UN/CEFACT United Nations Centre for Trade Facilitation and Electronic Business

Techniques and Methodologies Group (TMG)

UML Profile for UN/CEFACT's Modeling Methodology (UMM)

Base Module

Candidate for 2.0

Draft for IMPLEMENTATION VERIFICATION

2010-01-29

Table of Contents

Table of Contents 2
1 About this Document 3
1.1 Status of this Document 3
1.2 Revision History 3
1.3 Document Context 3
1.4 Conventions 4
2 Project Team 6
2.1 Disclaimer 6
2.2 Contact 6
2.3 Project Team Participants 6
3 Introduction 7
3.1 Audience 7
3.2 Related Documents 7
3.3 UN/CEFACT's Modeling Methodology (UMM): Overview 8
3.4 Objectives 9
3.4.1 Goals of the Technical Specification 9
3.4.2 Requirements 9
3.4.3 Caveats and Assumptions 10
3.5 Structure of the UMM Base Module 10
4 UMM Base Module 11
4.1 Abbreviations of Stereotypes 11
4.2 Stereotypes and Tag Definitions (normative) 11
Copyright Statement 14

1 About this Document

1.1 Status of this Document

This document is currently being developed, and is at Open Development Process Step 6: "Implementation Verification"

(i) Activities

The UNECE secretariat provides links on the UNECE website to the implementation. The FMG notifies Heads of Delegation and various e-mail distribution list subscribers that the Implementation Draft is available for implementation verification and provides them with details regarding the process for submitting comments. The project team processes comments and posts updated Implementation Drafts and comment logs to the PG website or UNECE website (through the secretariat). The comment/update/posting cycle continues until at least two independent implementations have been confirmed and the PG approves a project team recommendation to conclude ODP6. While the criteria, evaluation, and ultimate decision to conclude ODP6 is left to the PG, the PG must ensure that the project team has met all comment processing requirements

(ii) Artifacts

Typical artifacts produced by ODP6 include:

a. Final Draft
b. Updated Comment Log

The second review cycle of ODP 6 is open until 26th April 2010. Please send your comments to the Project Lead Christian Huemer (huemer@big.tuwien.ac.at)

1.2 Revision History

Version	Release	Date	Comment
Candidate for 2.0	Internal Draft	2008-04-11	
Candidate for 2.0	Public Draft	2008-06-27	
Candidate for 2.0	Implementation verification	2010-01-25	Changes from both previous drafts are marked in green

1.3 Document Context

The UMM meta model is divided into a set of meta modules. This means the UMM meta model is partitioned into functional levels, ranging from core, minimal functionality to complete functionality. The following partition levels have been defined for meta modules:

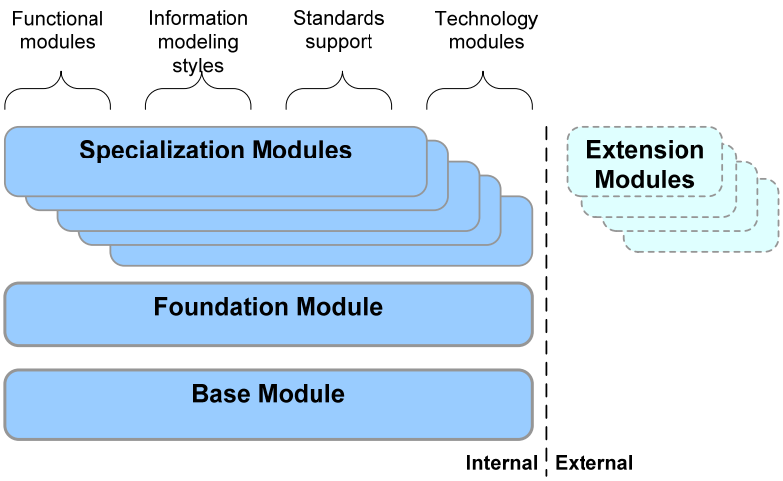

Figure 1 Module structure of the UMM meta model

Base: Covers the fundamental principles that are shared across all of the other modules.

Foundation: Includes the core concepts of the UMM. In addition, it defines all of the concepts that are used as part of the minimal methodology to produce a UMM compliant business collaboration model. Furthermore, it provides fundamental principles which are shared across all of other modules.

Specialization: Multiple specialization modules might define add-on concepts to the foundation module. Each specialization module addresses a specialized type of analysis that extends the foundation module at a well-defined extension point for a specific topic. Specialization modules might become candidates for later inclusion into the foundation module.

Extension: Extension modules serve the same purpose as specialization modules. Whereas specialization modules are developed and maintained by UN/CEFACT, extension modules are adding features that are created and maintained by organization(s) which are external to UN/CEFACT.

This specification defines the base module of UMM 2.0.

1.4 Conventions

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

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

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

2 Project Team

2.1 Disclaimer

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

2.2 Contact

Name:	Christian Huemer
Company:	Vienna University of Technology
Street:	Favoritenstraße 9-11/188-3
City, state, zip/other:	1040 Vienna
Nation:	Austria
Phone:	+43 1 58801 18882
Email:	huemer@big.tuwien.ac.at

2.3 Project Team Participants

Project Team Lead:	Christian Huemer	Austria
Editing Team:	Jens Dietrich	Germany
	Birgit Hofreiter	Austria
	Christian Huemer	Austria
	Philipp Liegl	Austria
	Glenn Miller	Canada
	Harry Moyer	Australia
	Rainer Schuster	Austria
	Marco Zapletal	Austria
Contributors:	Steve Capell	Australia
	Sylvie Colas	France
	Barbara Flügge	Switzerland
	William McCarthy	USA
	Thomas Motal	Austria
	Christian Senf	Germany
	Nita Sharma	USA
	Gunther Stuhec	Germany

The Editing Team of this UMM base module likes to thank former members of TMG's Business Process Working Group (BPWG) who have spent enormous efforts in putting the UMM into a stage that we were able to build upon in order to create this base module.

3 Introduction

3.1 Audience

A reader of the document must have a deep understanding of UML 2.1.2. She or he must be able to understand meta models denoted as UML class diagrams. She or he should be familiar with the UML 2.1.2. meta model, at least she or he must be able to check back the UML 2.1.2. meta model. The reader should be familiar with OCL 2.0 in order to understand the OCL constraints of this UMM profile – those who are not familiar with OCL are provided with a plain text description of the constraint.

The information described in this manual is aimed at

- advanced business process modelers that verify a UML model for UMM compliance (if not supported by a tool)
- advanced business process modelers who train other business process modelers and business process analysts
- software designers who want to produce UML tools providing support for this UMM foundation module
- software designers who want to produce tools to transform UMM compliant business collaboration models into specifications of the IT-layer (ebXML, Web Services, UN/EDIFACT, etc.).
- software designers who want to produce repositories to register UMM compliant business collaboration models

3.2 Related Documents

- **UN/CEFACT**
  - UN/CEFACT Open Development Process (TRADE/R.650/Rev.4/Add.1/Rev.1 - 19 April 2007)
    http://www.unece.org/cefact/cf_plenary/plenary07/trd_R650_Rev4_A1E.pdf
  - UPCC 1.0 - UML Profile for Core Components
    http://unstandards.org:8080/display/public/UPCC+-+UML+Profile+for+Core+Components
  - Core Component Technical Specification
    http://www.unece.org/cefact/ebxml/CCTS_V2-01_Final.pdf
  - Core Component Message Assembly
    http://www.unstandards.org:8080/display/public/CCMA+-+ODP+3+-+1st+Working+Draft
- **International Organization for Standardization (ISO)**
  - Open-edi Reference Model. ISO/IEC 14662
    http://standards.iso.org/ittf/PubliclyAvailableStandards/c037354_ISO_IEC_14662_2004(E).zip
- **Object Management Group (OMG)**
  - Unified Modeling Language Specification (UML), Version 2.1.2
    http://www.omg.org/docs/formal/07-02-05.pdf

3.3 UN/CEFACT's Modeling Methodology (UMM): Overview

UN/CEFACT's Modeling Methodology (UMM) is a UML modeling approach to design the business services that each partner must provide in order to collaborate. It provides the business justification for the services to be implemented in a service-oriented collaboration architecture. Thus, a primary vision of UN/CEFACT is to capture the business knowledge that enables the development of low cost software based on service-oriented architectures (SOA) helping the small and medium size companies (SMEs), as well as emerging economies, to engage in e-Business practices. UMM focuses on developing a global choreography of inter-organizational business processes and their information exchanges. UMM models are notated in UML syntax and are platform independent models. The platform independent UMM models identify which services need to be realized in a service-oriented architecture, in order to implement the business collaboration. This approach provides insurance against technical obsolescence.

The UMM, as described in this document, is the formal description technique for describing any Open-edi scenario as defined in ISO/IEC 14662 "Open-edi reference model". An Open-edi scenario is a formal means to specify a class of business transactions having the same business goal, such as, purchasing, or inventory management. The primary scope of UMM is the Business Operations View (BOV) and not the Functional Service View (FSV) as defined in ISO/IEC IS 14662. The BOV is defined as "a perspective of business transactions limited to those aspects regarding the making of business decisions and commitments among organizations", while the FSV is focused on implementation specific, technological aspects of Open-edi. The commitments of the BOV layer are reflected in the choreography of the inter-organizational business processes and their information exchanges. At the FSV layer, this choreography must be implemented by a set of composite services. Therefore it follows, that UMM, which targets the BOV layer, defines what the business is about; and the technologies on the FSV layer define how to implement the business by a service-oriented architecture.

This version of the UMM consists of three views each covering a set of well defined artifacts:

- Business Requirements View (bRequirementsV)
- Business Choreography View (bChoreographyV)
- Business Information View (bInformationV)

Business Requirements View (bRequirementsV): The Business Requirements View is used to gather existing knowledge. It identifies the business processes in the domain and the business problems that are important to stakeholders. It is important at this stage that business processes are not constructed, but discovered. Stakeholders might describe intra-organizational as well as inter-organizational business processes. All of this takes place in the language of the business experts and stakeholders. The business requirements view results in a categorization of the business domain (manifested as a hierarchical structure of packages) and a set of relevant business processes (manifested as use cases). The result may be depicted in use case diagrams. In order to model the dynamics of each business process, one may use a Business Process Activity Model, or a Sequence Diagram, which would be placed beneath the Business Process Use Case. As a practical note, the Business Process Activity Model may depict a process or processes which involve one or more Business Partners. A Sequence Diagram will depict information exchanges between two or more Business Partners. The Business Partners are described within their own package (Business Partner View). A Business Process Activity Model may show state changes to Business Entities. Business Entities are "real-word" things having business significance and are shared among the business partners involved in the collaboration. The Business

Entities and their lifecycles of state changes are modeled in the Business Entity View. Furthermore, the Business Entity View also contains one or more packages which represent the conceptual data structures of the Business Entities.

Business Choreography View (bChoreographyV): The Business Choreography View is used to define and document the global choreography between collaborating business partners in an inter-organizational business process. Within the Business Choreography View, the Business Transaction View contains and documents the requirements of Business Transaction Use Cases, and their participating Authorized Roles. The dynamics of a Business Transaction Use Case are described by a Business Transaction. A business transaction defines a simple choreography of exchanging business information between two authorized roles and an optional response. A business transaction identifies the business actions of each partner responsible for sending and receiving the business information. These actions correspond to the requirements of any solution that must be implemented on each business partner's side in a service-oriented collaboration architecture. Within the Business Choreography View, the Business Collaboration View contains and documents the requirements of Business Collaboration Use Cases and their participating Authorized Roles. The dynamics of a Business Collaboration Use Case are described by a Business Collaboration Protocol. A Business Collaboration Protocol choreographs the flow among business transactions, and/or nested Business Collaboration Protocols. This flow depends on the states of business entities. When a Business Collaboration Use Case is identified, but different sets of parties may execute this collaboration, the different Realizations (executions) may be modeled within the Business Realization View, as a Business Realization Use Cases.

Business Information View (bInformationV): An execution of a business transaction usually results in the change of state of one or more business entities. Thus, the information exchanged in a transaction should be limited to the minimum information needed to change the state of a business entity. Nevertheless, UMM allows the definition of an information exchange in a document-centric approach – even if this is not recommended. A Business Information View contains Business Information Artifacts. UMM does not mandate a specific Business Information Modeling approach. However, UMM strongly recommends that Business Information is modeled in accordance to UN/CEFACT's Core Components Technical Specification and Message Assembly Guidelines. In order to model Core Components by means of UML, UN/CEFACT provides the Profile for Core Components (UPCC).

3.4 Objectives

3.4.1 Goals of the Technical Specification

The goals of this specification are:

- To define a set of data types that may be shared between the UMM Foundation module and different UMM Specialization modules
- To define the fundament on which constitutive UMM specifications may based upon.

3.4.2 Requirements

This specification is guided by the following key requirements derived from the above goals:

- The UMM Base module contains only stereotypes that are currently used in the UMM Foundation module or in a UMM specialization and extension modules
- Today, the UML is the most commonly supported modeling language by modeling tools. In order to use the broad range of tools, a UMM model must be a special kind of UML model. Thus, the UMM

base module is based on the UML meta model. In fact, it provides a UML Profile consisting of stereotypes, tag definitions and constraints.

3.4.3 Caveats and Assumptions

This specification makes the following assumptions:

- This UML profile is based on the UML meta-model version 2.1.2. This version is the current OMG version. Using another UML meta-model as a basis for the development of a UMM compliant business collaboration model may not deliver correct results.
- The basic concepts of the UMM and the way they relate to each other are described and explained by means of a meta model (to be found in the non-normative "conceptual description" sections of this document).

3.5 Structure of the UMM Base Module

The UMM base module provides common data types, which may be used by the UMM foundation module or by other specialization and extension modules. This version of the base module consists of three artifacts named "*bInformation*" (BusinessInformation), "*InfEnvelope*" (InformationEnvelope) and "*bLibrary*" (Business Library).

A *business library* is realized as a package. Elements which inherit from a *business library* (or subtypes of it), are candidates for registration in a registry. A *business library* therefore acts as container for elements, which should be registered and retrieved together to be semantically complete.

BusinessInformation is realized as a class and represents the abstract concept of a business document exchanged in a business transaction between two business partners. In order to allow for an arbitrary *business information* to be exchanged in a UMM business transaction, the UML class based business information representation must inherit from a *BusinessInformation* or subtypes thereof.

An InformationEnvelope is a subtype of a *BusinessInformation* and represents a concrete business message which is exchanged in a UMM business transaction.

4 UMM Base Module

4.1 Abbreviations of Stereotypes

Stereotype Abbreviation	Full Stereotype Name
bInformation	BusinessInformation
bLibrary	BusinessLibrary
InfEnvelope	InformationEnvelope

4.2 Stereotypes and Tag Definitions (normative)

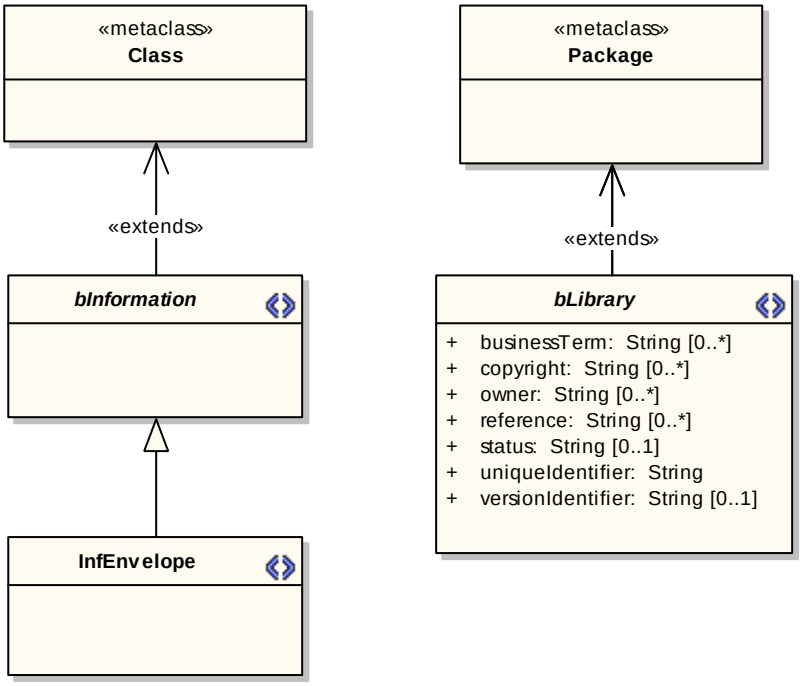

Figure 2 UMM Base Module – Abstract Syntax

Stereotype	**bLibrary (BusinessLibrary)**
Base Class	Package
Parent	-
Description	A business library is a container for objects, which together build a semantic unit.
Tag Definition	**businessTerm**
	Type: String
	Multiplicity: 0..*
	Description: A business term is a synonym, by which a business entity is commonly known.

copyright	
Type	String
Multiplicity	0..*
Description	Holds information about the copyright of a business library.

owner	
Type	String
Multiplicity	0..*
Description	The owner of the business library, who might be an organization, an institution or an individual.

reference	
Type	String
Multiplicity	0..*
Description	Identifies references to additional resources, where continuative information about the business library could be found.

status	
Type	String
Multiplicity	0..1
Description	An indicator for the current lifecycle status of an object if the object is registered in a registry. If so, the status must be set by the registry.

uniqueIdentifier	
Type	String
Multiplicity	1
Description	A unique identifier uniquely represents a business library if it is stored in a registry. There are no specific rules for the structure of the identifier; however the preferred identification scheme is the ITU-T Rec. X.667|ISO/IEC9834-8 Universally Unique Identifier (UUID) scheme. Implementers are free to use this scheme, or choose any other structure scheme, providing it guarantees uniqueness within the library to which it belongs.

versionIdentifier	
Type	String
Multiplicity	1
Description	A unique identifier representing the version of a business library if it is stored in a registry. There are no specific rules for the structure of the identifier; however the preferred identification scheme is the ITU-T Rec. X.667|ISO/IEC9834-8 Universally

			Unique Identifier (UUID) scheme. Implementers are free to use this scheme, or choose any other structure scheme, providing it guarantees uniqueness within the library to which it belongs.

Stereotype	bInformation (BusinessInformation)
Base Class	Class
Parent	-
Description	A *BusinessInformation* realizes abstract business document information that is exchanged between authorized roles performing activities in a business transaction. Since a *BusinessInformation* is defined as abstract it cannot be used directly in order to set the type of exchanged information in a *BusinessInformation*. Instead the concept of an *InformationEnvelope* is used.

Stereotype	InfEnvelope (InformationEnvelope)
Base Class	Class
Parent	-
Description	An *InformationEnvelope* is a subtype of a *BusinessInformation* and represents a concrete business message which is exchanged in a UMM business transaction. Any business document artifacts are connected to an *InformationEnvelope* using associations.

Copyright Statement

Copyright © UN/CEFACT 2007. All Rights Reserved.

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

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

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