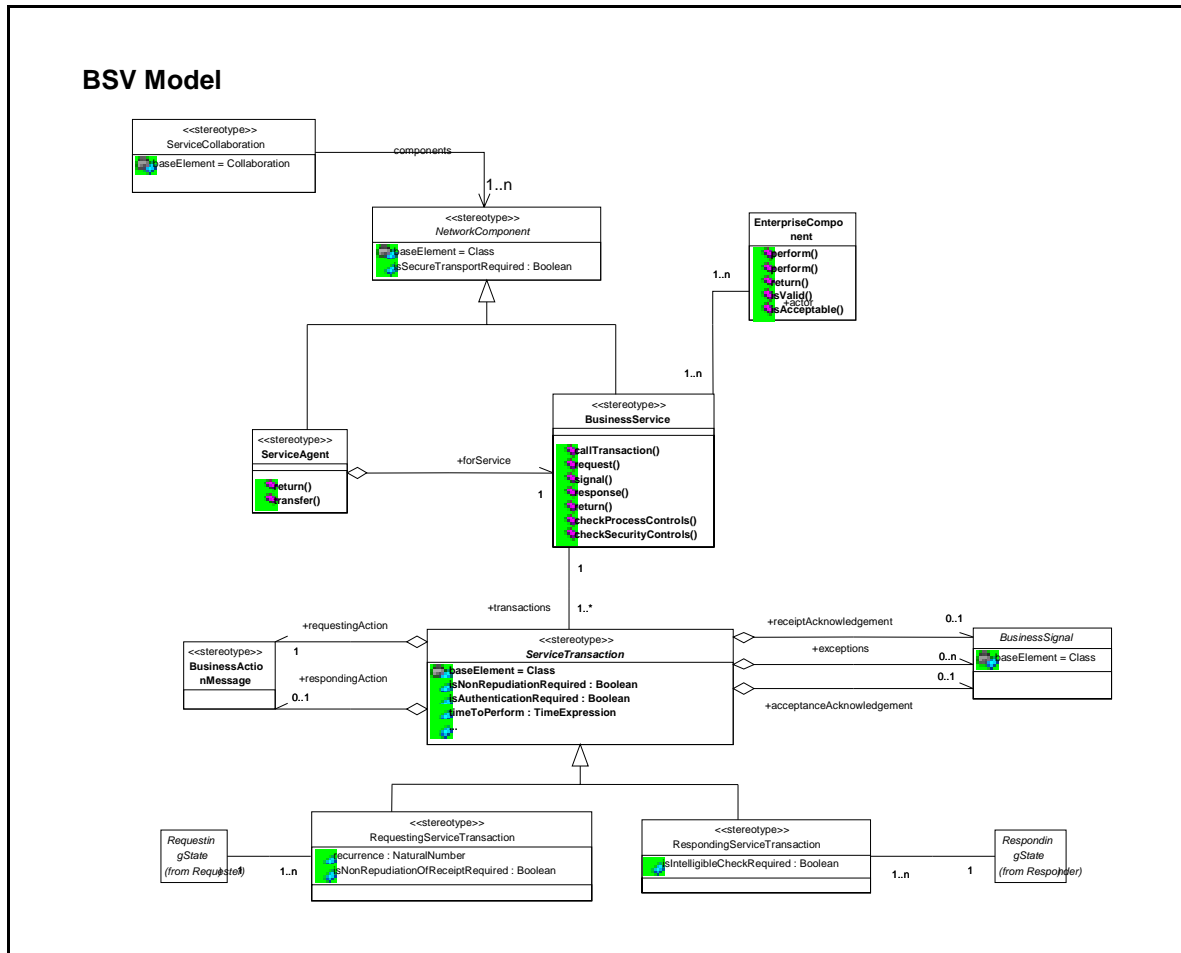
#### **9.4.1.2 Well-formedness Rules**

The following well-formedness rules apply to the BSV metamodel package.

### 9.4.2 Model Semantics

The semantics of each element of the BSV metamodel is defined in this section.

Figure 9-19 illustrates the interrelationships between the BSV modeling elements.



**Figure 9-19 BSV Model Semantics**

#### 9.4.2.1 Agent

An agent acts on behalf of a service. An agent can be a user agent such as a web browser but may also be an agent acting on behalf of another service. An agent is a network component that must implement protocols up to the agent layer of the e-business network application, communications model. An agent has no network identity as a business service component. A user agent acts as an intermediary between a business service and an employee.

#### 9.4.2.2 BusinessService

A business service is a network component that responds to business transaction requests initiated by other services. A business service

implements protocols in all of the layers of the e-business network application, communication model. Business services monitor the execution of service collaborations. A service component has network identity as a business service.

#### **9.4.2.3 ServiceTransaction**

A *ServiceTransaction* is a mutually binding interaction between an initiating service and a responding service. There may be zero or more business signals exchanged during the interaction that can be used for security, auditing and process control. A set of business transactions as defined by a *BusinessTransaction* (from BTV) is a unit of work. Both services in the *BusinessTransaction* (CT) must agree to the CT's conclusion or both sides must roll back to a state before the initial *RequestingServiceTransaction* was initiated.

A timed *ServiceTransaction* is a synchronous transaction that must complete within the specified time. An asynchronous transaction is a one-way exchange of a business action.

#### **9.4.2.4 NetworkComponent**

A network component is a logical computing component in a distributed network environment. Network transport security is specified and enabled by the network component.

#### **9.4.2.5 BusinessMessage**

A *BusinessMessage* is an information document that is exchange between business processes. The message header provides for security, signature and dictionary reference information.

#### **9.4.2.6 MessageEnvelope**

A *MessageEnvelope* is used to define routing information and privacy properties for one or more *BusinessActionMessage* that is contained within the message envelope. The *MessageEnvelope* is the highest level of containment for information that is exchange between two business processes.

#### **9.4.2.7 BusinessActionMessage**

A *BusinessActionMessage* is a specialized *StructuredMessage* used to convey *BusinessDocuments* (from BTV) between two business processes via a network component.

#### **9.4.2.8 BusinessSignalMessage**

A *BusinessSignalMessage* is a specialized *StructuredMessage* used to convey control and exception conditions between two business processes as it relates to a particular *BusinessActionMessage* request. A *BusinessSignalMessage* is transmitted asynchronously back to an business process that initiated the transfer of business process execution control.

#### 9.4.2.9 *RequestingServiceTransaction*

A *RequestingServiceTransaction* is the initial business transaction within a *CommercialTransaction*. When a *BusinessTransaction* fails, the rollback is to the state of the system and business process as it was just before the initiation of the transaction. If the recurrence property is set to a positive value the request is tried again until the count is decremented to zero. Retrys only occur on the receipt of a control exception which may an indicator that the failure could have been technical in nature. If the exception was a process exception then the recurrence counter is not applicable, since the exception was generated due to the failure of a business rule and must be redress by higher level processes.

If a *isNonRepudiationOfReceiptRequired* is true, this indicates that both partners agree to mutually verify receipt of a requesting business document and that the receipt must be non-reputable. A receiving partner must send notification of failed business control (possibly revoking a contractual offer) if a responding partner has not properly delivered their business document.

Non-repudiation of receipt provides the following audit controls.

**Verify responding role identity** (authenticate)<sup>7</sup> – Verify the identity of the responding role (individual or organization) that received the requesting business document.

**Verify content integrity** – Verify the integrity of the original content of the business document request.

#### 9.4.2.10 *RespondingServiceTransaction*

A *RespondingServiceTransaction* is the responding business transaction within a *BusinessTransaction* to a particular *RequestingServiceTransaction*. Typically all *BusinessTransaction* are defined in *RequestingServiceTransaction/RespondingServiceTransaction* pairs. If the *isIntelligibleCheckRequired* property is true then both partners agree that a responding partner role must check that a requesting document is not garbled (unreadable, unintelligible) before verification of properly receipt is returned to the requesting partner. Verification of receipt must be returned when a document is “accessible” but it is preferable to also check for garbled transmissions at the same time in a point-to-point synchronous business network where partners interact without going through an asynchronous service provider.

#### 9.4.2.11 *Service Collaboration*

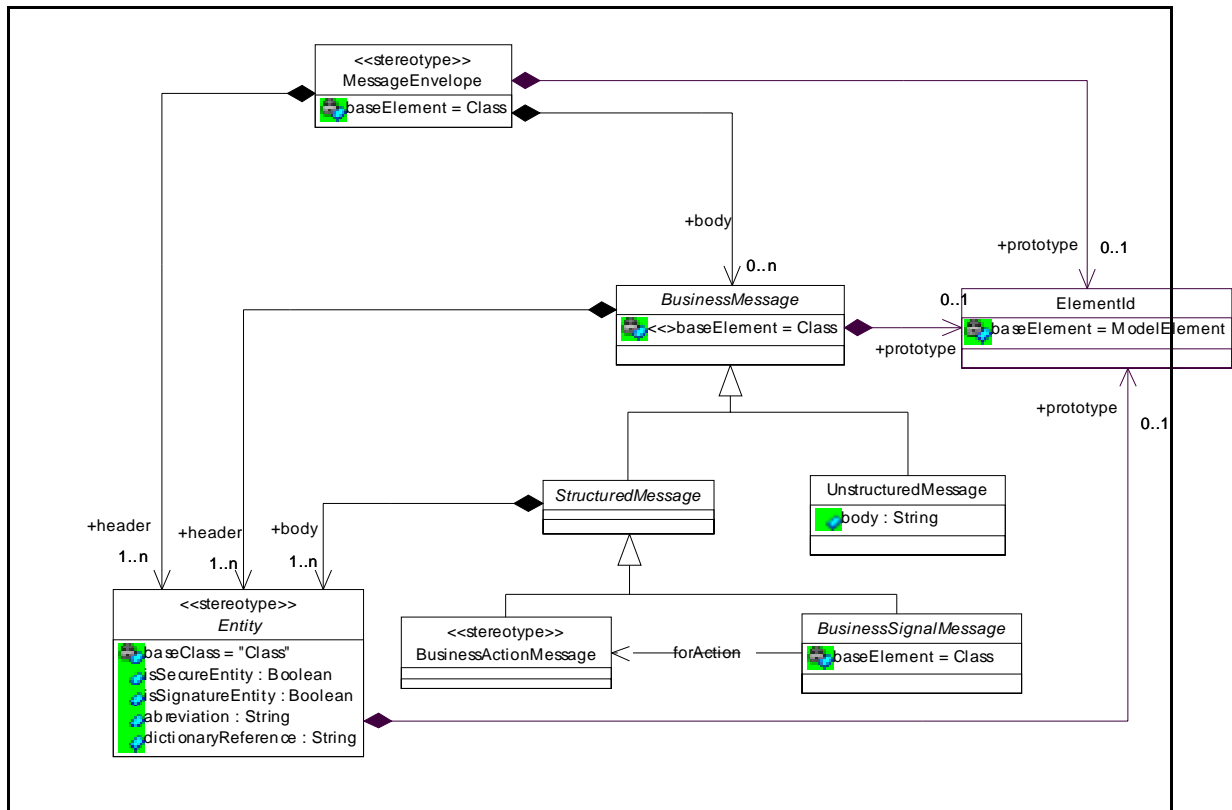
A *ServiceCollaboration* specifies the interactions between network components. It specifies the conditions and/or constraints by which interactions are executed.

### Message Model Semantics

---

<sup>7</sup> The BCF specifies digital signature for partner-to-partner non-repudiation of origin and content.

Figure 9-20 specifies the semantics for the definition of business messages.



**Figure 9-20 BSV Message Model Semantics**

#### 9.4.2.12 BusinessActionMessage

The *BusinessActionMessage* specifies the business activity that processes a business request and the header and body of the message. The *BusinessActionMessage* maps to the business document that was defined in the BTV and defines process routing and security constraints.

#### 9.4.2.13 ElementId

The *ElementId* identifies the dictionary prototype template that defines the *MessageEnvelope*, *BusinessMessage* and the *Entities* used in the construction of the message.

#### 9.4.2.14 InformationEntity

An *Entity* is the basic element for specifying information elements. Along with the name and type, it specifies privacy and security for the information.

#### 9.4.2.15 MessageEnvelope

A *MessageEnvelope* is the highest level container for transporting business documents between business processes via network components.

#### **9.4.2.16 *BusinessActionMessage***

A *BusinessActionMessage* is a specialization of a *StructuredMessage* used to invoke a business process in the receiving system.

#### **9.4.2.17 *BusinessSignalMessage***

A *BusinessSignalMessage* is a specialization of a *StructuredMessage* used to convey control and process exceptions occurring in a business process in the receiving system to a business process in the initiating system.

#### **9.4.2.18 *UnstructuredMessage***

A *UnstructuredMessage* is a specialization of a *BusinessMessage* used to transport arbitrary bit streams such as would be the case for images, video and audio.

#### **9.4.2.19 *StructuredMessage***

A *StructuredMessage* is a specialization of a *BusinessMessage* used to transport structured information.

### **9.4.3 *Model Management Abstract Syntax & Semantics***

The following stereotypes and tagged values are contained in the Business Service View management metamodel. Figure 9-21 illustrates the interrelationships between the BSV model management and model elements.

## Business Service View Model Management

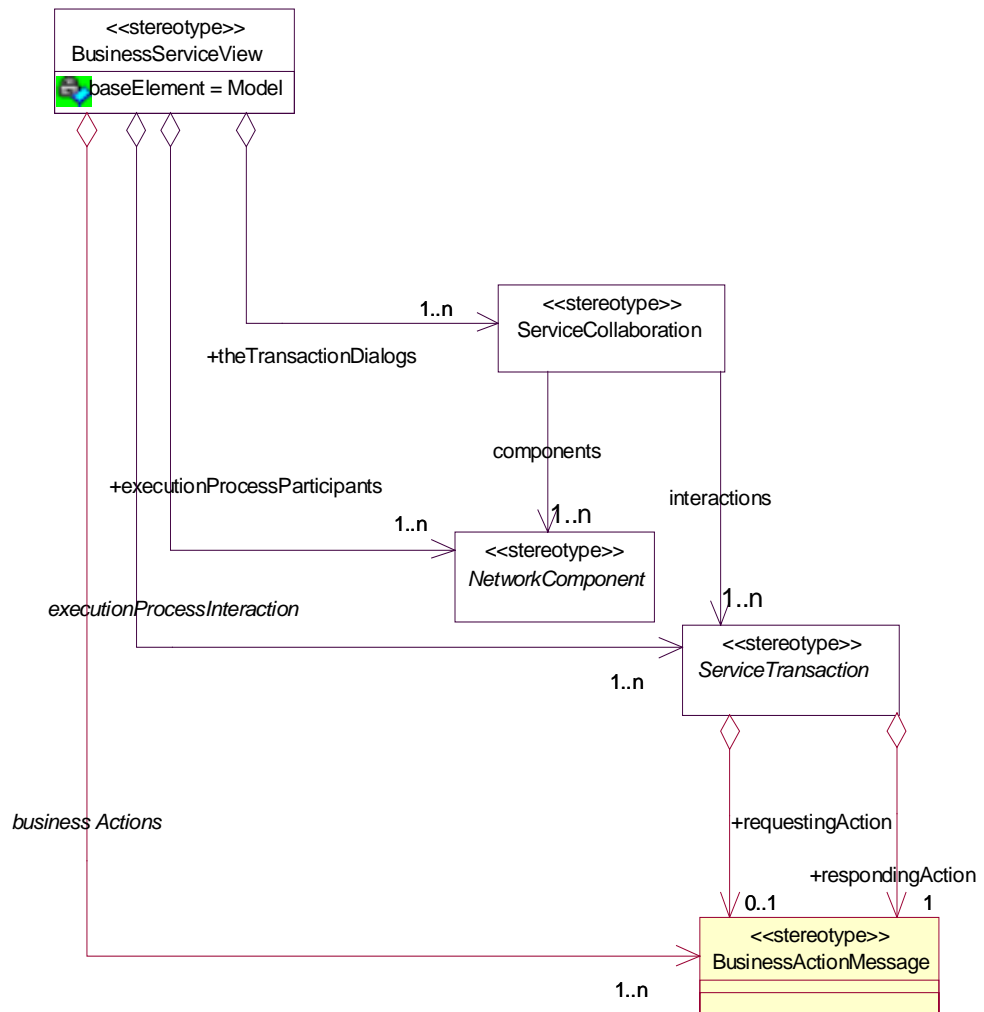


Figure 9-21 BTV Model Management Illustration



## **9.5 Business Information Structure Design Metamodel**

The e-business collaboration modeling metamodel provides a language and grammar for constructing business collaboration models. Business information structure design patterns are applications of the metamodel to common business object representations. Representations capture common structure and semantics applicable to specific business object domains.

This document describes the following design patterns.

1. Reference design pattern. The design pattern for referencing business information descriptions to describe aggregate business information containers.
2. Query/Response business document design pattern. The design pattern for both querying business information and for specifying the structure of the response.
3. Disjunction design pattern. The design pattern for representing business information entities that contain one or more of a disjunctive entity.
4. Reification design pattern. The design pattern for representing common business information entities.
5. UML/XML DTD translation design pattern. The design pattern for translating UML business document models into XML DTD document schema.
6. Business document design pattern. The design pattern for exchanging messages that can be interpreted as “legal writings” with respect to commercial law.
7. Request/Response business document design pattern. The design pattern for requesting complex query results and for specifying the structure of the response.

### **9.5.1 Business Information Model Abstract Syntax**

#### **9.5.1.1 Stereotypes and Tagged Values**

Figure 9-22 specifies the modeling elements and their interrelationships that are used to express the structure of business objects and documents in the BSV of a Business Transaction and Business Collaboration Protocol model. Each element and interrelationship permitted in a FSV Information Model is defined in the metamodel

specified in this section of the document.

### Information Model Abstract Syntax

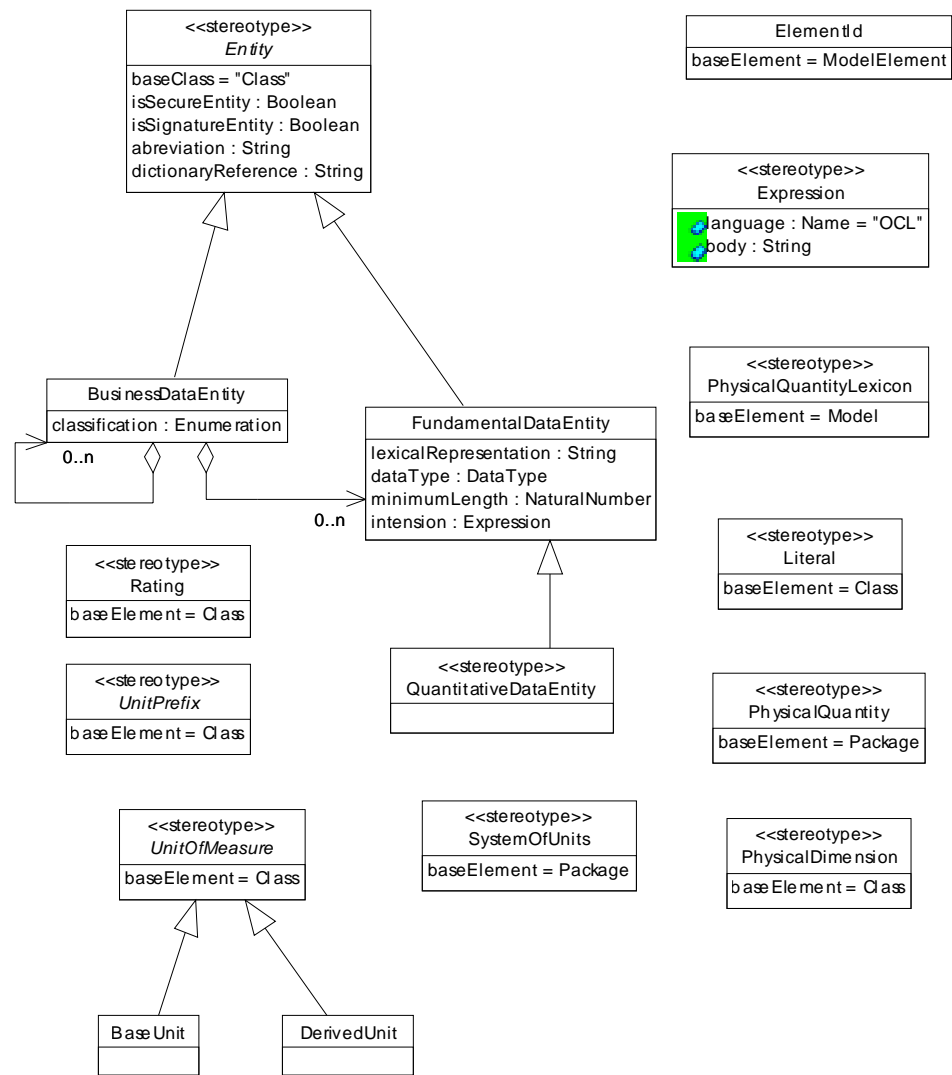


Figure 9-22 BSV Abstract Syntax

#### InformationEntity

An InformationEntity is the basic element used for modeling hierarchical information structures.

#### Tagged Values:

isConfidential. The information entity is encrypted so that unauthorized parties cannot view the information.

*isTamperProof*. The information entity has an encrypted message digest that can be used to check if the message has been tampered with. This requires a digital signature (sender's digital certificate and encrypted message digest) associated with the document entity.

*isAuthenticated*. There is a digital certificate associated with the document entity. This provides proof of the signer's identity.

#### *FundamentalDataEntity*

An *FundamentalDataEntity* is an atomic element used for modeling hierarchical information structures.

##### **Tagged Values:**

*lexicalRepresentative*. Defines the lexical representation of the element.

*dataType*. Defines the data type.

*minimumLength*. Defines the minimal length that this element.

*intention*. A OCL expression used to define the intended use of this element.

#### *ElementId*

The *ElementID* is used to provide a unique identification for a particular information element.

##### **Tagged Values:**

#### *Expression*

An *Expression* provides for the definition of context and business rules using OCL .

##### **Tagged Values:**

*language*. Defines the formal language used to define the expression.

*body*. Defines the business rules.

#### *StructuredMessage*

An *InformationEntity* is the basic element used for modeling hierarchical information structures.

##### **Tagged Values:**

*language*. Defines the formal language used to define the expression.

*body*. Defines the business rules.

### **9.5.2 *Model Semantic***

The semantics of each element of the Information Model metamodel is defined in this section.

## Directory Model Semantics

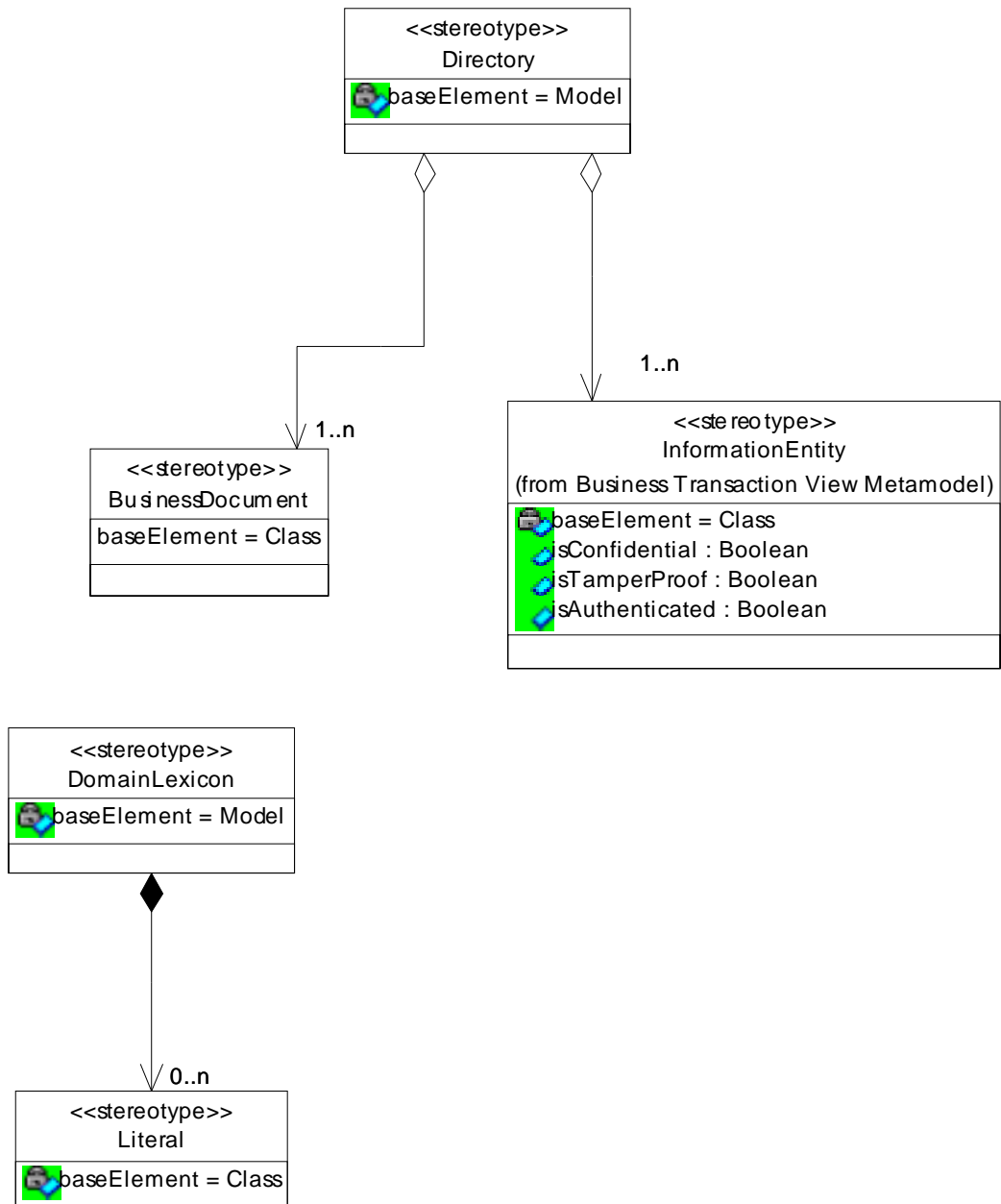


Figure 9-23 Directory Model Semantics

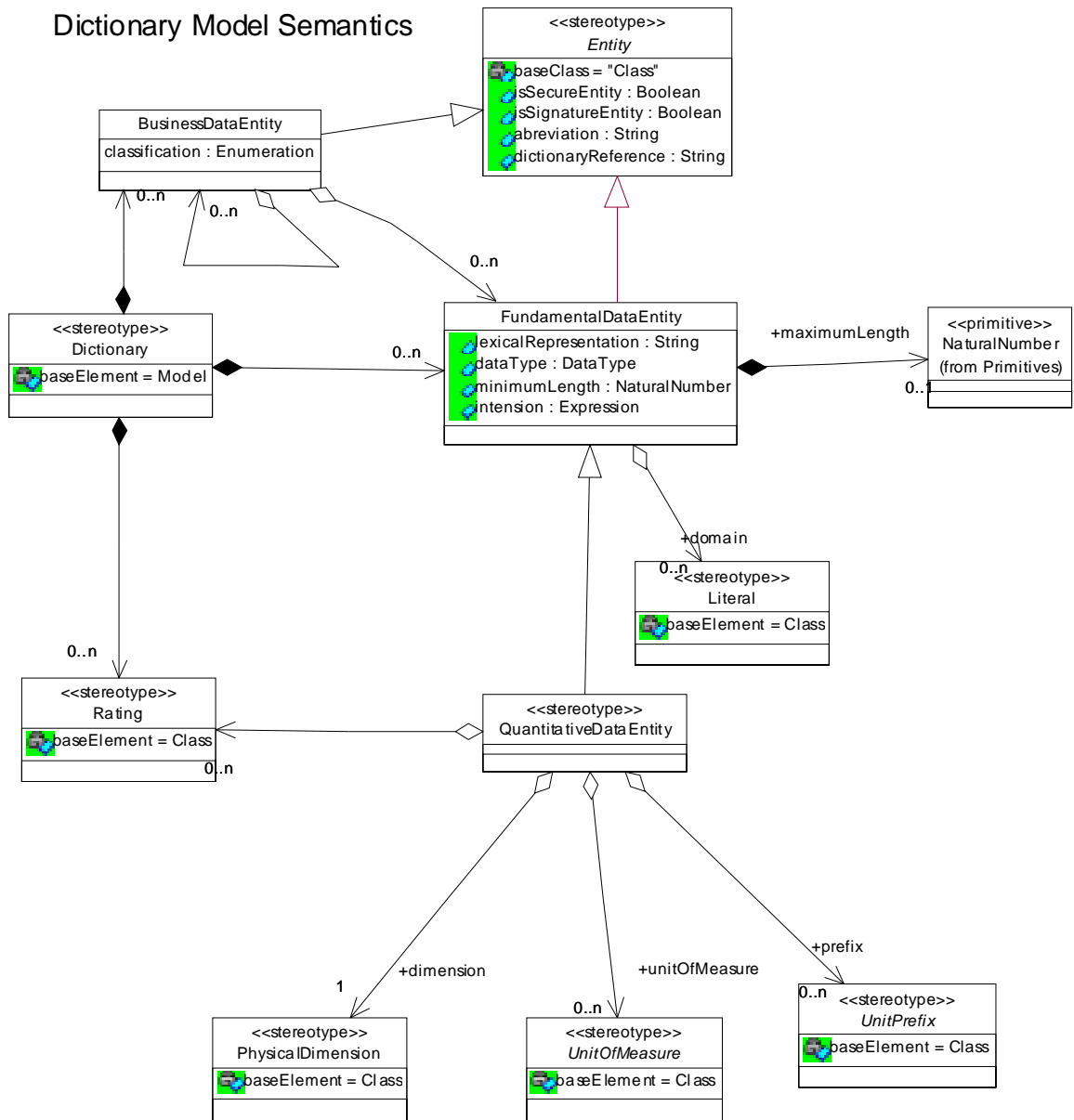


Figure 9-24 Dictionary Model Semantics

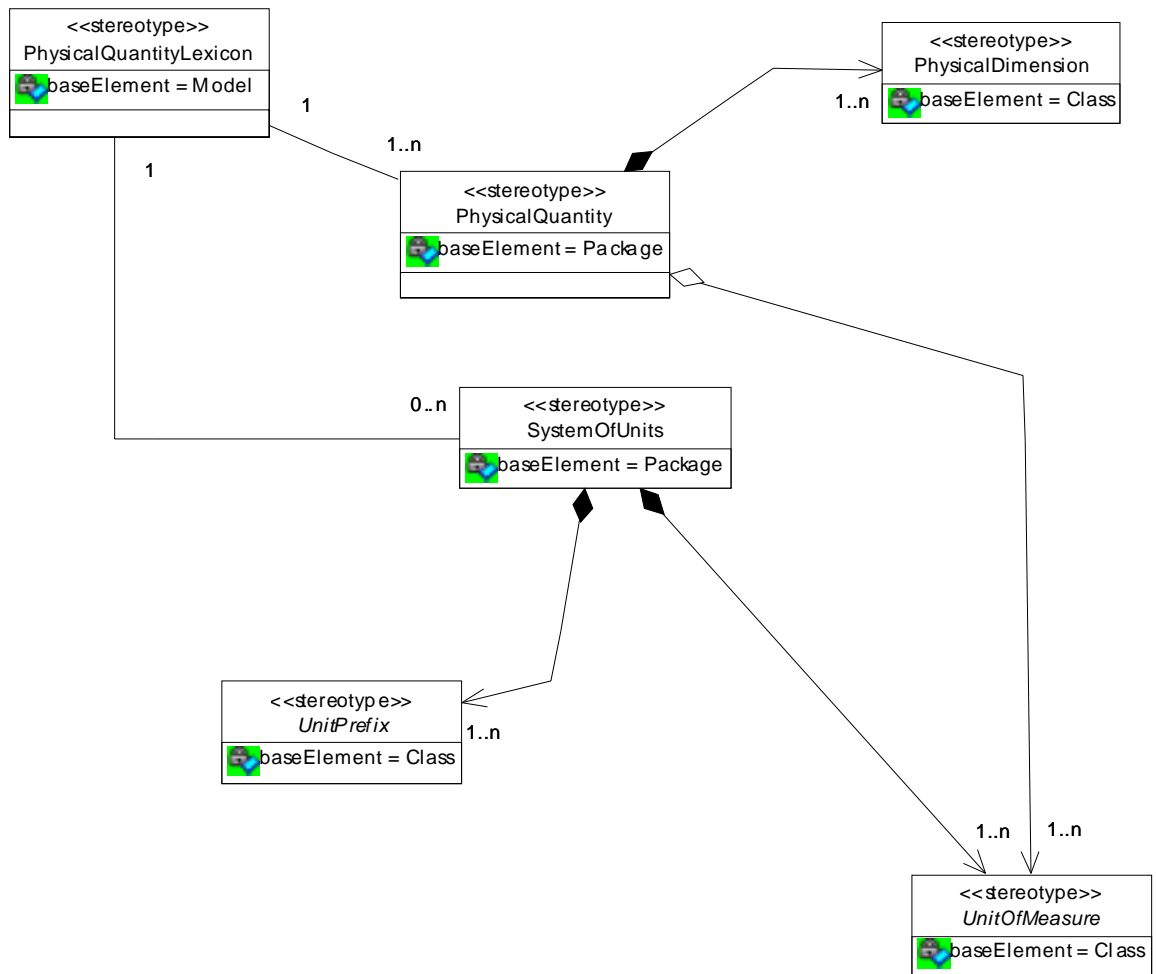


Figure 9-25 Physical Quantity Lexicon

## Units of Measure Semantics

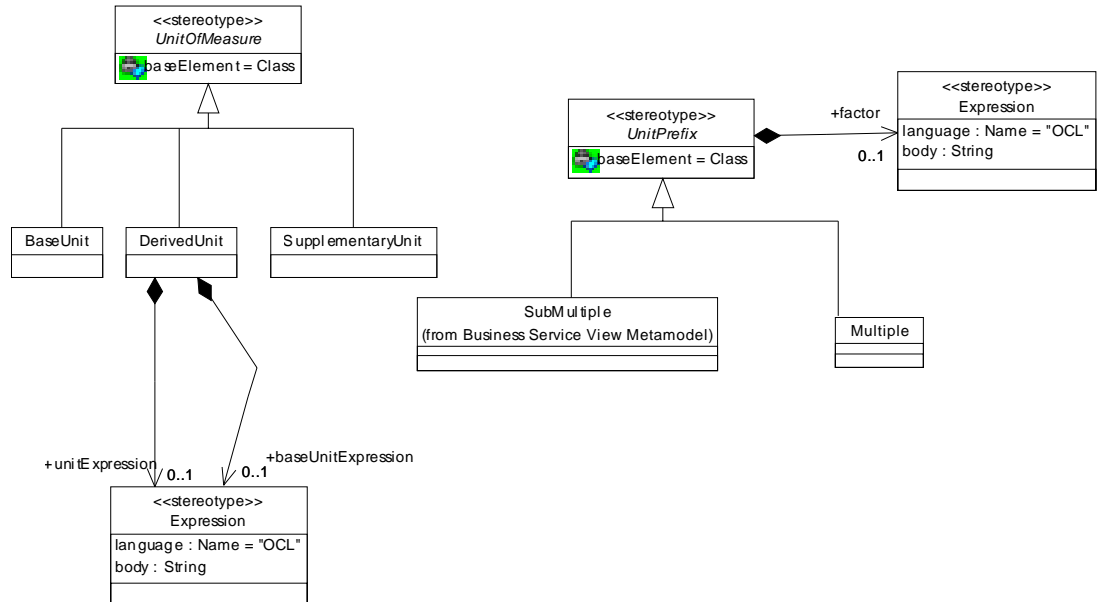


Figure 9-26 Units of Measure Semantics