Collaborative Network for Industry, Manufacturing, Business and Logistics in Europe

The EU Framework Programme for Research & Innovation 2014-2020

HORIZON 2020

D2.2

Semantic Modelling of Manufacturing Collaboration Assets

Project Acronym	NIMBLE	
Project Title	Collaboration Network for Industry, Manufacturing, Business and Logistics in Europe	
Project Number	723810	
Work Package	WP2 Platform Technology Specification	
Lead Beneficiary	University of Bremen (UB)	
Editor/Lead Author	Quan Deng	UB
Reviewer	Wernher Behrendt	SRFG
Reviewer	Suat Gönül	SRDC
Contributors		ALL
Dissemination Level	PU	
Contractual Delivery Date	30/09/2017	
Actual Delivery Date	02/11/2017	
Version	V1.0	

Abstract

The NIMBLE project performs research leading to the development of a cloud and IoT platform specifically targeted at supply chain relationships and logistics. Core capabilities will enable firms to register, publish machine-readable catalogues for products and services, search for suitable supply chain partners, negotiate contracts and supply logistics, and develop private and secure information exchange channels between firms. The intention is to support a federation of such platform instances, all providing a set of core services, and each potentially specifically tailored to a different aspect (regional, sectorial, topical, etc.).

In order to facilitate the discovery of suitable supply chain partners and enable business transactions, the manufacturing collaboration assets of companies need to be digitally represented in the NIMBLE platform. The main goal of this document is to explain in detail the semantic modelling of the manufacturing collaboration assets for the representation of various types of tangible and intangible resources.

This deliverable is the output of Task 2.2 and describes the semantic modelling of the manufacturing collaboration assets. It analyses the use case scenarios and derives from the scenarios the requirements for semantic modelling. Based on the requirements, we have summarized two core ontology modules in the semantic modelling: the catalogue ontology and the business process ontology. Subsequently, various ontological as well as non-ontological resources are reviewed for the purpose of reuse. Afterwards, the modules in the semantic modelling are described one by one. For each core module, the main concepts are specified, and the respective ontology schema is described in detail. With the semantic modelling, resources from different partners in various formats can be semantically annotated and published on the NIMBLE platform to facilitate business-to-business transactions. At present, the focus is on the domains of the use cases, but the semantic models can be extended or tailored to other domains or use cases. This deliverable also describes the Semantic Metadata Repository (SMDR) that is used to maintain all relevant data elements and their links.

Document History

Version	Date	Comments
V0.1	02/08/2017	Initial version and assignments distribution
V0.13	04/09/2017	First draft for internal review
V0.13_hlx	11/09/2017	Input from HLX related to product avatar
V0.13-rev_wb	14/09/2017	Revisions by Wernher Behrendt (SRFG)
V0.13-rev_qd	15/09/2017	Updates based on feedbacks from SRDC, SRFG, AID and ENEA
V0.14	18/09/2017	Input from AID related to furniture taxonomy
V0.15	18/09/2017	Input from SRFG related to eClass and metadata repository
V0.15-rev_wb	21/09/2017	Updates and Review by Wernher Behrendt (SRFG)
V0.16	22/09/2017	Input from ENEA related to Moda-ML
V0.17	25/09/2017	Input from SRDC related to UBL
V0.22	28/09/2017	Updates based on internal Review to build a complete pre-final version for final review
V0.23	12/10/2017	QA review by WB – returned to QD to address minor QA issues – then ready for submission.
V1.0	16/10/2017	Final version ready for submission
V1.0	02/11/2017	Final check and submission (SRFG/WB).

Table of Contents

List of Figures

List of Tables

Acronyms

Table 1: Acronyms

Acronym	Meaning
ABIE	Aggregate Business Information Entities
AI	Artificial Intelligence
B2B	Business to Business
BBIE	Basic Business Information Entity
BFO	Basic Formal Ontology
BOL	Beginning-of-Life
BPEL	Business Process Execution Language
BPM	Business Process Model
BPMN	Business Process Model and Notation
CCTS	Core Component Technical Specification
CEN	European Committee for Standardization
CSV	Comma separated value
DOLCE	a Descriptive Ontology for Linguistic and Cognitive Engineering
ebXML	Electronic Business XML
EOL	End-of-Life
IEC	International Electrotechnical Commission
IOT	Internet of Things
ISO	International Organization for Standardization
LDCache	Linked Data Caching
LDP	Linked Data Platform
LDPath	Linked Data Path
Moda-ML	Middleware tools and documents to enhance the textile/clothing supply chain through XML

MOL	Middle-of-Life
NIMBLE	Collaboration Network for Industry, Manufacturing, Business and Logistics in Europe
OWL	Web Ontology Language
RDBMS	Relational Database Management System
RDF	Resource Description Framework
RDFS	Resource Description Framework Schema
RFID	Radio-Frequency Identification
SKOS	Simple Knowledge Organization System
SMDR	Semantic Metadata Repository
SPARQL	SPARQL Protocol and RDF Query Language
SQL	Structured Query Language
TBD	To Be Defined
UBL	Universal Business Language
UN/CEFACT	United Nations Center for Trade Facilitation and Electronic Business
XML	Extensible Markup Language
XPath	XML Path Language
XSD	XML Schema Definition

1 Introduction and scope of this deliverable

NIMBLE aims at creating a B2B platform that can be utilised by smaller manufacturing businesses and suppliers. Within the platform, manufacturing businesses and suppliers interact with each other to perform business transactions.

In order to facilitate the discovery of suitable supply chain partners and enable business transactions, the manufacturing collaboration assets of companies need to be digitally represented in the NIMBLE platform. With this digital representation, suitable supply chain partners can be discovered and business processes can be initiated and executed. For the support of partner discovery, the platform needs to enable companies to introduce themselves to the NIMBLE platform with the resources, products and services they provide. After the discovery of partners, business processes are initialized and executed. As different documents are exchanged between partners in business processes, metadata of exchanged documents also have to be specified to guarantee that the right document is exchanged and to ensure that the exchanged document is understandable by different actors.

The NIMBLE platform uses semantic modelling to represent various types of resources and business processes. Since resources in different domains have their own individual features and these resources may have been modelled in domain-specific ontologies, the NIMBLE semantic modelling offers the capability of continuous extension through semantic mediation.

1.1 Purpose of the deliverable

This deliverable is dedicated to Task 2.2 of the NIMBLE project and describes the semantic modelling of manufacturing collaboration assets. It analyses the use case scenarios and derives from the scenarios the requirements for semantic modelling. Based on the requirements, various ontological as well as non-ontological resources are reviewed for the purpose of reuse. Afterwards, it summarizes main concepts in core data models and describes the data models for the representation of various types of tangible and intangible resources. With the data models, resources from different partners in various formats can be semantically annotated and published in order for other partners to discover them using semantic queries. At present, the focus is on the domains of the use cases, but the semantic models are easily extended or tailored to other domains or use cases. This deliverable contains also the description of the Semantic Metadata Repository (SMDR) that is used to maintain all relevant data elements and their links.

1.2 Approach

The approach to achieve the objective of Task 2.2 is adapted from well-recognized ontology modelling methodologies including METHONTOLOGY (Fernández et al. 1997) and NeOn (Suárez-Figueroa et al. 2008). METHONTOLOGY is a very mature methodology for building ontologies mainly from scratch, with support of reuse and re-engineering of other ontologies. Development-oriented activities in the methodology include specification, conceptualization, integration, implementation and evaluation. In the *Specification* phase, the main purpose and scope of the ontology are stated. The *Conceptualization* phase structures the domain knowledge into conceptual models including the domain vocabulary identified in the ontology specification activity. With the goal of speeding up the construction, *Integration* considers the reuse of definitions from other ontologies. *Implementation* builds ontology models in a computational language. Finally, the developed ontologies are *evaluated*. Within the development process, knowledge of a given domain is continually acquired. METHONTOLOGY is however not suitable for building large ontologies. It does not consider the reuse and re-engineering of non-ontological resources and provides no detailed guideline on the activities for ontology reuse and re-engineering processes (Ǵomez-Ṕerez et al. 2009). By contrast, the NeOn Methodology provides guidelines on developing ontology networks and takes into account the existence of multiple ontologies in such ontology networks, as well as the reuse and re-engineering of knowledge resources including both ontological and non-ontological resources (Ǵomez-Ṕerez et al. 2009). The NeOn Methodology identified 9 different scenarios for building ontologies and ontology networks. Among these is also the scenario for developing an ontology from scratch and the scenario of reusing non-ontological resources.

In our approach, the guidelines of the two methodologies are applied in different steps. At the top level, the following steps are carried out:

- **Understanding use case requirements**: in this activity, the use case scenarios are analyzed to derive the ontology requirements specification, which includes the purpose, the scope and the implementation language, target group and intended uses of the ontologies to be used in the NIMBLE platform, and the set of requirements the ontologies should fulfill. This analysis identifies the main usage of the ontologies as well as the required ontology modules.

- **Identifying reusable data elements** as well as ontological and non-ontological resources. In this activity, the relevant data elements are extracted based on the ontology requirements specification. Relevant ontological resources such as GoodRelations[1], SKOS (Simple Knowledge Organization System), Moda-ML (Middleware tools and documents to enhance the textile/clothing supply chain through XML) [2] as well as non-ontological resources including Universal Business Language (UBL) standard, eCl@ss[3] etc. are reviewed to determine to which extent they are reusable for our purposes.

- **Modelling, Extension and Adaptation**. In this step, the required ontology modules are modeled and reusable resources are identified. The ontology modules can either be built from scratch or by reusing ontological and non-ontological resources. The different ways of reusing ontological and non-ontological resources which are mentioned in the NeOn Methodology can be taken as guidelines in this step. Since a formal foundational Ontology such as BFO or DOLCE can provide a set of upper level concepts to facilitate interoperability, we take this approach (using BFO) during the structuring and organizing of the ontology elements.

[1] http://purl.org/goodrelations/
[2] www.moda-ml.org
[3] http://www.eclasscontent.com/

- **Evaluation with use cases**. In the NIMBLE platform, we will initially evaluate the semantic models based on the use case scenarios. Later, the semantic models will be continually adapted and improved until we reach a usable and stable version. For example, in order to support product publishing and product search the metadata of the catalogue needs to be maintained as an ontology in the NIMBLE platform. The catalogue related ontology would be then be continuously used and evaluated in the process of product publishing and product search in future use case scenarios.

This deliverable describes the architecture of the ontology network for the NIMBLE platform, and core modules in the ontology network. It explains how to extend the core modules for use cases. One limitation of the description is that we do not list and describe exhaustively, all concepts and properties of the ontology modules. There are two reasons for this:

- The number of ontology elements including concepts and properties makes the description hard to maintain and understand without consulting the context of the use cases. In the product catalogue ontology module, there are already more than hundred elements. To support detailed descriptions of different product resources, we have transformed the product category taxonomy, eClass, into an ontological resource, which contains 41,000 product categories and 17,000 properties[4]. In the NIMBLE platform, we have not only the product catalogue ontology module and the eClass taxonomy, but also many other modules.

- There are many different extension modules: the NIMBLE platform needs to maintain various tangible and intangible resources provided by companies from different domains and each resource can have its own features. Thanks to the extensible mechanism of the ontology network, domain-specific ontologies can be extended to precisely describe the resources in different domains. The modules include at present a product taxonomy for the furniture industry and for the textile industry, and will be extended to support other industries. It is impractical to describe each module as well as the elements in the module in detail, in the form of a word processor document.

2 Requirements on Semantic Modelling and Reusable Resources

2.1 Major requirements on Semantic Modelling

NIMBLE provides a B2B platform for smaller manufacturing businesses and suppliers to interact with each other in business workflows. A successful B2B platform must ensure that the customers can find their required product and that different actors can communicate with each other within business processes, which implies that the meaning of interchanged information must be understandable across systems. Companies can describe their products and product catalogue in different ways, with different structure and vocabularies. This may not cause a

[4] http://wiki.eclass.eu/wiki/Category:Products

problem when there is only very limited number of actors in a B2B platform, because different actors can possibly find a way to understand the structure and terminologies of other partners. However, this assumption cannot be made for the NIMBLE platform. Without a shared common structure and vocabularies, it is difficult for the NIMBLE platform to integrate the product description published by different actors and ensure that the published products can be found by potential customers. XML schemas can define a common data structure with user-defined vocabulary to enable syntactic interoperability, but XML itself has no intrinsic semantics and is not sufficient to enable semantically interoperability. Tags in XML do have explicit pre-defined semantics but this semantics is maintained in the application. So, with the same vocabulary, actors or AI-based agents can have different interpretations of the same tags. Moreover, there is no explicit semantic relationship between vocabularies and between the members of the vocabulary. To have the same mutual understanding of the shared vocabularies and to ensure interchanged information remains understandable across heterogeneous systems, semantic modelling i.e. ontology is required. In computer science, ontology is a representation of a shared conceptualization of a specific domain (Gruber 1995). An ontology provides explicit formal specifications on common vocabularies including concept, properties and constraints, used in a domain. It can be used to support the communication and semantic interoperability across distributed heterogeneous systems. In the NIMBLE platform, the most prominent requirement on the semantic modelling is therefore the virtualization of the various types of tangible resources (e.g. products) and intangible resources (e.g. services, catalogues) that companies provide.

The platform design foresees to maintain all relevant data elements and their links via the Semantic Metadata Repository (SMDR). Based on the semantic modelling as well as the metadata repository, resources can be semantically annotated and discovered through semantic search. Details on the resource publishing and search are presented respectively in the deliverable D3.2 Catalogue Ingestion and Semantic Annotation, and D3.3 Product and Service Search Engine and Search Mediator. After the sought-for resources are discovered within the platform, the next important step is to initialize negotiation between partners and start business workflows. In the process of the negotiation and different activities in business workflows, various documents are exchanged, e.g. product design documents and product order documents. Thus, the next important requirement on the semantic modelling is the digital representation of the numerous business processes, so that the right activity can be initialized and the right information can be exchanged.

In order to have requirement specifications for the semantic modelling, we went through the use case scenarios described in deliverable D1.1 “Requirements and Collaboration Design for Manufacturing and Logistics in Four European Use Cases”. One example of a requirement specification expressed with semantic modelling comes from the Micuna use case that is summarized in Table 2. An excerpt of entities involved in this use case is shown in Annex A. In the table, we have sketched a list of competency questions to determine the scope of the ontology. Specification of the competency questions follows the techniques defined ontology engineering methodologies (NOY and MCGUINNESS, 2001, USCHOLD and GRÜNINGER, 1996). It serves as constraints on what the ontology can be (USCHOLD and GRÜNINGER, 1996), and do not need to be exhaustive (NOY and MCGUINNESS, 2001). The listed competency questions are provided with linkage to intend uses, and have been discussed with technical partners in Micuna use case. They are in relative high-level and can give good coverage for the ontology requirements in Micun us case. Based on the requirements specification and the extracted entities, a draft data model related to Micuna use case is created, as illustrated in Figure 1. In the example data model, PostalAddress and ContactPerson can be used to describe contact details of a company. PaymentTerms and DeliveryTerms are related to business conditions. Each company can also provide items i.e. resources, which can be specified in detail with sector-specific properties.

Table 2 Ontology Requirements Specification from Micuna use case

1	**Purpose (of the furniture ontology)**
	Create a semantic representation of the entities managed in the scope of childcare furniture to enable business activities between partners from this sector with actors from other industrial sectors when they get involved in joint activities. - Enable the search of all the resources required for the supply chain definition in the furniture scenario through their semantic representation. - Semantic representation of physical resources, such as products, components, pieces or materials, as well as intangible resources such as delivery terms, product specifications, production plans or business conditions. - To provide common understanding of the information to be monitored in the furniture domain supporting adaptive supply chain logistics.
2	**Scope**
	The ontology should cover all the resources involved in the childcare furniture scenario: - Partner profile: manufacturer of furniture, manufacturer of supply goods, retailer, etc. - Product specifications: quality, reliability, price, etc. - Service specifications: delivery term, conditions, price, etc. - Production processes: machining, painting, assembly, packaging, etc. - Regulatory aspects: normative, legislation in markets, certifications, etc. - Environmental issues: recyclability, material sustainability, etc.
3	**Implementation Language**
	OWL-DL
4	**Intended Users (of the Ontology)**
	The main users in the Micuna use case are: Manufacturer, Supplier, Logistics Operator, Retailer. They can be grouped into the following two user roles: **Resource requesters** who are looking for external resources which cannot be covered by its own resources or aims at improving at a different level. It could be a manufacturer, supplier, logistic operator or retailer. For example, the manufacturer needs to find adequate supply

	partners to reach certain production objectives. **Resource Providers** who can offer some kind of resources. For example, the manufacturer may want to offer child furniture to retailers.
5	**Intended Uses**
	Use 1. **Search for needed resources and related business entity information**. This resource could be products, components, pieces, materials or production processes. Use 2. **Publish product and services catalogue** increasing the visibility of the goods they offer. Use 3. **Publish profile and contact information of business entity**. Use 4. **Negotiating business issues** with potential partners, exchanging data in order to achieve collaboration agreements for regular or occasional supplying; continue business transactions after negotiating. Use 5. **Managing the product's End-Of-Life phase** from the customer negotiating the product renovation or donation to charitable organisations.
6	**Ontology Non-Functional Requirements**
	The ontology should be aligned to standards as much as possible. Terms managed in the ontology should be those commonly used in the respective domain.
7	**Ontology Functional Requirements: Competency Questions**
	CQ1. What is the profile of the business entity with respect to the information needs of other business partners? This question is close related to the intended use 1 and intended use 3 for the publishing and search of business partners. CQ2. What kind of products and services are offered by a business entity? This question is close related to the intended use 1 for the search of business partners that can provide sought-for resources. CQ3. What kind of features/specification does the published resource have, provided these features need to be known by customers in order to refine their queries? This question is close related to the intended use 1 and intended use 2 for publishing and search of sought-for resources. CQ4. Given some product or service-related characteristics, which business entities are qualified providers on the NIMBLE platform? This question is close related to the intended use 1 and intended use 5 for the search of sought-for resources. CQ5. Which business transaction process is related to a resource, which activities should be carried out in the business transaction process and which kind of documents will be exchanged? This question is close related to the intended use 4 and intended use 5 for the negotiation of business issues and guiding business transactions.

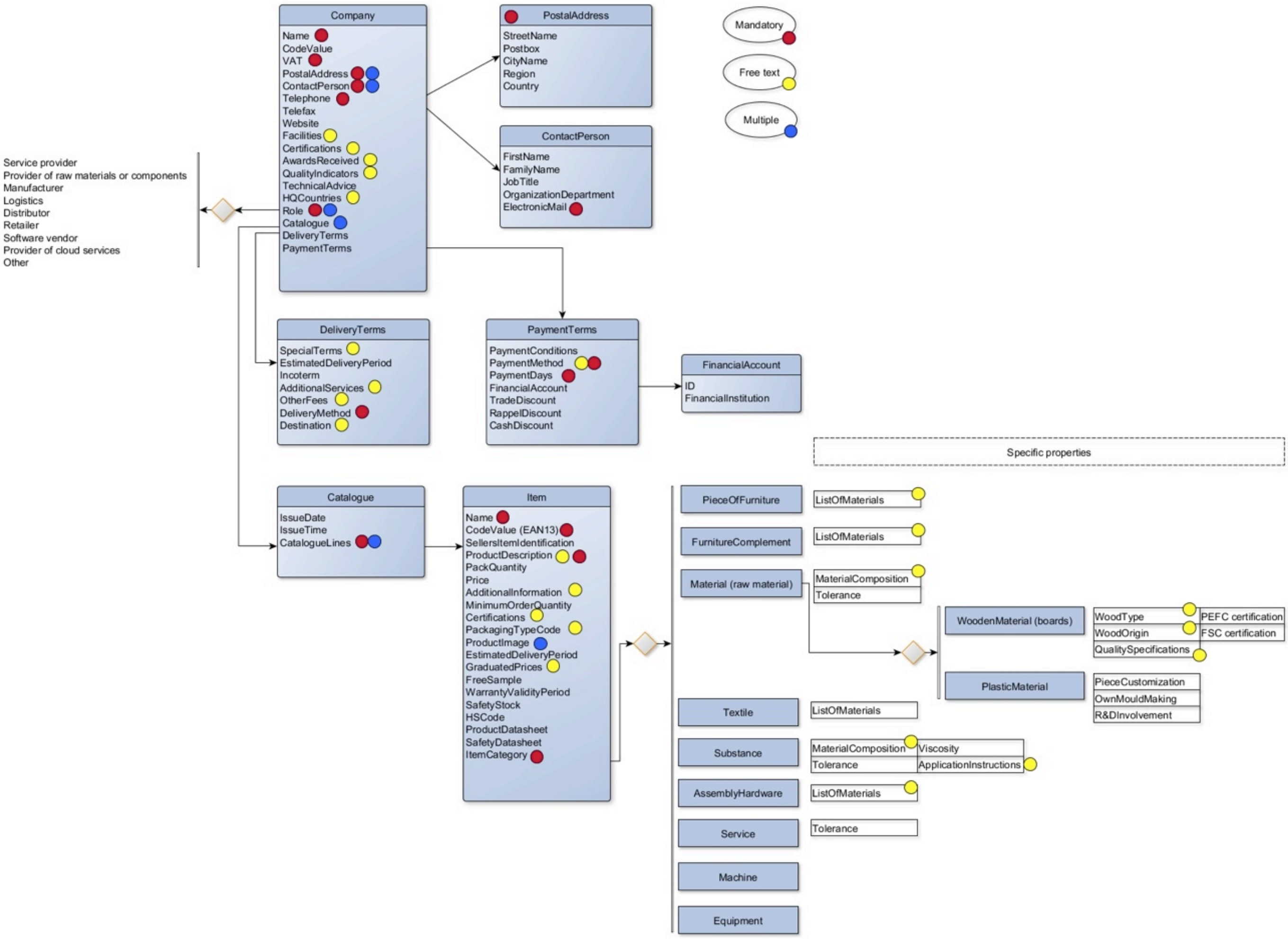

Figure 1 Excerpt of Data Model related to Micuna Use Case

Based on the analysis of the general purpose of the NIMBLE platform, as well as requirements from the use case scenarios, we have summarized two core ontology modules: the catalogue ontology and the business process ontology. The catalogue ontology includes machine readable metadata intended for sharing knowledge and exchanging information in relation to resource descriptions. The demand on a resource is a trigger for business processes. The business processes are semantically described by the business process ontology.

2.2 Reusable ontological and non-ontological Resources

This section describes reusable ontological and non-ontological resources related to the catalogue ontology and business process ontology of the NIMBLE platform.

2.2.1 Resources for Catalogue Ontology

Lee et al (2016) proposed an operational product ontology for constructing an e-Catalog as well as for the search and discovery of products. The proposed product ontology includes the key semantic concepts such as products, classification schemes, attributes for each class and domain constraints on each attribute, units of measure and their relationships among each other. Each product can be classified with more than one classification scheme. This ontology can be used for the description of products, but does not cover important aspects in E-commence such as delivery and payment.

GoodRelations (Hepp, 2008) is a lightweight ontology for describing offerings of tangible goods and commodity services on the Web. It covers many important aspects of E-commence, including Web resources, offerings made by means of those Web resources, legal entities, prices, terms and conditions etc. The GoodRelations ontology can be extended with the general product category taxonomy eClassOWL[5], which has been transformed from the international products and services classification standard eCl@ss version 5.1 [6], for the description of types and properties of products and services on the Semantic Web (Hepp 2005). Meanwhile, it has been added almost entirely to the schema.org vocabulary[7], can be used and is fully supported by Google and Yahoo for E-commerce scenarios. It is a valuable candidate reusable ontological resource for the catalogue ontology in the NIMBLE platform. However, it has still some limitations. Concepts in many important E-commence aspects, such as business entity, delivery, warranty and payment, are not specified at the detailed level that the B2B E-commerce use cases of the NIMBLE platform (e.g. Micuna use case) expect. In addition, some concepts e.g. Certificate, which are required for various reasons in B2B e-commerce, for example, for the description of business entity and products, are not covered in the GoodRelations ontology.

In B2B e-commerce, many standards provide means to exchange document between trading partners, such as ebXML, UBL and Moda-ML (Liegl 2010). Among them, UBL is the first implementation of the United Nations Center for Trade Facilitation and Electronic Business (UN/CEFACT) Core Components Technical Specification in XML (Yarimagan and Dogac 2009). It provides a free library of standard electronic XML business document for supporting e-commerce. It is conformance with the ebXML CCTS ISO/TS 15000-5:2005 standard (OASIS Universal Business Language TC 2014). In addition, the UBL-version 2.1 has already been officially specified as ISO/IEC 19845:2015 standard, which defines a generic XML interchange format for business documents that can be restricted or extended to meet the requirements of particular industries (ISO/IEC 19845:2015). The library in UBL covers XML schemas for almost all data elements such as "Address", "Item", "Payment", “Delivery” and “Warranty”, which are necessary for catalogue management in NIMBLE platform. These elements connected to broader data elements related to supply chain operations such ordering, fulfilment, delivery and invoicing, and are in a sufficient detail level for B2B E-commerce. However, it contains not only the data elements what are necessary for catalogue management, but also many data elements which are not needed for the purpose of e-catalogue. In order to have a suitable set of data elements are well as relevant XML schemas for catalogue, Forsberg has tailored UBL schemas to Svekatalog UBL Catalogue 2.1 (Forsberg 2011). It is however still in the form of XML schema and cannot achieve semantic interoperability to avoid ambiguous meaning across heterogeneous information systems. To achieve semantic interoperability, Yarimagan and Dogac have taken a step forward to develop a conversion tool that reads UBL

[5] http://www.heppnetz.de/projects/eclassowl/
[6] https://www.eclass.eu/en.html
[7] http://schema.org/

schemas and creates corresponding class, object property, and existential restriction definitions in OWL (Yarimagan and Dogac 2009).

In summary, there are ontological and non-ontological resources such as GoodRelations and UBL, reusable for the representation of various types of tangible and intangible resources that companies provide. But they all have different kind of disadvantages and are not sufficient for the NIMBLE platform. The GoodRelations could be used for describing offerings of tangible goods and commodity services on the Web. It can be extended with other product category taxonomies such as eClassOWL for detail specifications of the offered tangible goods and commodity services. But the concepts in many important E-commence aspects, such as business entity, delivery, warranty and payment, are not in a detail level that the B2B E-commence use cases in NIMBLE expect. Furthermore, the eClassOWL is based on eCl@ss version 5.1, while the current eCl@ss version is already 10.0.1. The library in UBL covers almost all data elements such as "Address", "Item", "Payment", “Delivery” and “Warranty”, which are necessary for catalogue management in NIMBLE platform. They are also in sufficient detail level. However, UBL contains too much elements which are not useful for our needs, and it is in XML schemas which is not able to achieve semantic interoperability. Some efforts have been done to tailor the UBL schemas and to transform the UML schemas to ontology (Forsberg 2011, Yarimagan and Dogac 2009). However, these efforts have not been integrated. We will reuse the ideas in GoodRelations as well as the UBL standard to have a UBL standard-based catalogue ontology for the purpose of catalogue management in NIMBLE platform for the representation of various types of tangible and intangible resources. This UBL standard-based catalogue ontology will be able to be extended with other product and service taxonomies/ontologies for detail specification of the offered tangible and intangible resources.

2.2.2 Resources for Product Catalogue Taxonomies

Product category taxonomies are knowledge bases structured as a hierarchy of categories used to classify products. Each category in the hierarchy might be associated with a set of properties to be used to describe products with detailed and semantically relevant characteristics. We analysed a set of available category taxonomies in order to identify the most appropriate one to be used as the default taxonomy in NIMBLE. (Stolz et al. 2014) provides the following table about the coverage of the available taxonomies called classification systems.

Table 3 Statistics about taxonomies (Stolz et al. 2014)

Classification system	Number of levels	classes	properties	individuals	top-level c.	Class distr. (%)
CPC Ver.2	5	4,409	0	0	10	18
CPA 2008	6	5,429	0	0	21	53
CPV 2008	4	10,419	0	0	254	6
eCl@ss 5.1.4	4	30,329	7,136	4,720	25	18
eCl@ss 6.1	4	32,795	9,910	7,531	27	16
ETIM 4.0	2	2,213	6,346	7,001	54	8
FreeClass 2012	4	2,838	174	1,423	11	21
GPC 2012	4	3,831	1,710	9,562	37	17
proficl@ss 4.0	≤ 6	4,617	4,243	6,815	17	36
WZ 2008	5	1,835	0	0	21	33
Google prod. tax.	≤ 7	5,508	0	0	21	17
productpilot	≤ 8	7,970	0	0	20	28
BMEcat	na	na	0	0	na	na

As can be seen from the table 3, eClass is the taxonomy with the highest number of classes (i.e. categories) and properties. It is actively maintained by European authorities and improved constantly with a continuous release cycle. Being a cross-sector taxonomy and having the most suitability with our use cases as well, we decided to adopt eClass as the built-in, cross-sector taxonomy to be utilized in the scope of product publishing operations in NIMBLE.

In addition to enriching product descriptions with machine processable semantic metadata, such taxonomies establish interoperability through the usage of standardized set of elements they introduce. Thus, as many products are described with these standardized concepts, search capabilities provided on top yield better results. Standardized concepts can also be given multiple translations, enabling the discovery of products in a multilingual manner.

NIMBLE also has the ambition to be extensible with sector-specific taxonomies that are more suitable to describe products with even more relevant characteristics. Driven by our use cases, we utilize a Furniture Sector Taxonomy derived from the ISO 10303-236 and; Moda-ML ontology as resource for better annotation of products in child furniture and textile sectors respectively. Details about these resource are given in Section 3.2.3.2 and 3.2.3.3.

2.2.3 Resources for Business Process Ontology

A substantial part of literature deals with aspects of modelling business processes and many business process management approaches, such as BMPN and BPEL, have been proposed and accepted. Each has its own focus and defined vocabularies. In order to have shared vocabulary and solve the semantic interoperability problem, many Business Process Modelling related Ontologies have been proposed in the last few years (Hepp and Roman, 2007; Pospocher et al 2014; Von Rosing et al 2015). Hepp and Roman (2007) have discussed the representational requirements of Semantic Business Process Management and proposed a related ontology framework. Pospocher et al (2014) proposed the BPMN Ontology which describes a formal ontological description of the BPMN and provides a classification of all the elements of BPMN. Von Rosing et al (2015) introduces a BPM ontology that provides fundamental process related concepts within the area of process modelling, process engineering, and process architecture.

These business process management approaches as well as ontologies can be taken as basis for understanding the areas in the business process ontology. However, this deliverable is not dedicated to have a full survey of all business process modelling approaches as well as related business process ontologies.

This deliverable is not intended to have a theoretically perfect business process ontology to cover each business process elements in enterprises, but only focuses on the employment of a simple, practice business process ontology to facilitate the interoperability of business transaction processes in B2B e-commerce environment i.e. in NIMBLE platform. Based on the use case requirement analysis in NIMBLE platform, the main usages of the business process ontology in NIMBLE platform include:

- Organize roles and party/person for a process/activity
- Prepare sequence of activities for a business process
- Help to exchange documents based to the right schema e.g. UBL Order in different activity

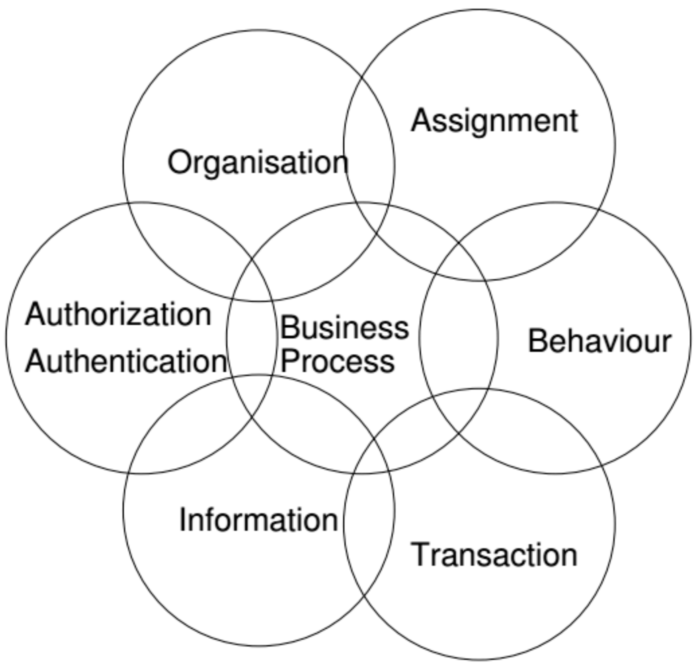

Figure 2 A Business Process and some of its aspects (Axenath et al 2005)

As shown in Figure 2, Axenath et al (2005) have list several aspects of business process. Among them, the Information Aspect, Organisation Aspect and Behavioural Aspect are generally agreed to be the most three important aspects of business process. Thus, ontological and non-ontological resources, which cover the most three important aspects of business process and are in line with the main usages of the business process ontology in NIMBLE platform, will be most interesting for us in NIMBLE platform.

Moda-ML is a vertical standard for the electronic data exchange in the Textile-Clothing sector (Censoni et al 2002). It defines the reference collaborative processes, activates as well as related

XML documents for the business transaction in in the Textile-Clothing sector (Reference Collaborative Processes in eBIZ/TC Upstream 2013). In addition, it has proposed a business process ontology [8] for the purpose of business process interoperability in B2B e-commerce in the textile domain. The business process ontology covers the three basis aspects of business process with the concepts such as “Actor”, “Process”, “Activity” and “Document”. Although it focuses on the B2B e-commerce in textile domain, it can be handled as a reusable ontological resource for the construction of a business process ontology in NIMBLE platform to support business transaction process interoperability in different domains.

3 The NIMBLE Semantic Modelling

In order to maintain semantic structure within the vocabulary of the NIMBLE platform, we have adopted a gradual approach to introducing ontologically sound concepts into the system.

In an ideal world, the system would be knowledgeable about the business domains both in terms of vocabulary available and in terms of processes that need to be executed in certain situations. The aim of such an idealised, AI based NIMBLE would be to run all necessary processes autonomously: search for potential suppliers or logistics providers, negotiate with the (automated) counterparts favourable business terms, and then initiate the agreed processes that will in turn, be executed by autonomous workers, e.g. a self-driving truck.

In the practical world of Internet platforms, there are many unknowns that still require human intervention while some of the processes can be automated as long as there is common agreement on the terms that define such processes. The NIMBLE semantic modelling takes an evolutionary approach: firstly, we collect domain specific vocabularies that cover some domain of knowledge: e.g. the furniture ontology, or the eClass ontology, or the Moda ML ontology. Secondly, we use existing tools to interpret semantic annotations of data to better support certain user tasks. The Apache Marmotta Linked Data Server is one such tool. Thirdly, we map data values coming from business processes, to ontological terms and use the semantically enriched values in advanced services offered on the NIMBLE platform. In this chapter, we focus on the first aspect: to collect diverse ontological resources and to align them sufficiently so that they can later be used for semantically enriched business automation available via the NIMBLE platform.

3.1 Ontology Network Structure

As illustrated in Figure 3, the NIMBLE ontology network revolves around the core ontology modules: *catalogue ontology* and *business process ontology*. The catalogue ontology holds core vocabularies for semantic description of partners in the NIMBLE platform as well as tangible and intangible resources that partners provide. Based on the catalogue ontology, resources can be semantically published and searched. When partners have found their desired resources in the NIMBLE platform, business processes can be initialized. The business process ontology

[8] http://www.moda-ml.org/moda-ml/Ontologies/Nimble/ModaML-ProcActDoc.owl

contains machine readable, core vocabularies for the semantic description of business processes. Surrounding these two core ontology modules, many further ontology modules can be added to specify the resources and business processes in more detail. For example, the general product category taxonomy eClass can be used to better specify the description of individual resources, For sector-specific purposes, appropriate domain ontologies can also be added. As shown in the figure, the furniture taxonomy is an extension that allows a more precise description of the resources found in the furniture industry.

All relevant data elements and their links in the ontology modules are maintained in the semantic metadata repository. In the NIMBLE platform, we are using an open source RDF triple store and Linked Data Server – Apache Marmotta – as semantic metadata repository. This allows us to combine data triples from the ontology's TBox with data triples from the ABox and to draw inferences that are otherwise difficult to achieve. Based on the ontology modules in the semantic metadata repository, data triples would be populated with different approaches, such as semantic annotation. With the managed ontology TBox and data triples, it allows the execution of SPARQL to semantically retrieve and manipulate data stored in the SMDR. More details on the semantic web technologies within the NIMBLE platform is given in the Annex B. A detail description of the semantic population for catalogue ingestion, and semantic search is given respectively in the deliverable D3.2 "Catalogue Ingestion and Semantic Annotation" and D3.3 "Product and Service Search Engine and Search Mediator".

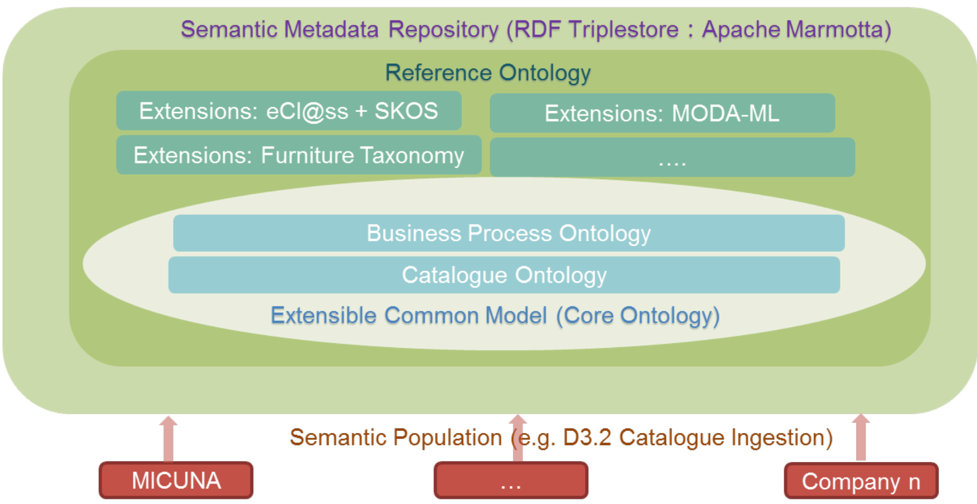

Figure 3 Overview of NIMBLE ontology network structure

The relationship between the ontology modules in the NIMBLE platform is illustrated in Figure 4. The catalogue ontology is mainly used to represent particular aspects of companies as well as the resources provided by the companies. Demand for specific resources is the trigger for the initialization of business processes, which is represented in the business process ontology. Within the business processes, each company can play different roles. Extension modules are used to facilitate detailed specification of resources in the catalogue ontology as well to specify metadata for to be exchanged (business) documents in the business process ontology.

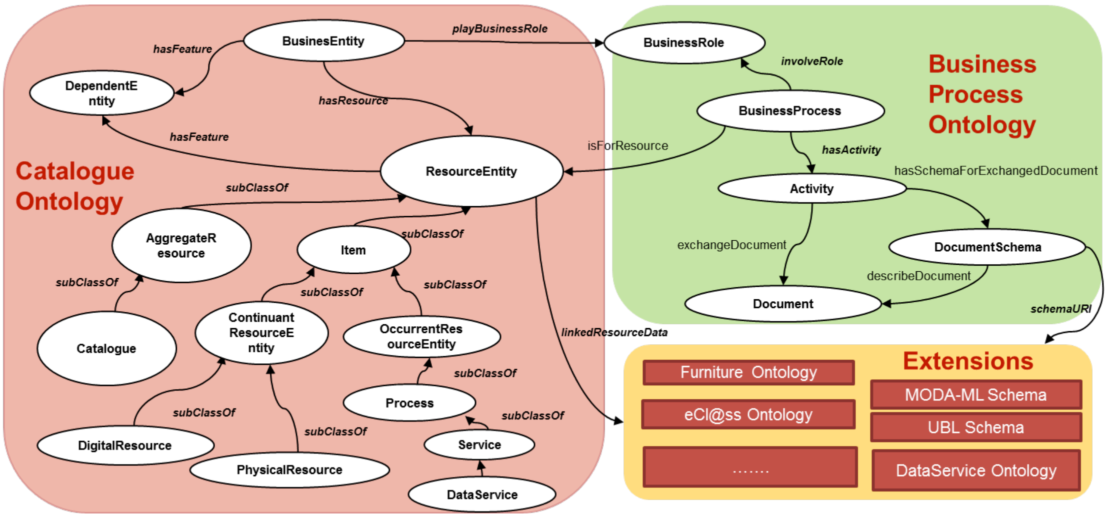

Figure 4 High-level Overview of the ontology network classes and properties

3.2 Ontology described module by module

In this section, we will go through the core modules one by one, and then give a short discussion on the extension modules.

3.2.1 Catalogue Ontology

The catalogue ontology is a machine readable repository of metadata intended for sharing knowledge and exchanging information in relation to resource description and resource discovery. The ontology is based around the concepts Business Entity, Resource Entity and Dependent Entity. Main usages of the catalogue ontology include:

- Help to publish resources in the NIMBLE platform, so that they can be found by other partners on the platform, in order to initiate business processes.
- Support customers with information related to products/services and providers, so that offers can be compared and evaluated to decide if makes sense to contact the supplier of the product/service.
- Provide basic information such as contact details of companies etc. to help start the negotiation for business processes.

3.2.1.1 Basic Elements

The core aspects within the product catalogue domain are Business Entity, Resource Entity and Dependent Entity. When there is a need for more detailed domain specific descriptions for the Resource Entities, the core concepts can be extended with additional taxonomies such as eClass taxonomy or furniture domain specific taxonomy with linked data principles. The basic relationship between the entities is depicted in Figure 5. A business entity owns resource entities, and can offer some of the resource entities to others. The business entity and the resource entities are described in the dependent entity with properties such as contact and price.

In addition, the resource entities can be detailed and further specified with extensions. In the following, these four concept categories are described:

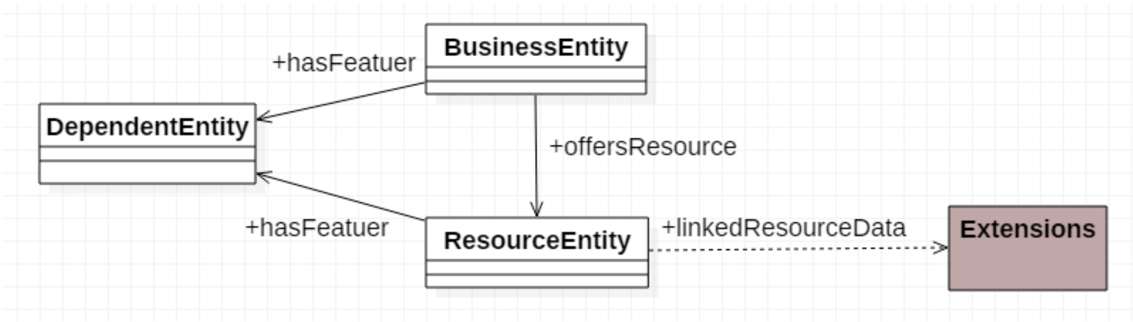

Figure 5 Relationship between main concept categories

Business Entity Concept

A *Business Entity* is a legal agent that participates in some activities in a supply chain. It can be a legal *Party* or a *Person,* which has naturally detailed description such as name and contacts. A *Business Entity* can offer some *Resources*, or take part in sub-processes in a supply chain, for example, for resources acquisition. When a *Business Entity* performs some activities in a supply chain, it takes usually a *Business Role.* Depending on varying point of views, there are different way to categorize supply chain partners. In NIMBLE platform, we will follow the categorization of supply chain participants described by Gibson et al. (2013). A typical *Business Role* in a supply chain include *End User*, *Manufacturer, Supplier, Retailer* etc. The *Business Role* in supply chain is more detail described in the Business Role Concept in section 3.2.2.1.

Resource Entity Concept

A *Resource Entity* is a product or service that a *Business Entity* holds. It can be an *Item* or *Aggregated Resource* such as *Catalogue.* The entity *Item* is arranged based on the taxonomies in the Basic Formal Ontology (BFO) (Smith et al. 2015): *Continuant Resource Entity*, *Occurrent Resource Entity*. *Continuant Resource Entity* is a kind of resource that continues to exist through time, for example, some *Physical Product* or *Digital Resources*. Whereas, *Occurrent Resource Entity* is a kind of resources that exists for a certain time period, such as *Logistic Service*. Each *Resource Entity* has some resource specific technical characteristics or commercial properties such as price and specification. These kinds of specifications are detailed in the *Dependent Entity* description.

Dependent Entity Concept

The next major category of concepts is the *Dependent Entity.* The concept is derived from category *Dependent Continuant* entity in the BFO ontology. In the BFO ontology, the category of *Dependent Continuant* entity includes entities that are ontologically dependent on other *Independent Continuant* for their existence. Qualities and Realizable Entities such as mass, or colour are examples of *Dependent Continuant* entities that depend on other *Independent Continuant* e.g. tomato. In the catalogue ontology, we handle the *Business Entity* and *Resource Entity* as a kind of *Independent Continuant*, and the concepts in the category of *Dependent*

Entity such as *Quality Indicator* and *Price* as a kind of *Dependent Continuant*. From this point of view, concepts in the category of *Dependent Entity* are dependent upon and can be used to better specify the *Business Entity* and *Resource Entity*. Some example concepts under the category of *Dependent Entity* are: e.g. *Contact*, *Financial Account, Business Role, Price* and *Commodity Classification*. In the above listed example concepts, the *Contact*, *Financial Account,* and *Business Role* are attributes of the *Business Entity*. And the *Resource Entity* is bearer of the *Price* and *Commodity Classification.*

Extensions

The *Extensions* is not an ontological concept such as *Business Entity* for the share of common semantic meaning. It is more like an open interface slot saved for further product ontologies for detail specifying *Resource Entity*. For example, the product ontology, furniture taxonomy, is a type of extension for detail describing the products in furniture domain. The reason for providing this *Extensions* slot is for keeping extensibility and usability of the data model for diverse use cases. In different use cases or supply chains, diverse resources are exchanged. Each resource has its own domain specific properties. It is typically hard to manage every single property for all different resources in various domains in a single ontology. There are usually domain specific product ontologies for the description of resources in a specific domain. In the NIMBLE platform, we intend to reuse this kind of domain specific resources to detail the *Resource Entity*. For example, the eClass taxonomy can be applied to classify a *Resource Entity,* and the respective properties in an eClass category are able to specify the *Resource Entity*. In the same way, Furniture Taxonomy and Textile Taxonomy can be used to extend the data model for the specifying the products respectively in the furniture domain and in the textile domain. More details on how to extend the catalogue ontology is given in the section 3.2.1.4.

3.2.1.2 Reuse of non-ontological resource: UBL

Universal Business Language (UBL) is a world-wide standard providing a royalty-free library of standard electronic XML business documents that are commonly used in supply chain operations. UBL also defines common business processes as well-defined data exchange flows among trading partners.

We use Universal Business Language (UBL) as the common data model of our Catalogue Registry and messages to be exchanged via the built-in business processes; as it contains appropriate data elements for catalogue/product management that are also connected to broader data elements related to supply chain operations such ordering, fulfilment, delivery and invoicing. In addition to document schemas that are rather complex, UBL provides a set of reusable building blocks for constructing these documents. UBL provides two main groups of reusable elements. Each element in the first group is named a Basic Business Information Entity (BBIE). As the name implies such elements can only have one value with varying data types. While data types can be built-in data types such as integer, string, date and so on; they can also be associated with additional attributes such as the measurement unit for a quantity, mime type for a binary object, etc.

Entities in the second group are called Aggregate Business Information Entities (ABIE). These are collections of BBIEs and can also have references to other ABIEs. Having specific semantics, document schemas like Catalogue, Request for Quotation, etc. have the same structure with ABIEs.

UBL has good coverage in terms of the data elements required in our use cases. In addition to the high-level Catalogue and other document schemas, it includes concepts for representing companies, persons, catalogues, products, product properties, delivery terms, trading terms and so on. In order to assess the suitability of UBL for NIMBLE, we also mapped the data elements required in the Micuna use case to the data elements defined in the UBL data model. As can be seen in Annex A: Excerpt of Entities involved in Micuna Use Case, which shows the initial mapping effort, most of the Micuna data elements had corresponding elements in the UBL data model.

Considering the catalogue management part of NIMBLE, UBL defines the Catalogue document schema such that it would contain a set of Catalogue Lines corresponding to products traded by a party. Figure 6 shows a graphical representation of a portion of the Catalogue Line data schema. The elements referred with a “cbc” prefix are BBIEs defined in the common BBIE library provided by UBL. Likewise, the ones referred with a “cac” prefix are ABIEs from the common ABIE library. Although we preserved the original schema provided by UBL, we did some modifications based on the requirements of our use cases. First, in order to start with a relatively simple data model, we removed unnecessary elements that are not required in our use cases, mainly for the Micuna use case. Nevertheless, we did some modifications as well. In Annex A, we initially had indicated the Micuna use case data elements that do not have a corresponding in UBL with a TBD mark. Mainly, those are the elements for which we customize the original UBL data model. For example, we added concepts related to payment and delivery conditions to the company the concept.

Considering the messages to be exchanged via the business processes, UBL has a large set of document schemas. Although the complete list of document schemas can be found online at [9], we are using a small subset of those schemas including Catalogue, Order, OrderResponseSimple, RequestForQuotation and Quotation documents. Based on the feedback to be collected in WP4, we will consider reusing documents like CertificationOfOrigin, TransportationStatus, Invoice, etc. Figure 7 shows a portion of the RequestForQuotation document schema which is used in the Negotiation business process. The schema shows the relevant smaller data elements like seller/buyer parties, line item included in the root document schema. The item for which the negotiation process is executed is linked to the root document via the RequestForQuotationLine reference, which in turn includes detailed about the Item, i.e. the product, price, delivery terms, warranty information and so on.

[9] http://docs.oasis-open.org/ubl/UBL-2.1.html

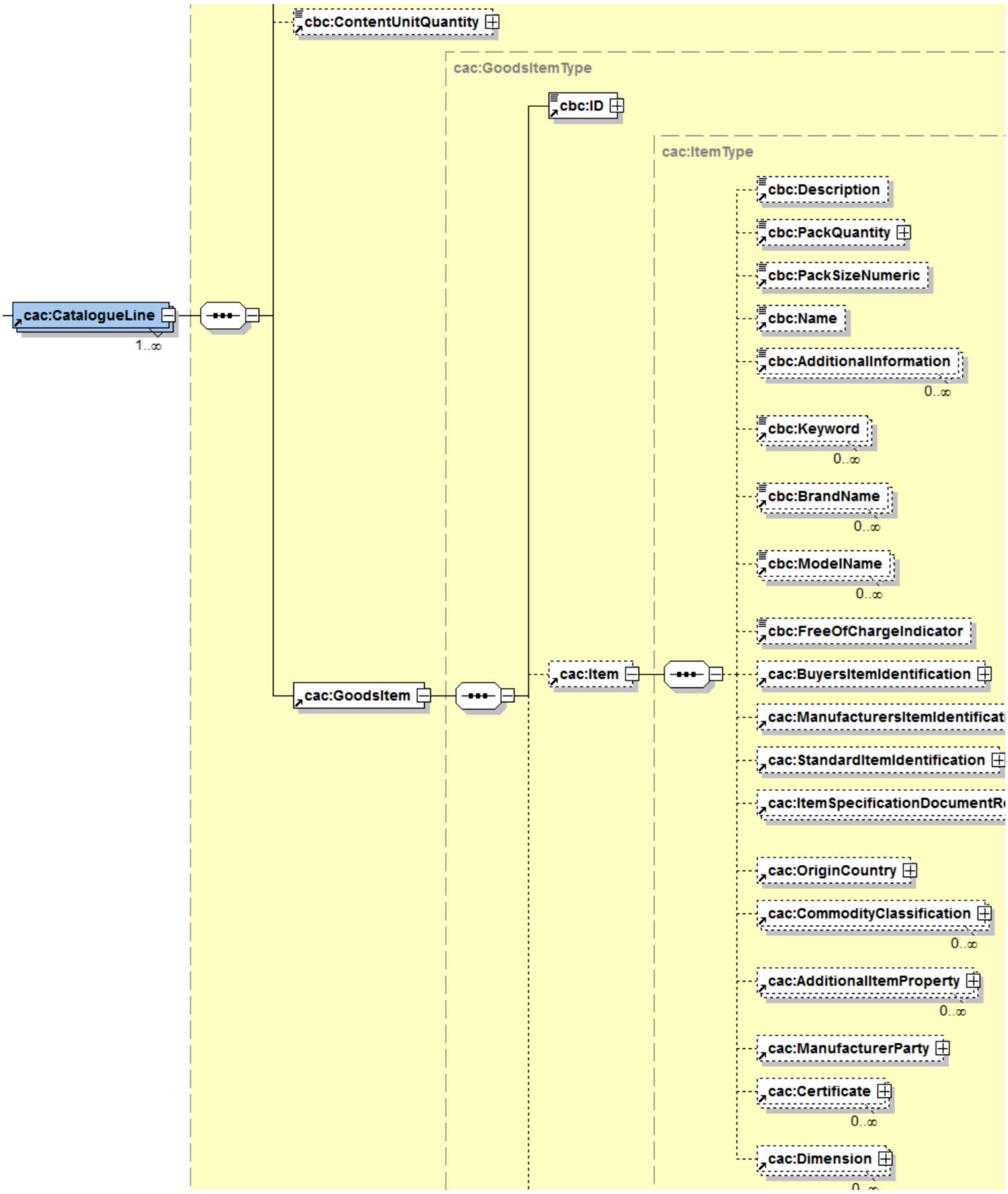

Figure 6 UBL Graphical representation for Catalogue Line schema

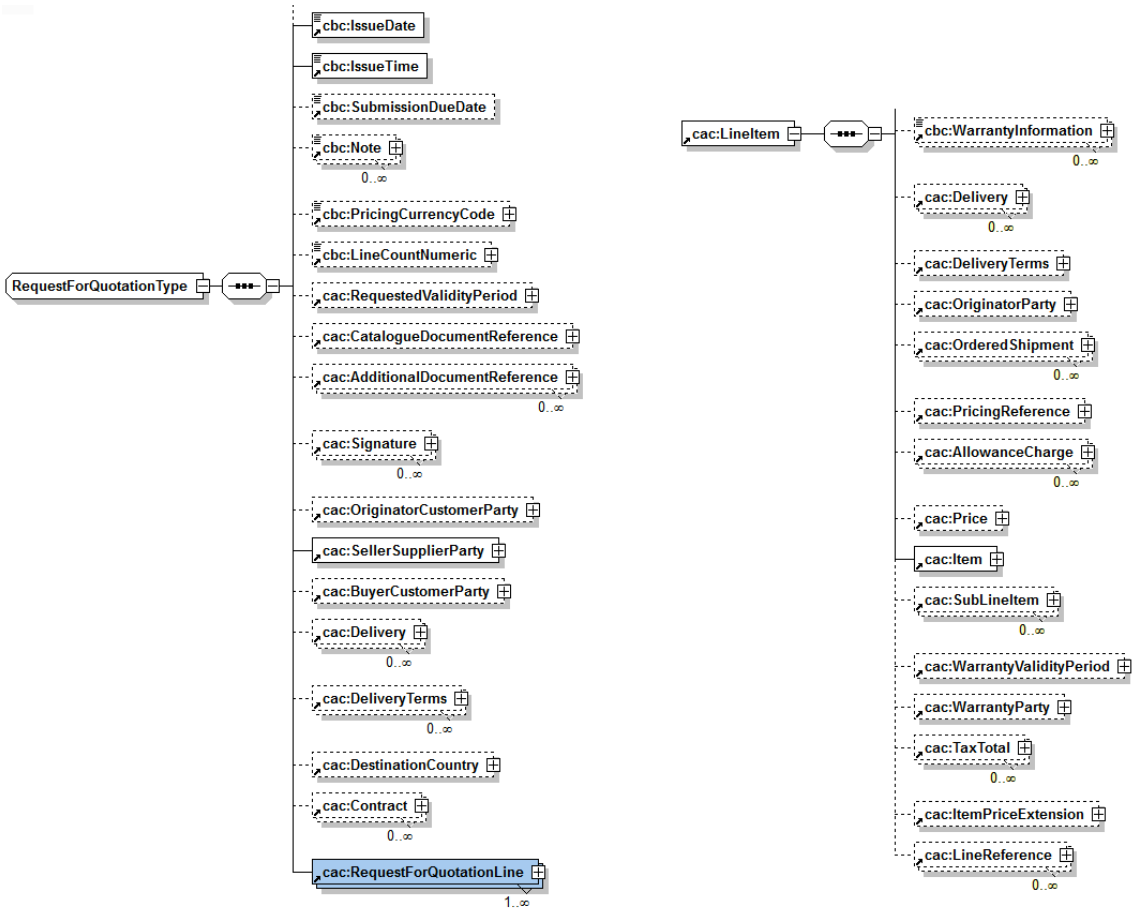

Figure 7 A snapshot of the Request for Quotation document schema

3.2.1.3 Ontology Schema

This sub-section provides a high-level visualization of the ontology schema. It follows the key concepts listed in sub-section 3.2.1.1 Basic Elements and definitions in sub-section 3.2.1.2 Universal Business Language. A high-level visualization of the catalogue domain ontology is illustrated in Figure 8. For visibility reasons, only some of the most important concepts are shown in the figure, and the object properties as well as data type properties are not detailed.

As depicted in the figure, the key concept categories described in sub-section 2.2.1 are included. In a high-level view, a *Business Entity* can offer some *Resource Entities*. The *Business Entity* as well as *Resource Entities* have properties which are detailed in *Dependent Entities*. In addition, the *Resource Entities* can be further specified with other *Extensions:* e.g. eClass ontology for general product classification, Furniture taxonomy for detail specification of products in furniture domain or Moda-ML in the textile domain.

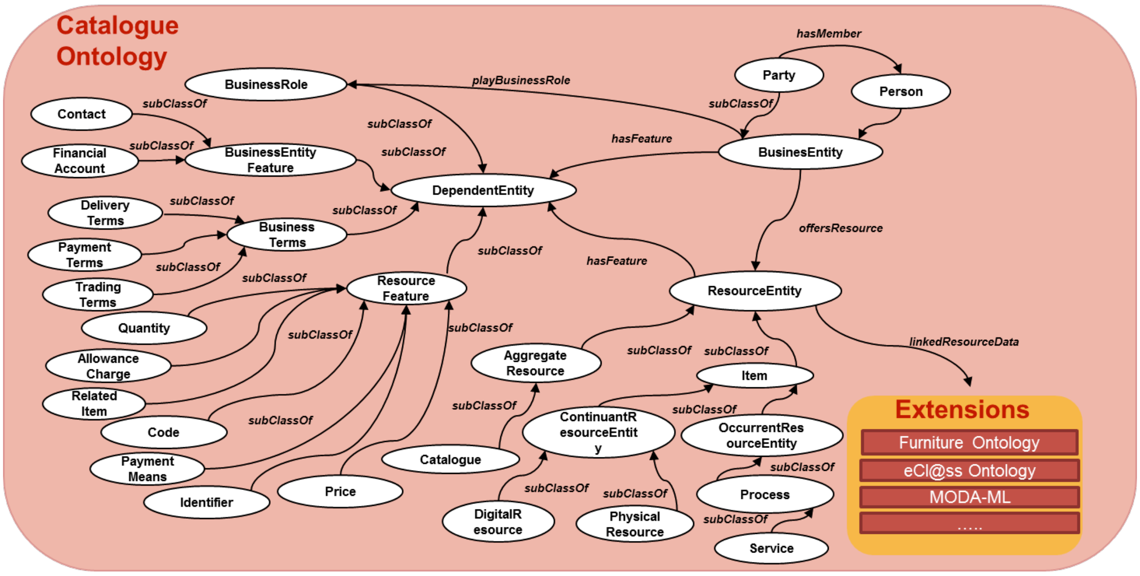

Figure 8 High-Level schema of the catalogue ontology model

Most of the concepts in the ontology schema are derived from the UBL standard data model, which is specified in sub-section 3.2.1.2. The catalogue related concepts and properties in the UBL data model are tailored and transformed into ontology classes and object/data type properties in the above shown catalogue ontology schema. In order to tailor the UBL schema for the needs of catalogue ontology, we have used the following approaches: 1) Mapping the data models (e.g. data model in Figure 1: Excerpt of Data Model related to Micuna Use Case) and the required entities in use cases to UBL elements (e.g. entities in Annex A: Excerpt of Entities involved in Micuna Use Case); 2) Identifying and categorizing the catalogue related data elements in UBL schemas based on the basic elements described in the section 3.2.1.1; 3) adaptions and modifications based on the known reference resources e.g. GoodRelations and Svekatalog UBL Catalogue 2.1 (Hepp, 2008, Forsberg 2011). To transform the tailored UBL schemas to catalogue ontology, we have used a tool called Ontmalizer[10] to have a draft version of RDF ontology. Ontmalizer is able to transform any XML document to RDF as long as the XML document is compliant with an XSD schema. On the base of the draft version, manual adaptions and optimizations are followed. For example, the UBL concepts of *Catalogue* or *Item* are transformed into the ontology classes *Catalogue* and *Item*. In order to describe commercial conditions for the resource entity, the payment, price, warranty as well as delivery related concepts from UBL such as *PaymentTerms* are included. Many other concepts, which are derived from the UBL data models, such as *Period*, *Address*, are for the reason of visibility not listed in the above diagram. The relationships that are in the form of ontology object properties and datatype properties are for the same reason not detailed in the above diagram. For example, a *Catalogue* can have many *Catalogue Lines*. Each *Catalogue Line* can have its own *Warranty Information*, *Warranty Party* and *Warranty Valid Period*. In the example, the Warranty Information is a datatype property in the ontology, the *Warranty Party* and *Warranty Valid Period* are object properties with links to the respective ontology class *Party* and *Period*.

[10] https://github.com/srdc/ontmalizer

After the catalogue related concepts are derived from the standard UBL data model, these concepts are re-arranged based on the top ontology BFO as well as the key concept categories described in the sub-section 3.2.1.1. The current version of the catalogue ontology model can be found at the NIMBLE project repository[11].

3.2.1.4 Extensibility

The catalogue ontology provides common metadata and terminologies for the description the business entities and resource entities in all industry sectors in the NIMBLE platform. However, as resources in each industry sector i.e. product categories have their own features, it is necessary to extend the catalogue ontology with product category taxonomies in order to enrich the specification of the resources in different domains. Product category taxonomies provide knowledge bases on hierarchy of product categories as well as associated properties. eClass, which is an ISO/IEC compliant industry standard for cross-industry product and service classification, is one example of general product category taxonomies. For better utilizing the semantic relations among the taxonomy, the product category taxonomies need to be first transformed into ontology modules in the NIMBLE ontology network.

The product category taxonomy can extend the core catalogue ontology in two ways: inheritance mechanisms, linked data mechanisms. The first way is mainly based on inheritance mechanisms. When we describe an actual resource instance within the NIMBLE platform, we make it as an instance of both the *ResourceEntity* i.e. *Item* class in the catalogue ontology and the respective product category class in the product category taxonomy such as eClass or the Furniture Taxonomy. We can then describe the resources using all the properties from the catalogue ontology as well as the properties from the product category taxonomy. An example is illustrated in Figure 9. The right hand side of the figure is a snapshot of the furniture taxonomy. The namespace "Furniture" refers to the module of furniture taxonomy and the namespace "Catalogue" means the module of the catalogue ontology. As shown in the example, we can specify a resource instance "Furniture:Bed_Individual_1" as instance of both the class "Item" in the catalogue ontology and the product category class "AirBed" in the furniture taxonomy. By doing so, the resource instance is able to be specified with properties both from the core ontology modules as well as from the domain-specific extensions.

[11] https://github.com/nimble-platform/catalog-service-srdc/blob/master/business/ubl-data-model/src/main/schema/NIMBLE-UBL-2.1-Catalog-Subset/ontology/ubl_catalogue_ontology.owl

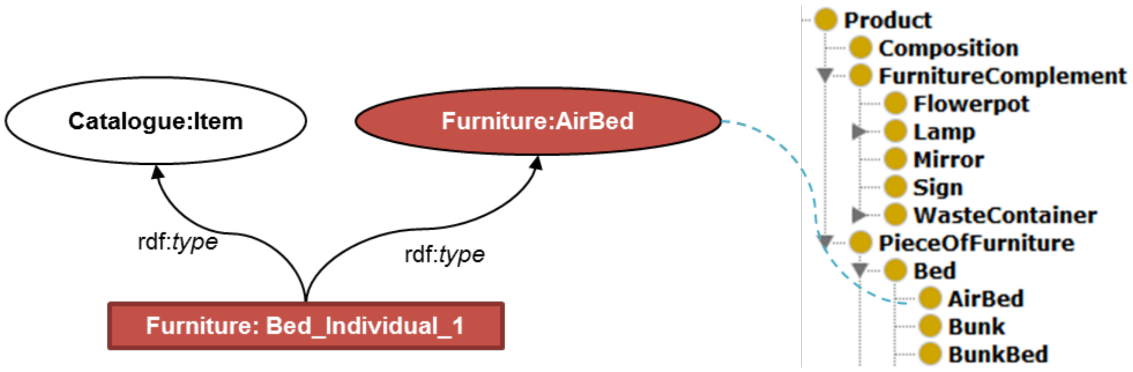

Figure 9 Example Extension with Inheritance Mechanisms

The second way to extend the catalogue ontology is based on the linked data mechanisms. An example is illustrated in Figure 10. Linked Data is a method for interlinking structured data. It refers to data published in such a way that it is machine-readable, its meaning is explicitly defined, it is linked to other external data sets, and can in turn be linked to, from external data sets (Bizer et al. 2009). In our approach we use RDF links to link the subject in the namespace of the catalogue ontology to the object in the extension modules i.e. product category taxonomy such as eClass ontology. When we describe an actual resource instance within NIMBLE platform, we first create an instance under the *ResourceEntity* i.e. *Item* class in the catalogue ontology with properties that is general for all resources. In addition, we create another instance under the respective product category class in the product category taxonomy. The instance in the product category taxonomy is specified with the category associated properties. Afterwards, we link the instance in the catalogue ontology to the instance in the product category taxonomy. By doing so, the specification of the resource instance in the catalogue ontology is enriched with the instance descriptions in the product category taxonomy. Meanwhile, a resource instance can be linked to multiple instances in product category taxonomies, because an instance can simultaneously belong to multiple categories under different product category taxonomies (e.g. UNSPSC and eClass). In this case, the interpretation is that a resource instance linking to instances in two product category taxonomies has the specifications from both of them.

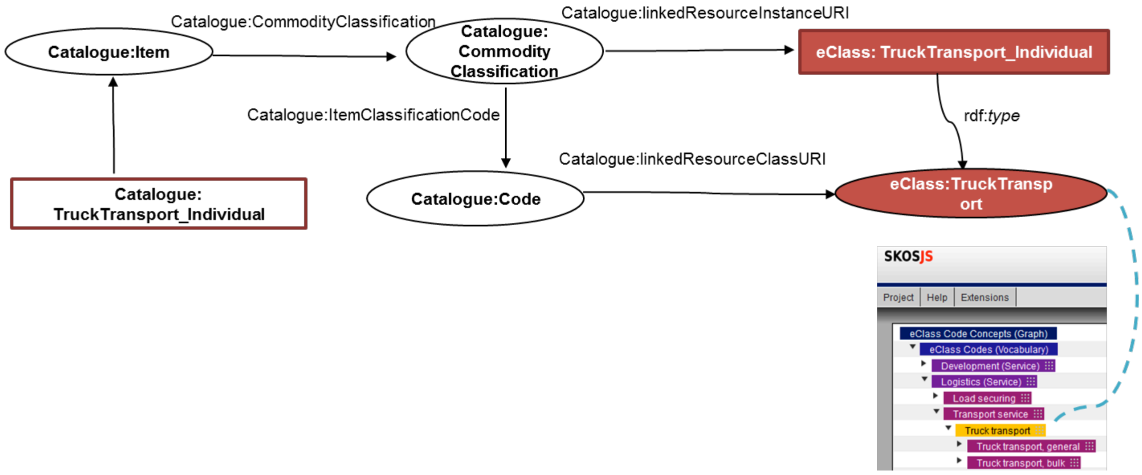

Figure 10 Example Extension with Linked Data Mechanisms

Figure 10 illustrates an example which has extended the catalogue ontology with eClass taxonomy under the linked data mechanisms. The namespace "eClass" refers to the module of the eClass taxonomy and the namespace "Catalogue" means the module of the catalogue ontology. The element *CommodityClassification* and *ItemClassificationCode* in the catalogue ontology are adopted from the UBL standard for interlinking the resource instance to its respective commodity classification information i.e. category information. The element *linkedResourceClassURI* is an RDF link to connect the Resource Instance "TruckTransport_Individual" in the catalogue ontology to the commodity classification i.e. category "TruckTransport" that the instance belongs to under eClass classification schema. The element *linkedResourceInstanceURI* is a RDF link to interlink the Resource Instance "TruckTransport_Individual" in the catalogue ontology to the instance in the eClass taxonomy. In this way, the specification of the resource instance in the catalogue ontology is enriched with instance specifications in the eClass taxonomy.

For the sake of providing a uniform and intuitive user experience for product publishing, which is also enhanced with external category taxonomies, we use a modified version of the second alternative above. In the current approach, each product category residing in external taxonomies are linked as a *Commodity Classification* to the *Item*. Actual RDF resource describing the category is accessed through the *linkedResourceClassURI* of the *Code* representation of the commodity classification. However, we do not create a separate individual for each category but a single for the product. We create *ItemProperties* for all properties specified by all categories classifying the product. Each *ItemProperty,* in turn, has a link to the corresponding property definition. While the Figure 11 summarizes our current approach, details about this ingestion mechanism is given in D3.2 Catalogue Ingestion and Semantic Annotation.

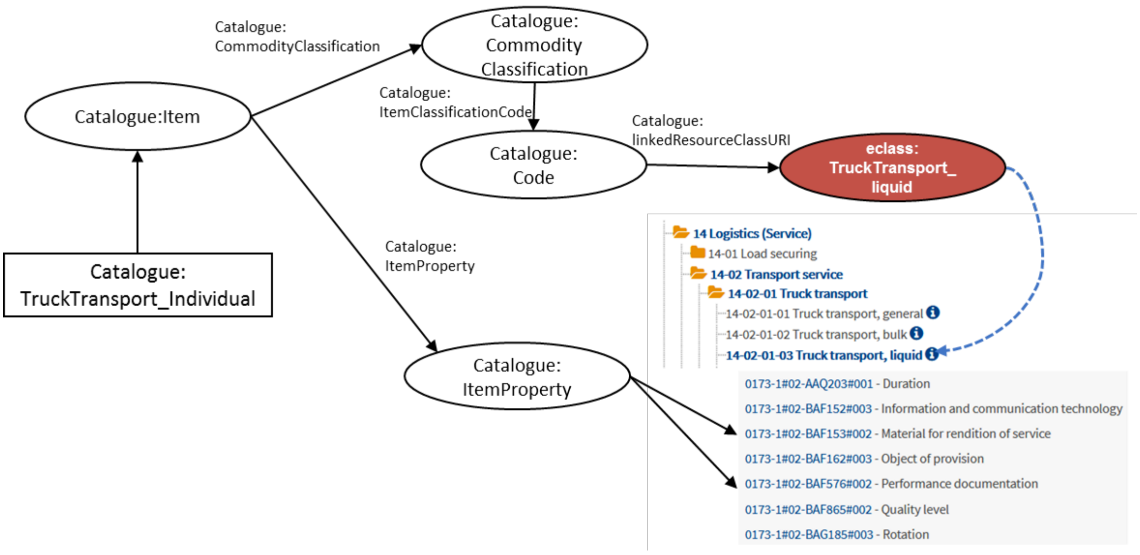

Figure 11 Current extension mechanism

3.2.2 Business Process Ontology

The business process ontology contains machine readable metadata for exchanging information in relation to business processes. *Process*, *Activity* and *Document* are basic elements in the ontology. Main usages of the business process ontology are:

- Organize roles and parties/persons for a process/activity
- Prepare a sequence of activities for a business process
- Help to exchange documents based to the right schema e.g. *UBL Order* in various activities

We consider the following aspects in the ontology: Behavioural Aspect, Organization Aspect, and Document Consumption Aspect. Definitions of the aspects are in line with the descriptions given by Axenath et al. (Axenath et al. 2005) The *Behavioural Aspect* concerns the order in which the corresponding activities are executed. The *Organization Aspect* is about the Business Roles and Business Entities that are involved in the Business Processes. The Document *Consumption Aspect* defines the Information that is exchanged among different actors in the business activities.

Some core competency questions for the business process ontology are listed in the following:

- Which company plays which role for a specific activity/process?
- Who creates the business process?
- Who is responsible for an activity?
- Which activities are performed in the process, in which sequence?
- Which business roles participate in the process and its activities?
- Which document schema is defined for some activity?
- Which actual document is exchanged in an activity?
- What is the real business process execution? Which activity is performed in the real business process execution?

3.2.2.1 Basic Elements

As stated above, the Business Process Model Aspect, Business Process Execution Aspect, Organization Aspect, and Document Consumption Aspect are the main aspects that are considered in the business process. It involves the following basic entities: BusinessEntity, BusinessRole, BusinessProcess, Activity, Document, DocumentSchema and Extensions. The principal relationships between the entities are depicted in Figure 12. A business entity can play various business roles in the business processes. Each business process contains usually more than one activity. And in each activity, information is exchanged via a document. The exchanged document should be compliant with a defined document schema, such as UBL order document schema and UBL invoice document schema, so that it can be semantically understood and processed by business entities. In the following, the seven concept categories are described:

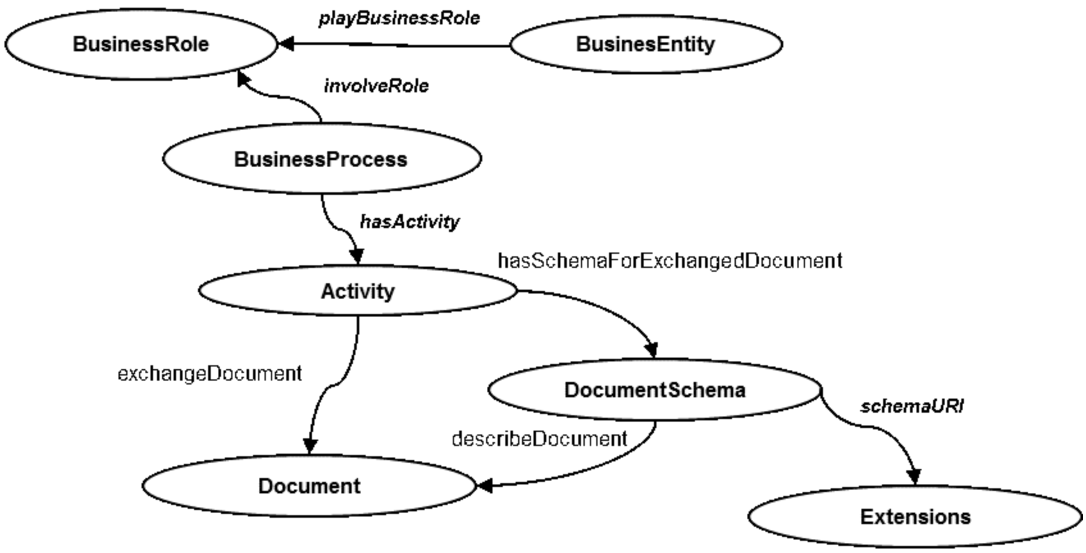

Figure 12 Relationship between main concepts in Business Process Ontology

Business Entity Concept

The Business Entity concept is reused from the catalogue ontology, and has therefor the same definition as described in the section 3.2.1.1.

Business Role Concept

Business Role is commonly used to describe interactions within a business process. When a business entity participates in a *Business Process*, it takes a Business Role. Each *Business Entity* can have multiple roles in different business processes. In order to have a more homogenous and understandable description of business roles in the NIMBLE platform, we have followed the categorization of supply chain participants described by Gibson et al.: *Direct Stakeholder* and *Facilitator* (Gibson et al. 2013). The categorization is based on the ownership stake of the participants in the product. According to their definition, *Direct Stakeholder* is *Business Entity* that owns the goods at various stages of the supply chain. This category of business roles includes End User, Manufacturer, Supplier, and Retailer etc. *Facilitator* is a *Business Entity* that supports the flow of materials, information, and money in the supply chain. They do not typically take title to the goods but play a critical role in the safe, efficient execution of supply chain activities. *Logistics Services Provider* is for example a typical business role under this category.

Business Process Concept

A *Business Process* is a set of structured *Activities* with logical behaviors that produce a specific service or product (von Rosing et al. 2015). The Activities are executed by some *Business Entities* within the supply chain. During the execution of Activities, each Business Entity takes a certain *Business Role*. A Business Process Model is a formal detailed description of the involved Activities, Business Roles, and Documents as well as rules governing the business process execution. A *Business Process Instance* is used to represent an actual process execution instance of a Business Process Model. Each Business Process Model has a set of allowed activities, and the Business Process Execution consists of activities that have actually been carried out. Actual Business Process Executions (i.e. the instances) can be different from each other, but they would all follow the rules defined in the Business Process Model.

Activity Concept

An *Activity* is a specification of action that is carried out to achieve a Business Process. It can either be atomic or compound. An atomic Activity cannot be split into further detailed activities, while a compound activity consists of several atomic activities. In each single atomic Activity, one and only one *Document* can be exchanged. The exchanged Document is either an atomic document or a composed document. A composed document is composed of other documents under some specified Document Schema or possibly no schema. Depending on the defined Business Process Model, some loops of activities are allowed to be executed repeatedly.

Document Concept

Documents are the digital resources that get exchanged between business entities in an activity. A purchase order is a typical document that gets exchanged between the buyer and the seller in a business activity. Documents are described by a Document Schema. For example, a purchase order can be an XML business document that follows a schema defined in the UBL standard.

Document Schema Concept

The *Document Schema* is a representation of metadata for (business) documents. It provides a means of interoperability between Business Entities. When a business process model is defined for the NIMBLE platform, Business Entities are able to generate document instances that will be exchanged in order to perform an activity. In order to specify which type of document should be exchanged within an activity, document type e.g. invoice document can be used. However, it is still not precise, as different standards or organizations can have their own definitions on the same document type with respect to document structure, file formats etc., due to differences on understanding and usage domains and so on. To avoid such semantic ambiguities, the concept of *Document Schema* is suggested. With this schema, the metadata of Documents can be captured, so that the exchanged actual documents can be semantically understood and handled by machines. The entities in the category of *Document Schema* are categorized into different types according to the UBL standard. For example, *InvoiceDocumentSchema* is used to represent the metadata of invoice documents. Furthermore, the *Document Schema* can be further extended with the schemas that have been defined by other standards or organizations, for example, through the Moda-ML standard.

Extensions

Extensions is not an ontological concept such as *Document* for the share of common semantic meaning. Its role is to serve as an open interface slot for structuring of further document schemas. The reason for keeping this *Extensions* slot is because we need extensibility and usability of the data model for diverse use cases. In different use cases or supply chains, many documents under different document schemas are exchanged. The document schemas can be from general international standards e.g. UBL, or from some domain specific standards or even defined by some major companies. It is typically hard to manage all different kinds of document schemas in various domains within a single ontology. In the context of the business process ontology, the *Extension* is mainly used to maintain the linkage to the document schemas that are defined by standards or business entities. More details on how to extend the business process ontology is given in the section 3.2.2.4.

3.2.2.2 Reuse of ontological resource: Moda-ML Business Process Ontology

The Moda-ML Business Process Ontology provides a description of the business processes in the textile domain. This description is formalised in an OWL ontology, available online, that is based on 4 main concepts: Process, Actor, Activity, Document (a very brief, high-level visualization of the Moda-ML business process ontology is depicted in Figure 13). This set of concepts is connected with a set of Object Properties that allows to create a description of a business process, and exploits a wide-ranging categorization of the textile domain (the ontology contains about 102 classes for document description). . With these terms defined, the Moda-ML Business Process Ontology offers significant added value to NIMBLE for the proper creation and customisation of business processes in the textile domain. In this sense, the Moda-ML Business Process Ontology can provide components, or modules specific for the textile sector for the Nimble platform These can be reused and connected to compose domain specific business processes and their ontological description.

The ontology is generated from the specification of the Moda-ML initiative (http://www.moda-ml.org/moda-ml.asp?lingua=en&) that provides business documents and defines business scenarios for the textile domain, and is maintained always aligned with the development of the specification.

The Moda-ML Business Process Ontology can be connected with an ontology of products (a catalogue ontology). The default product ontology in Moda-ML is quite simple, but it is possible to connect other more detailed catalogue ontologies (defined both by standardisation activities or private enterprises) in order to provide ontological definition of specific business processes related to specific classes of products.

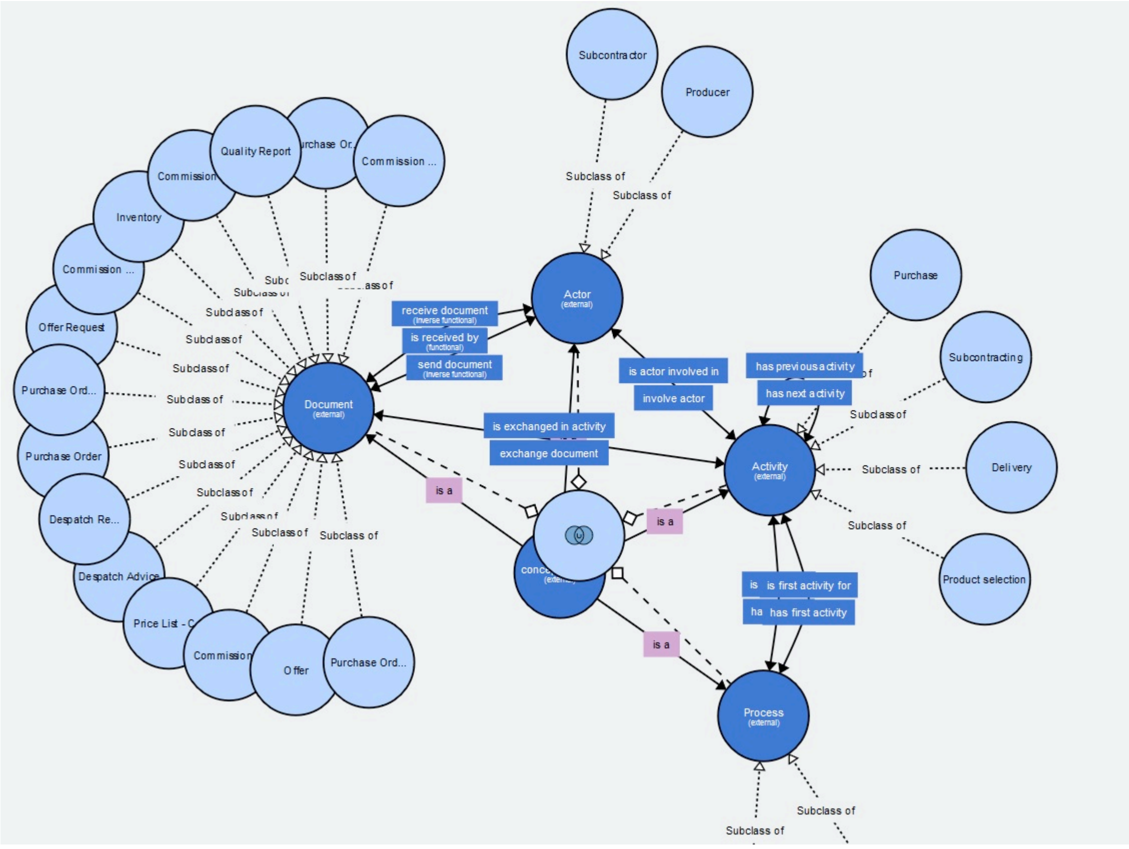

Figure 8 High-Level schema of the Moda-ML Business Process Ontology

3.2.2.3 Ontology Schema

This sub-section provides a high-level visualization of the ontology schema. It follows the key concepts and properties listed in sub-section 3.2.2.1 Basic Elements and definitions in sub-section 3.2.2.2 Moda-ML Business Process Ontology. Many concepts and properties such as Process, Activity, Document and hasActivity are reused from the Moda-ML Business Process Ontology. The concept Document Schema is derived from the concept DataSchema in the DRM (Data Reference Model) ontology [12] for representation of metadata of exchanged documents. In line with the intended main usage of the business process ontology in the NIMBLE platform, our focus is on structuring the metadata of exchanged documents to guarantee that the right document is exchanged in the right activity and to ensure that the exchanged document is understandable by different actors. A high-level visualization of the business process ontology is illustrated in Figure 14. For visibility reasons, only important concepts and object properties are shown in the figure. *Extensions* are described in sub-section 3.2.2.1 The role and usage of *Extensions* will be described separately in sub-section 3.2.2.4.

[12] http://vocab.data.gov/def/drm

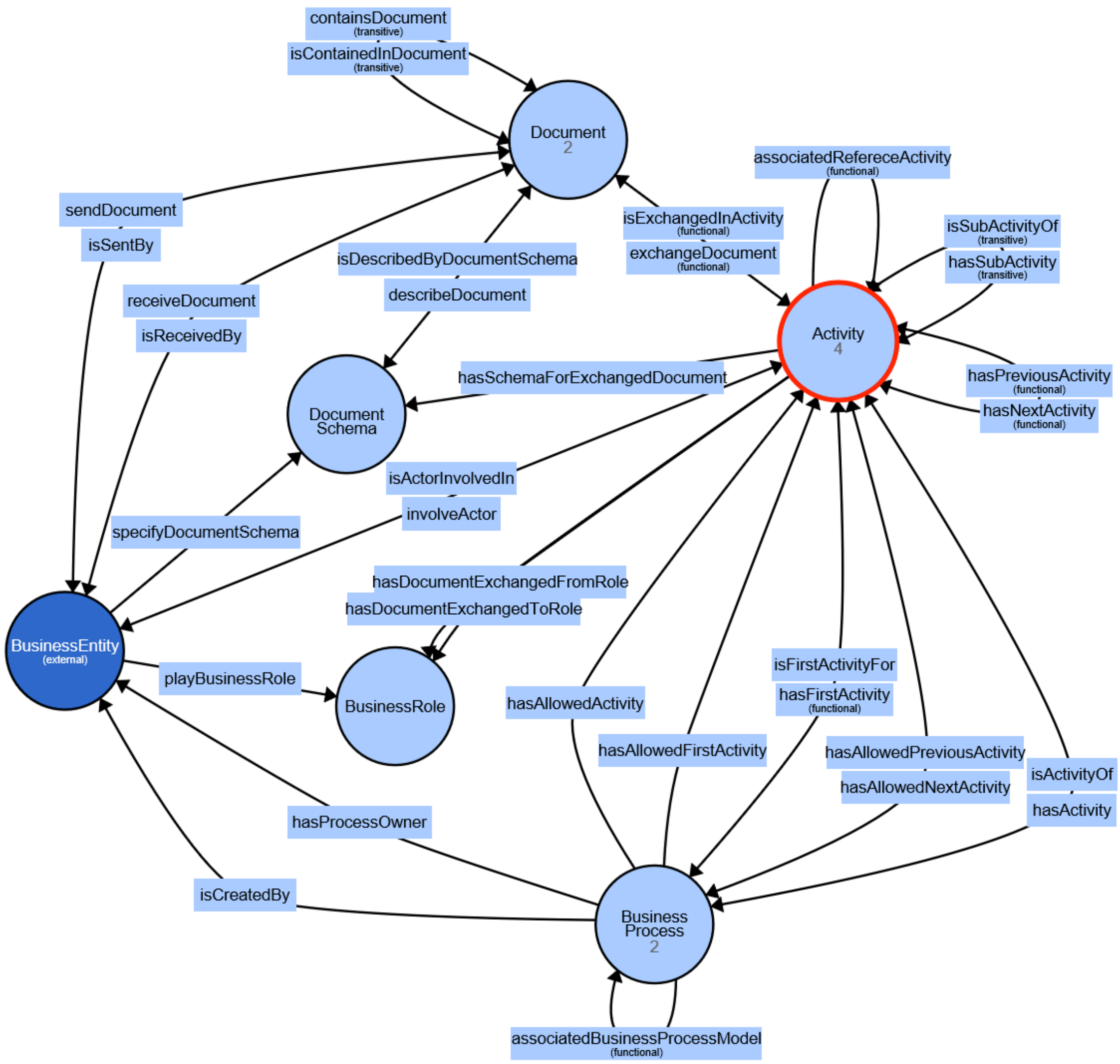

Figure 9 High-Level schema of the business process ontology model

There are three aspects that are covered in this ontology: Behavioural Aspect, Organization Aspect, and Document Consumption Aspect. In order to better understand the business process ontology as well as the aspects in the ontology, we will present the ontology shown in the above figure from these three different angles.

The Behavioural Aspect of the business process ontology is about the activities as well as the order of the activities in that business process. Business Process *Model* and Business Process *Execution* are both children classes of the concept *Business Process* and are handled semantically in different ways. A Business Process Model is a formal definition of a process, while a Business Process Execution Instance represents an actual process execution. The Business Process Execution Instance as well as Activities in the process execution instance can be linked to the respective Business Process Model and Activity with object properties "*associatedBusinessProcessModel*" and "*associatedReferenceActivity*".

In principle, the activities that belongs to Business Process Model are linked with object property "*hasAllowed*Activity*"; and the actually carried out activities that belongs to Business

Process Execution Instance are linked with object property "*has*Activity*". The character "*" here is a placeholder for none or one word. For example, the first activity in the business process execution instance is represented with the object property "*hasFirstActivty*", which is a functional property, as each business process execution instance can only have one single first activity. Each activity can consist some sub activities represented in the object property "*hasSubActivity*". To better illustrate the given concepts and properties, an example of a generic business transaction process is given in Figure 15. The given business transaction process model *hasAllowedActivity* "Receive Payment" "Acceptance" and "Claim" etc. The activity "Acceptance" and "Claim" are both potential next activity of the activity "Receive Payment". This kind of relationship can be represented with the object property "*hasAllowedNextActivity*". In an actual business process execution instance, the activity "Receive Payment" has however only one next activity: either "Acceptance" or "Claim". In the Business Process Execution Instance, the activity "Receive Payment" can then use the object property "*hasNextActivity*" to link the next activity: either "Acceptance" or "Claim".

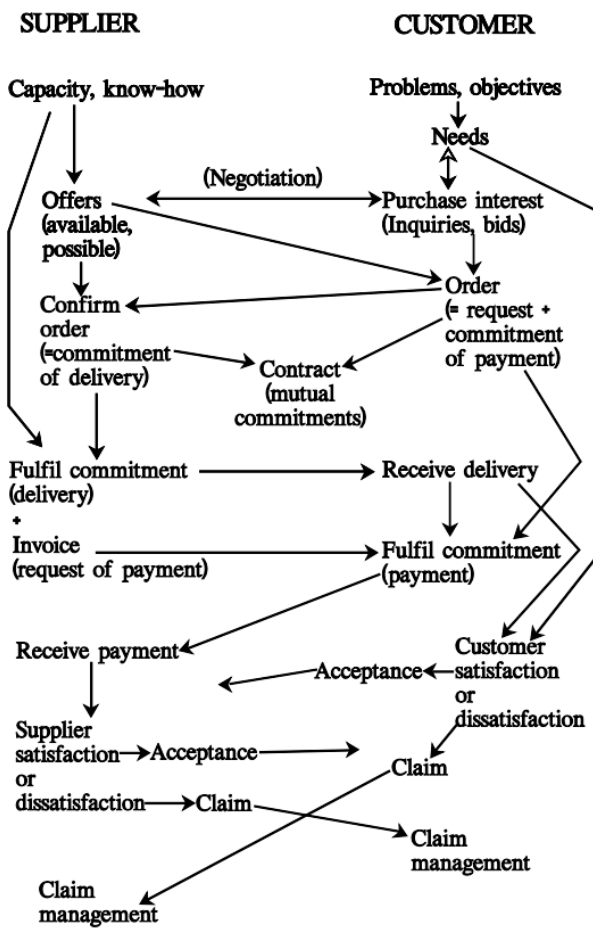

Figure 10 Illustration of generic e-commerce business transaction process (Goldkuhl 1996)

Organization Aspect is about the organizations that are involved in Business Processes. *Business Entity* and *Business Role* are the most relevant concepts in this aspect. The concept of

Business Entity is imported from the catalogue ontology, which is used to represent the information related to organizations. Business Role is the role that a Business Entity takes when it is involved in a particular Business Process. When a Business Process is created by a Business Entity, the process has an owner, which is represented in the property “hasProcessOwner”. Meanwhile, Business Entity is involved in different activities and plays various Business Roles in the activities. This kind of involvement is represented in the object property “isActorInvolvedIn”, and the role of Business Entity is linked with object property “playsBusinessRole”.

Document Consumption Aspect defines the document i.e. information that is exchanged among different activities. The metadata of the exchanged Document is specified by the Document Schema, so that the exchanged Document is semantically understandable and machine readable. The relationship between Document and Activity can be represented in the object property “exchangedDocument”. And the relationship between Document Schema and Document can be specified with the object property “isDescribedByDocumentSchema”. A Document can contain further documents with the object property “containsDocument”. When a Business Process Model is defined, no Document is actually exchanged in activities. At the definition time, the Document Schema is linked to activities with the object property “hasSchemaForExchangedDocument”, specifying which kind of Document can be exchanged in the activity between Business Entities. Since many standards, e.g. UBL or STEP, have already defined many different document schemas, and it is impractical to cover every document schema in this single ontology for each exchanged document in different business processes, we intend to make this business process ontology highly extensible to reuse the document schemas that have been defined by others. Details are described in the sub-section 3.2.2.4 Extensibility.

3.2.2.4 Extensibility

Similar to the extensions in the catalogue ontology, the business process ontology can be extended in two ways: inheritance mechanisms and linked data mechanisms. Most concepts in the business process ontology including Business Process, Activity and Document Schema can be extended with the inheritance mechanisms. Based on the UBL standard, we have listed some general business process concepts such as “OrderingProcess” or “ProductionProcess” in the ontology. But they may not have covered yet, all different kinds of business processes in B2B e-commerce. In addition, the defined business process concepts are not domain specific. In this case, concepts defined in other domain specific business process ontologies can be integrated as a sub-class of the concept in the current business process ontology for the purpose of extension. For example, the “SubcontractedFabricManufacturing” process in the Moda-ML business process ontology can extend the “ProductionProcess” in the current ontology by using the inheritance mechanisms.

Through a linked data mechanism we can reuse document schemas that were defined in other standards or by other business entities. The current business process ontology lists some document schemas based on the UBL standard. However, in different use cases or supply chains, many documents complying with different document schemas are exchanged. The document schema can be possibly from UBL, but can also come from some other domain specific standard e.g. Moda-ML in the textile domain, or can even be defined by some major companies. In order to reuse the document schemas that were defined by others, ensure that exchanged documents are under the right document schema and understandable by different actors, linked data mechanisms can be used to connect the external documents. As depicted in Figure 12, this kind of RDF linkage is achieved through the data property "schemaURI", which connects a document schema instance within the business process ontology to the URI of the external document schemas within other ontologies or standards. For example, in a process of ordering garment accessories, an order document has to be sent to a supplier, who has adopted the Moda-ML standard in textile domain. However, systems from the supplier can only understand the order document that was structured under the document schema ACSOrder [13] (a document schema also defined in the Moda-ML standard). In this case, we can extend the existing document schemas by creating a RDF linkage to the document schema ACSOrder with the data property "schemaURI", so that customers can prepare an order document that is understandable by the supplier. Besides the schemas from standards such as UBL or Moda-ML, it is also allowed to extend Document Schema with the schemas that were personalized by other business entities. In this way, a Business Entity has also the possibility to customize the metadata of exchanged documents.

3.2.3 Extension Modules

3.2.3.1 eClass Taxonomy

eClass is an ISO/IEC compliant industry standard for cross-industry product and service classification. Although we use its English version, it is an international standard with translations to several other languages. The latest version (10.0.1) of the taxonomy was released in February 2017 with 41,000 product categories and 17,000 properties [14]. In line with NIMBLE's objectives, the eClass taxonomy contains resources for both tangible products like furniture fittings and intangible services like rail transportation. The complete hierarchy can be browsed at their website [15]. More details about eClass, its elements and structure are given in "*D3.2 Catalogue Ingestion and Semantic Annotation*".

In order to use the eClass terminology with the Semantic Metadata Repository, it is required to transform the provided CSV files into RDF triples and to upload the triples to the Semantic Metadata Repository.

eClass is organised as a set of hierarchical concept classes, where each concept class has a preferred name, a definition and assigned properties. This makes it possible to also use the SKOS ontology (skos:ConceptScheme, skos:Concept) that allows for the organization of concepts in vocabularies, the coverage of hierarchical structures (skos:broader, skos:narrower) and other semantic relationships between eClass concepts.

Figure 16 shows the structure as published by the organization maintaining eClass. Since the maintainers of eClass only provide plain CSV tables containing the data, it is required to

[13] http://www.moda-ml.org/moda-ml/repository/schema/Draft/ACSOrder.xsd
[14] http://wiki.eclass.eu/wiki/Category:Products
[15] http://www.eclasscontent.com/

transform the data into RDF. For this step, OpenRefine[16] (formerly Google Refine) in combination with the RDF extension for exporting processed data into RDF, is used.

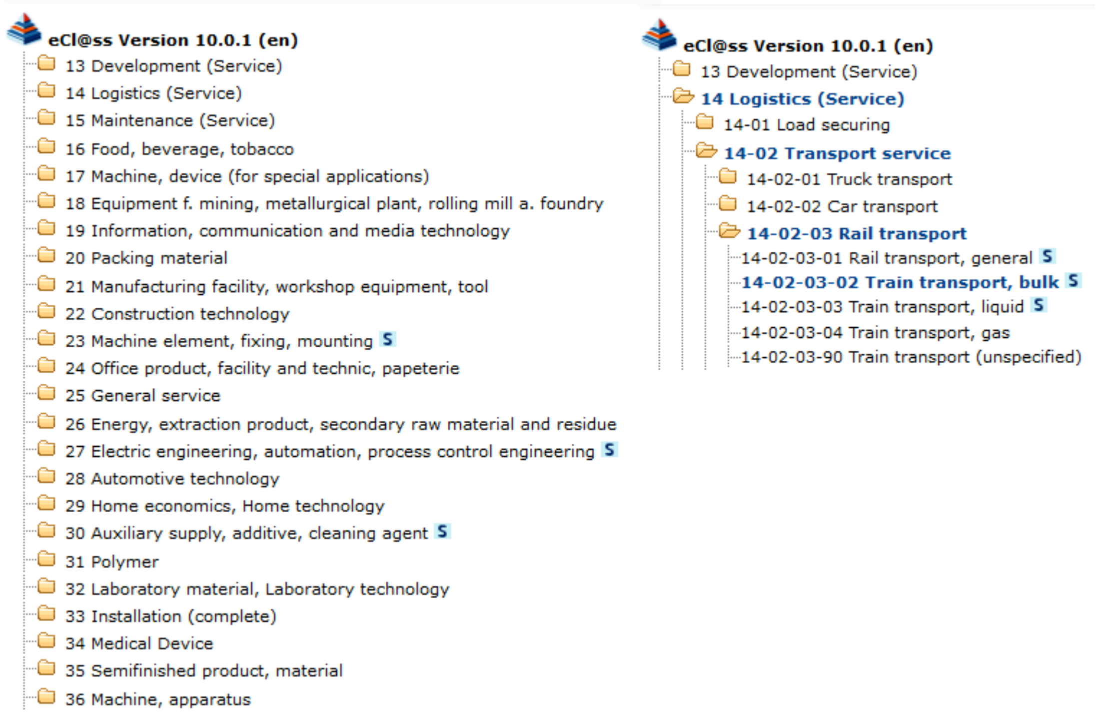

Figure 11 eClass Concepts (top level elements); Excerpt from concept hierarchy[17]

As a result, the corresponding RDF triples expressing the SKOS structure are created and uploaded to the Semantic Metadata Repository. As an example, the transformed RDF data for the term *rail transport* is given in Turtle-format, in Listing **1**.

```
!!Namespace defintions omitted …

<http://www.nimble-project.org/resource/eclass/14020300> a skos:Concept ,
               <http://www.nimble-project.org/onto/eclass#CodeConcept> ;
        skos:prefLabel "Rail transport"@en ;
        dcterms:publisher "0173-1" ;
        dcterms:identifier "0173-1#01-ADT268#003" ;
        dcterms:hasVersion "003" ;
        dcterms:issued "2012-11-27"^^xsd:date ;
        dcterms:language "en" ;
        skos:inScheme <http://www.nimble-project.org/resource/eclass/CodeHierarchy> ;
        skos:narrower <http://www.nimble-project.org/resource/eclass/14020301> ,
               <http://www.nimble-project.org/resource/eclass/14020302> ,
               <http://www.nimble-project.org/resource/eclass/14020303> ,
               <http://www.nimble-project.org/resource/eclass/14020304> ,
               <http://www.nimble-project.org/resource/eclass/14020390> ;
        skos:broader <http://www.nimble-project.org/resource/eclass/14020000> .
```

Listing 1 eClass (SKOS) Concept exported in Turtle format

After transformation, every imported eClass concept has the skos:Concept type and is assigned to a vocabulary where all the eClass concepts are organized. Furthermore, the semantic relations

[16] http://openrefine.org/
[17] http://www.eclasscontent.com/index.php?version=10_0_1&language=en

denote broader (parent element) and narrower (child elements) relationships. As a result, the eClass data may be viewed/edited in Marmotta's SKOS editor (see Figure 17) where it is possible to provide labels in various languages and to add more knowledge (alternate, hidden labels, other semantic relationships) to the stored eClass data structure.

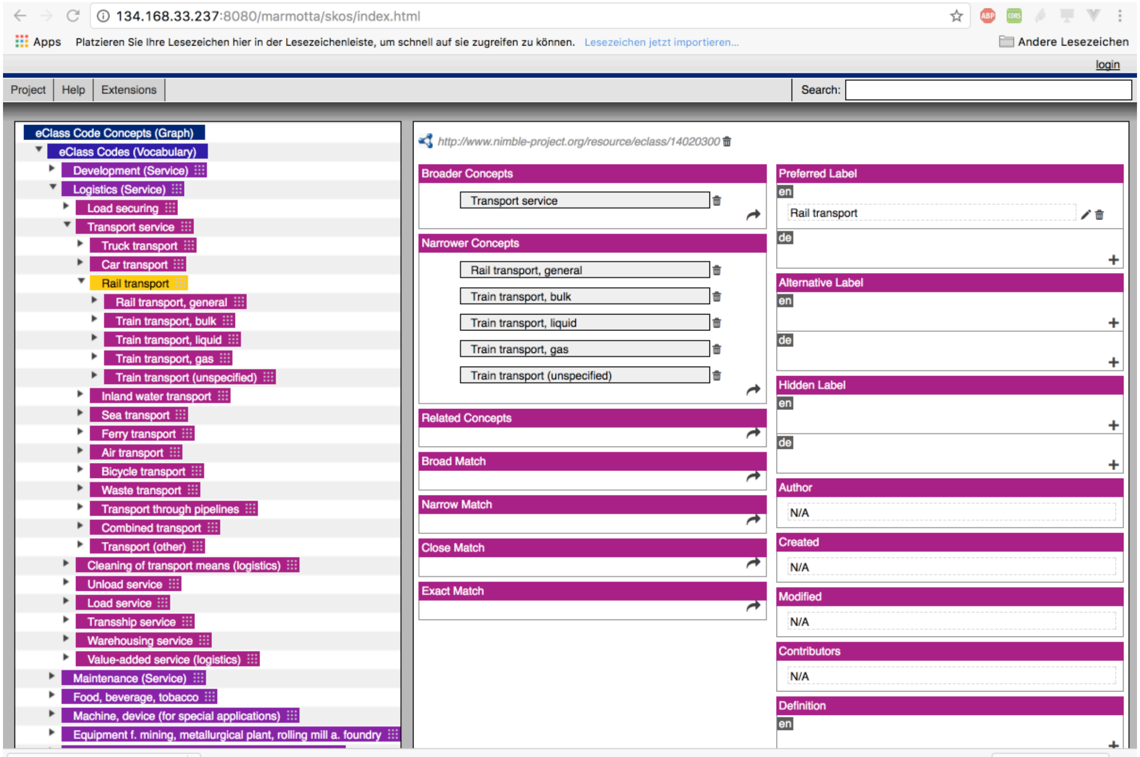

Figure 12 eClass - SKOS editor

Once transformed to RDF, the eClass taxonomy may be used as a categorization system and also to semantically enrich product descriptions which additional information when searching for products.

3.2.3.2 Furniture Sector Taxonomy

The Furniture Sector Taxonomy is the result of the merge of a previous version of the furniture taxonomy - which originally focused on industrial processes and machinery in the furniture industry – and the furniture ontology – focused on the product categorisation and based on the funStep ISO standard.

Main entities included in the Furniture Sector Taxonomy are:

- Processes involved in the furniture industry, covering from wood processing until product delivery.
- Machines and other related equipment used in furniture manufacturing.
- Techniques used by industrial machines.
- Materials and substances used during the manufacturing process.
- Components and assembly hardware as elements that are part of the furniture.
- Product catalogue including finished products, complements and compositions organized according to a wide categorization.

The product catalogue subset of the taxonomy has its origins in the Furniture Ontology, which is based on the international standard for the information data exchange FunStep ISO 10303-236 (Industrial automation systems and integration -- Product data representation and exchange -- Part 236: Application protocol: Furniture catalogue and interior design) also known as funStep.

Focused on the furniture and wood cluster, this ontology was developed to define a common vocabulary expressed in a formal way and capturing the most widely used concepts inside the furniture and furniture-related industry, including properties and relationships between them.

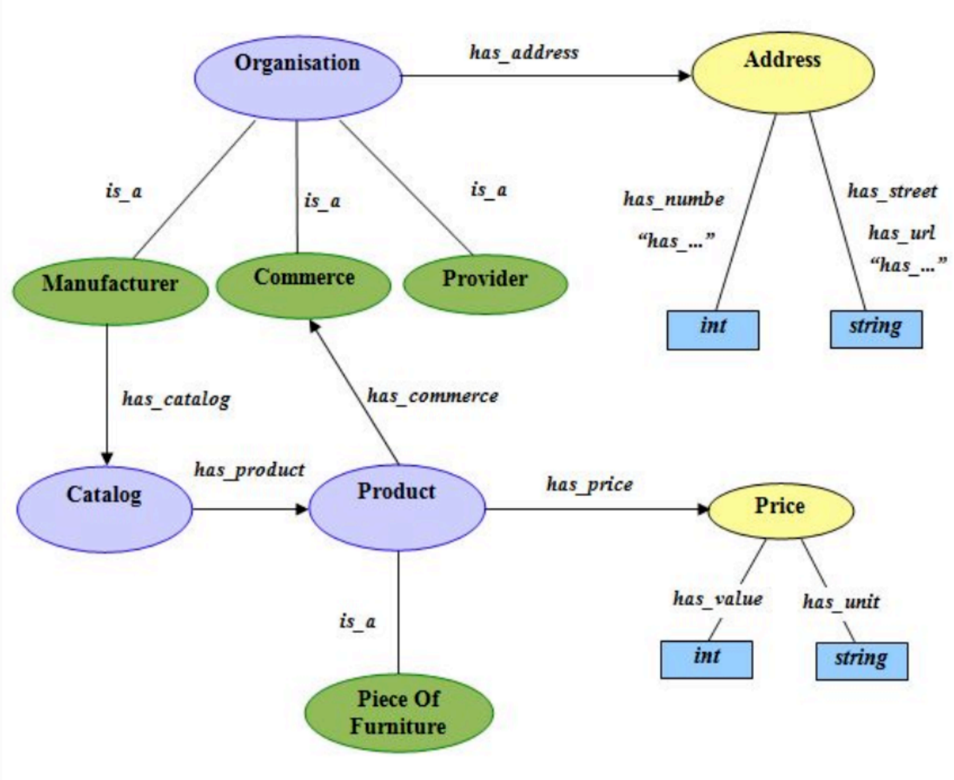

Figure 13 Excerpt of relationships between entities based on the funStep ISO standard

The Figure 18 illustrates some relevant properties such as Organisation, Catalogue and Product from the Furniture Ontology based on the funStep ISO standard, as well as the relationships between them and some relevant properties. Elements in green represent descendant classes of these main entities, while the properties are represented as yellow nodes with its corresponding basic data properties in blue. The labelled arrows in the diagram represent the properties, which define a relationship in the ontology between the entities. i.e.: a Manufacturer is a kind of Organisation which has Catalogues composed of Products with their corresponding Price.

As commented earlier, the Furniture Sector Taxonomy integrated in NIMBLE takes the Furniture Ontology as a key pillar maintaining the basics of funStep but also incorporating additional resources from the industrial furniture sector (processes, machinery, materials, etc.) generating a wider categorisation.

Figure 19 below shows the breakdown of the Process entity in the Furniture Sector Taxonomy revealing the branch of subclasses of *process* reaching the finishing operations. Also the relationships between Processes, Machines and Techniques can be noticed. The taxonomy also manages specific Object and Data properties for the represented entities that connect to relationships between them. For example, processes are performed by machines and machines are based on manufacturing techniques.

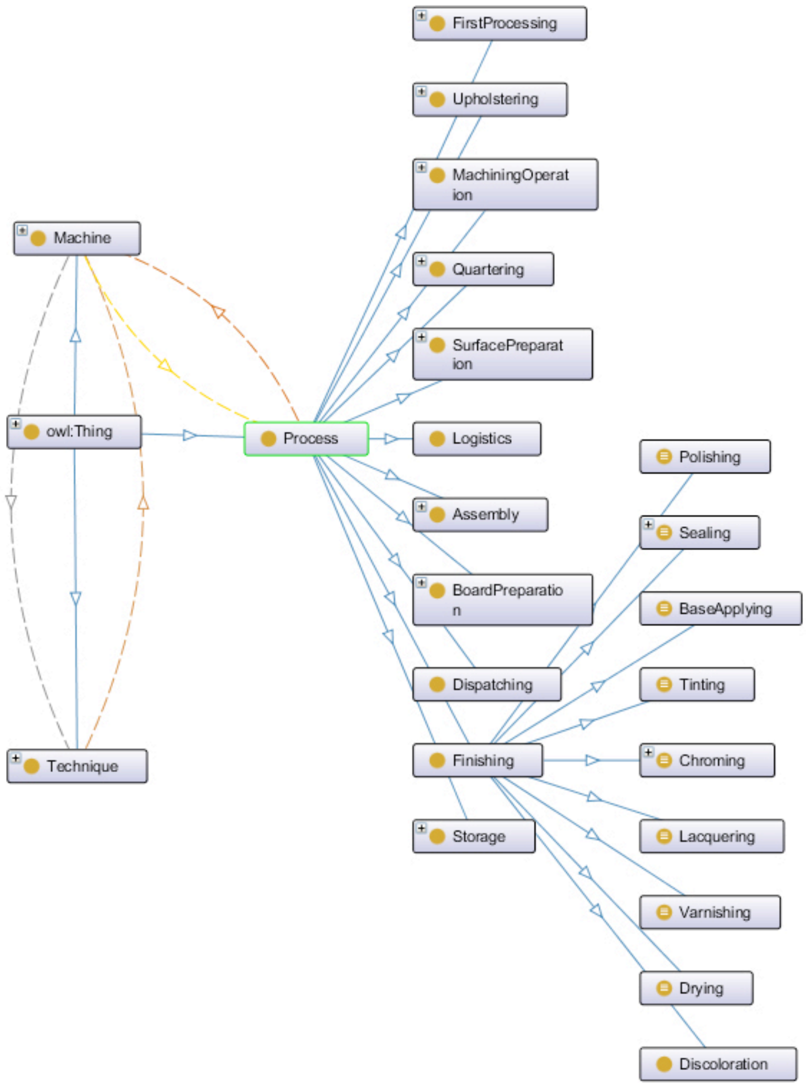

Figure 14 Detail of the activities involved in the finishing process represented in the taxonomy

The Furniture Sector Taxonomy is based on the knowledge of AIDIMME and industrial partners in a Spanish furniture cluster. Their knowledge also covers regulations (UNE, EN, ISO) and the taxonomy captures the vocabulary of the sector based on interviews, books, articles and regulations.

3.2.3.3 Textile Taxonomy

The textile taxonomy has been derived from an ontology based on the eBIZ/Moda-ML business standard, which was developed by the CEN initiative.

The eBIZ/Moda-ML standard focuses on the collaboration aspect of textile/clothing production and covers processes that involve two or more companies or actors. For this reason, the elements that can be derived from this standard are more related to these collaboration aspects than the classification/categorization of goods. As a result, the OntoMODA ontology, derived from the standard, describes and categorizes different business processes, but is poor on the categorization of the products. However, the richness of different types of data in the Moda-ML dictionary related to textile/clothing products, can be used to develop a catalogue of textile goods. Thus, a taxonomy of concepts can be derived from the following resources:

- **Business Processes**: activities organised by different collaboration steps linked with relevant documents for the specific interaction;

- **Product catalogues**: collection of yarn, fabric, garment and accessory products.

At the moment, the catalogues are a plain list of goods (for example a list of yarns) with different properties (both commercial information and technical datasheet relevant for all the commercial aspects). These properties are described in a very detailed way taking into account almost all the properties that characterize a textile/clothing product.

However, real product catalogues in the textile/clothing sector can be very complex objects containing, for example, not only the images of the yarns, but also real samples that differ in colour, tactile properties and/or in production processes. In the textile/clothing sector, buyers want to touch the product, experience the colour and texture in a very precise way and select the goods directly.

For this reason, some examples of eBIZ/Moda-ML catalogues for textiles and yarns have so far been produced to show their potential, and these are published on the web site of the Moda-ML initiative. However, companies in the textile sector have so far, not adopted the Moda-ML catalogues for the reasons described above. Another running project is testing the use of the eBiz/Moda-ML catalogues for data exchange among different software tools, thus opening at the future possibility for the adoption of such documents in business collaborations. In the validation phase of NIMBLE, we will investigate ways in which the realization of electronic textile catalogues for yarn and fabrics may be boosted through the NIMBLE platform.

3.2.3.4 Semantic data model of the product avatar

As shown in Figure 20, the entities within the three blue areas are designed to capture all data relating to the lifecycle phases of a product. The product avatar can extend items in the catalogue ontology with detailed specifications of product life cycle data. There are three macro areas:

1. Beginning of Life (BOL). BOL is the first stage of the product life cycle. It includes the phases of product development, production and distribution and data generated in these phases can be collected and connected to individual products or to aggregations.
2. Middle of Life (MOL). MOL refers to the phases of product usage, repair and maintenance. In this stage, the product usage data can be collected through Internet of things (IOT) technologies e.g. sensors and RFID tags. In addition, consumer data and service data can also be linked to products.
3. End of Life (EOL). EOL covers primarily the phases of reuse and disposal, associated to products.

Detailed descriptions on the semantic data model of the product avatar can be found in the deliverable D3.6 “Lifecycle Data Management and Analytics”.

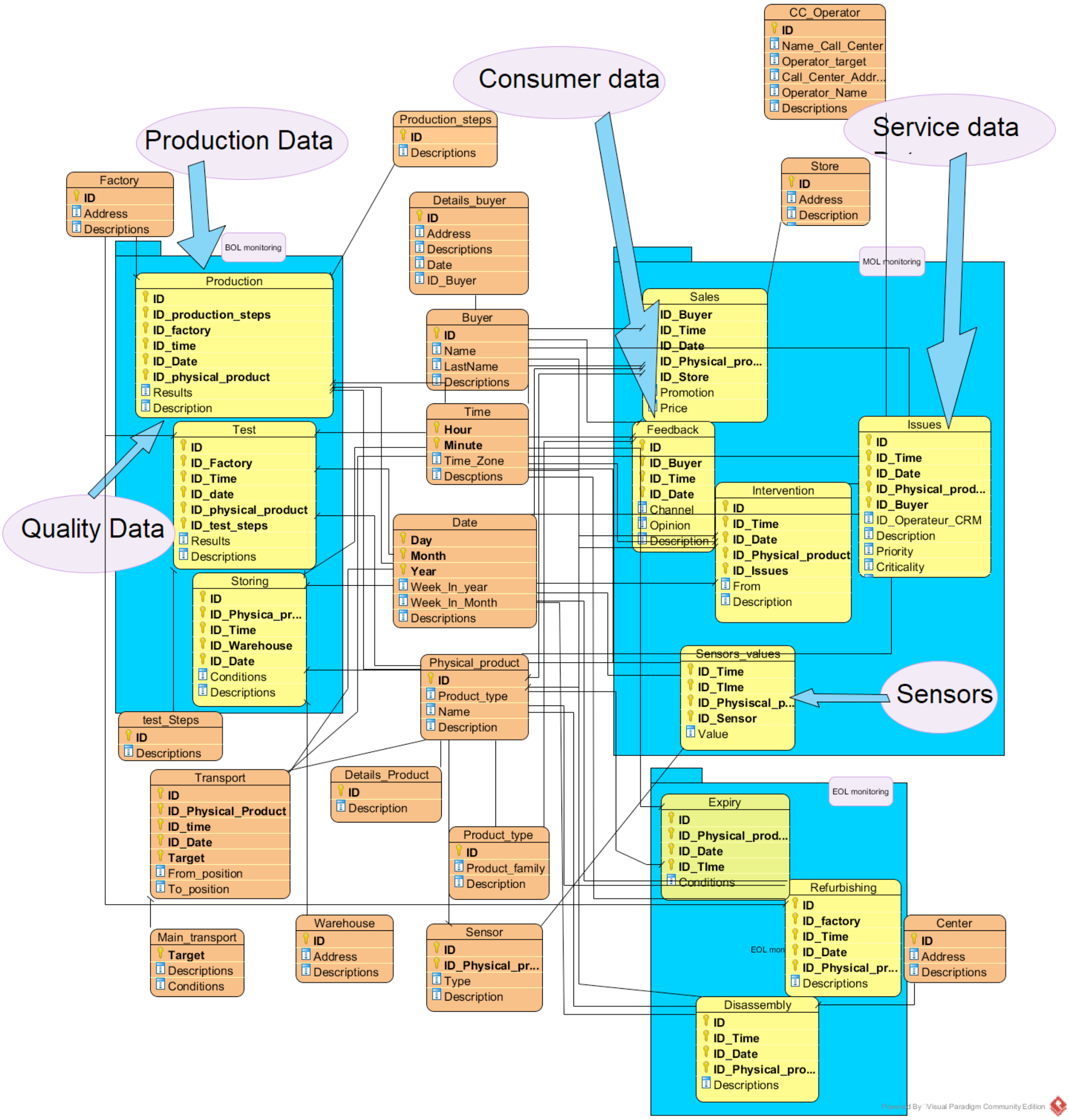

Figure 15 Semantic data model of product avatar

3.3 Relation to standards

This section describes how the ontology network in NIMBLE platform relates to existing standards.

- **UBL standard**. As mentioned previously, we adopt UBL as the base data model of the Catalogue Registry. Besides, we utilize this standard for supply chain related documents (e.g. Order, Request for Quotation, etc.) to be communicated between trading partners through business processes.

- **eCl@ss standard**. eCl@ss is an ISO/IEC compliant industry standard for cross-industry product and service classification. In the ontology network, the eCl@ss taxonomies have been transformed into RDF triples expressing the SKOS structure for the detailed specification of products and services in general.

- **FunStep ISO 10303-236** (Industrial automation systems and integration -- Product data representation and exchange -- Part 236: Application protocol: Furniture catalog and interior design). The Furniture Sector Taxonomy is the result of the merge of an old version of the furniture *taxonomy* - which focused on industrial processes and machinery in furniture industry – and the furniture *ontology* – focused on product categorisation and based on the funStep ISO standard. It has been integrated into the ontology network for the detailed specification of products and services in the furniture industry.

- **Moda-ML standard**. Moda-ML is an European standard for data exchange for the Textile and Clothing sector. In this ontology network, the Moda-ML is partly integrated in two ways: 1) the taxonomy for the textile domain has been adopted and transformed for the detailed description of products and services in this sector; 2) the data schemas for supporting data exchange in the Textile and Clothing sector have been integrated for supporting B2B e-commerce business processes in the NIMBLE platform.

Not limited to the aforementioned standards, ontologies and data schemas based on other standards can be integrated likewise, as extensions, and thus added to the ontology network.

3.4 Semantic Metadata Repository

As explained in previous sections and also introduced in *D3.2 Catalogue Ingestion and Semantic Annotation*, NIMBLE relies on semantically enriched metadata to store product catalogue information, to drive a (free-text) search engine capable of detecting features of stored products automatically and furthermore, to enable data exchange based on Linked Data principles (Berners-Lee 2006, Bizer et. al 2009).

Therefore, the Semantic Metadata repository is based on Apache Marmotta[18] (a.k.a Marmotta), an implementation of the emerging LDP[19] standard. Marmotta provides features such as:

- Read/Write Linked Data server for the Java Enterprise Stack
- Pluggable RDF triple stores based on the Eclipse RDF4J (formerly Sesame) framework
    - Custom Triple stores built on top of RDBMS, with transactions, versioning and rule-based reasoning support.
- LDP, SPARQL and LDPath Querying

[18] http://marmotta.apache.org
[19] https://www.w3.org/TR/ldp/

- Transparent Linked Data Caching
- Integrated Security Mechanisms

An overview of the functionality afforded by Apache Marmotta is given in Figure 21. The distinct pillars of the given overview represent functionality used by the Semantic Metadata Repository.

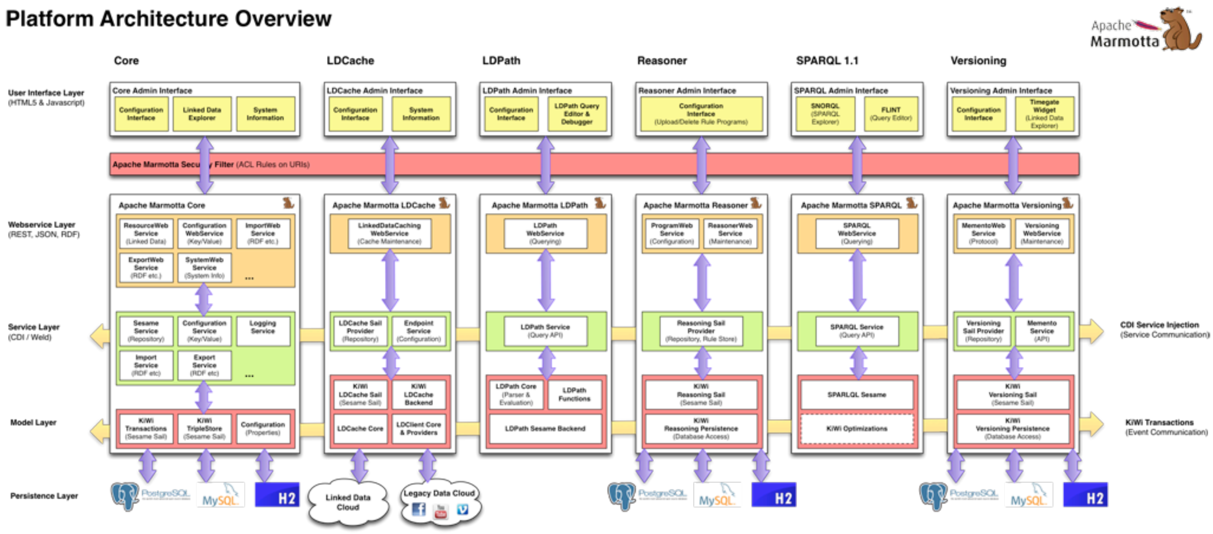

Figure 16 Apache Marmotta – Platform Architecture Overview

Apache Marmotta provides methods to maintain data in the form of RDF triples. It is easily possible to query the data by means of SPARQL or LDPath queries and also to identify new information by using the built-in rule-based reasoner.

For the NIMBLE Semantic Metadata Repository we use the following functionality of Apache Marmotta:

- **SPARQL endpoint**: A SPARQL 1.1 endpoint is provided by Marmotta and can be used for querying and updating metadata. The SPARQL[20] endpoint is accessible in the browser through the Apache Marmotta admin interface, and as webservice conforming to the SPARQL protocol at /marmotta/sparql/select and /marmotta/sparql/update. Using SPARQL is the most powerful method of interacting with content items and their metadata, but might be too complex for simple use cases.
- **Transaction System**: data manipulation is secured by a transactional system that ensures data integrity.
- **LDPath**: LDPath is a simple path-based query language, similar to XPath. In combination with the transparent LDCache it is possible to integrate and query data located on remote Linked Data servers.
- **Reasoner**: The reasoning capabilities are used to identify new facts based on existing information.

[20] http://www.w3.org/TR/sparql11-query/

During development, the NIMBLE platform employs KiWi[21] as its triple store implementation. This triple store uses a Relational Database Management System, in particular PostgreSQL[22], to manage the data. However, there are also big-data back-ends available, based on Apache Hadoop. These back-ends will be used for production.

3.4.1 Marmotta Search

The general architecture of Apacha Marmotta as given in Figure 21 is extensible. Each of the pillars represents a component adding functionality to the running system. Marmotta supports the creation of new functional components and allows seamless service lookup and ingestion at the Service Layer (CDI/Weld). Any added component therefore has access to the triples and also to the transaction services.

Marmotta Search is a functional component adding semantic search capabilities to the Semantic Metadata repository. For this, the search component

- provides a (faceted) search engine based on Apache Solr[23] (a.k.a. Solr),
- allows the definition of search definitions, e.g. uses LDPath statements to obtain relevant information from the triple store,
- creates the search index based on the search definitions and finally
- keeps the search engine in sync with the triple store. E.g. the index is automatically updated whenever the underlying data changes.

It is important to note that this approach applies its semantic annotations to the search index during indexing time, not at query time. As an example, SKOS based terminologies such as a product configuration allow the definition of preferred labels, alternate labels and also hidden labels. A search definition might add all different labels for indexing to allow retrieval of products based on all distinct labels. However, the labels are retrieved whenever a product is indexed. By default, label changes after indexing are not reflected in the search index until re-indexing.

More details of the search functionality for the Semantic Metadata Repository as well as an example of a search configuration for product catalogues are given in *D3.2 "Catalogue Ingestion and Semantic Annotation"*.

4 Conclusion and Outlook

This deliverable provides a generalized description of the semantic modelling of manufacturing collaboration assets for the NIMBLE B2B platform. The semantic models support the discovery

[21] http://marmotta.apache.org/kiwi/
[22] http://www.postgresql.org/
[23] http://lucene.apache.org/solr/

of suitable supply chain partners and enable business to business transactions on the NIMBLE platform by specifying various types of tangible resources and intangible resources. It analyses the use case scenarios and derives from the scenarios the requirements on semantic modelling. Based on the requirements, it defines two core ontology modules in the semantic modelling: catalogue ontology and business process ontology. Further ontology modules for domain specific use case scenarios can be added to extend the core ontology modules. They are described in the deliverable module by module. For each core module, the main concepts are specified, and the respective ontology schema is presented. These ontology modules are interlinked into the NIMBLE ontology network and managed within a Semantic Metadata Repository residing on the NIMBLE platform. With the semantic modelling, resources from different partners in various formats can be semantically annotated and published on the NIMBLE platform to facilitate business-to-business transactions.

The semantic modelling is based on existing standards and domain knowledge. For the building of the catalogue ontology, the UBL standard, is analysed and reused. For the building of the business process ontology, ontology statements within the existing ontological resource, Moda-ML process ontology, is referenced. In the extensions, several standards and resources are analysed and transformed into ontology modules, including the eCl@ss standard, ISO 10303-236 (Industrial automation systems and integration -- Product data representation and exchange -- Part 236: Application protocol: Furniture catalogue and interior design), Moda-ML etc. The ontology network, though it does not cover all mentioned modules and concepts in detail, provides connection points to other deliverables, most notably to D3.2 “Catalogue Ingestion and Semantic Annotation”, D3.3 “Product and Service Search Engine and Search Mediator” and D3.6 “Lifecycle Data Management and Analytics”

Open issues concern

- the further extension of the ontology network by other domain specific product ontologies
- the potential for semantic mappings between various product ontologies
- the ontology governance mechanisms for the process of continuous introducing ontologically sound concepts into the NIMBLE platform

These issues will not be addressed during the NIMBLE project, as they are out of scope of the task. They should be considered during validation and regular use in case such issues prove to pose problems for further exploitation of the framework offered by NIMBLE.

5 References

AXENATH, Björn, KINDLER, Ekkart, and RUBIN, Vladimir (2005). The aspects of business processes: An open and formalism independent ontology. Technical Report TR-RE-05-256, University of Paderborn, http://wwwcs.uni-paderborn.de/cs/kindler/Publikationen/copies/AKR05.pdf, April 2005.

BERNERS-LEE, Tim (2006): Linked Data. Design issues, World Wide Web Consortium (W3C), https://www.w3.org/DesignIssues/LinkedData.html, Retrieved 2017-08-30.

BIZER, Christian, HEATH, Tom and BERNERS-LEE, Tim (2009) Linked Data - the story so far International Journal on Semantic Web and Information Systems, 5, (3), pp. 1-22.

CENSONI, Piergiorgio, DE SABBATA, Piero, CUCCHIARA, Guido, VITALI, Fabio, MAINETTI, Luca, IMOLESI, Thomas (2002), Moda-ML, a vertical framework for the Textile- Clothing sector based on XML an SOAP, Challenges and Achievements in E-business and E-work, B. Stanford-Smith et al. (Eds.), IOS Press, 2002

FERNÁNDEZ-LÓPEZ, Mariano, GOMEZ-PEREZ, Asuncion and JURISTO, Natalia (1997) METHONTOLOGY: from ontological art towards ontological engineering. In: Spring symposium on ontological engineering of AAAI,Stanford University, Standord, CA, pp 33–40

FORSBERG, Martin (2011). Svekatalog UBL Catalogue 2.1. http://www.sftiverifiering.se/svekatalog/XSD/1_1.htm. Last visited 2017-09-20.

GIBSON, Brian J., HANNA, Joe B., DEFEE, C. Clifford, CHEN, Haozhe (2013): The Definitive Guide to Integrated Supply Chain Management. Optimize the Interaction Between Supply Chain Processes, Tools, and Technologies. Upper Saddle River New Jersey: Pearson Education.

GOLDKUHL, Göran (1996). Generic Business Frameworks and Action Modelling. In Proceedings of Conference Language/Action Perspective'96 , Springer-Verlag

Gruber, Thomas R. (1995). Toward Principles for the Design of Ontologies Used for Knowledge Sharing. International Journal of Human and Computer Studies, 43(5/6): 907-928

GOMEZ-PEREZ, Asuncion and SÚAREZ-FIGUEROA, Mari Carmen (2009). Neon methodology for building ontology networks: ascenario-based methodology.Proceedings of the International Conference on Software, Services & Semantic Technology, So-fia, 28-29 October 2009.

HEPP, Martin (2005): eClassOWL: A Fully-Fledged Products and Services Ontology in OWL. In: Poster Proceedings of the 4th International Semantic Web Conference (ISWC2005), November 7–11, 2005. Galway, Ireland, 2005

HEPP, Martin (2008). Goodrelations: An ontology for describing products and services offers on the web. In Proceedings of the 16th International Conference on Knowledge Engineering and Knowledge Management, Acitrezza, Italy, 2008.

HEPP, Martin and ROMAN, Dumitru (2007): An Ontology Framework for Semantic Business Process Management, Proceedings of Wirtschaftsinformatik 2007, February 28 - March 2, 2007, Karlsruhe

ISO/IEC 19845:2015 (2015), https://www.iso.org/standard/66370.html Last visited 2017-09-20.

LEE, Taehee, LEE, Ig-hoon, LEE, Suekyung, LEE, Sang-goo, KIM, Dongkyu, CHUN, Jonghoon, LEE, Hyunja, SHIM, Junho (2006): Building an operational product ontology system. Electronic Commerce Research and Applications 5, 16–28 (2006)

LIEGL, Philipp, ZAPLETAL, Marco, PICHLER, Christian, STROMMER, Michael (2010). State-Of-The-Art In Business Document Standards. Industrial Informatics (Indin), 8th Ieee International Conference On. IEEE, 2010

NOY, Natalya F. and MCGUINNESS, Deborah L. (2001) Ontology development 101: A guide to creating your first ontology. Technical Report SMI-2001-0880, Stanford Medical Informatics, 2001.

OASIS Universal Business Language TC (2014) http://docs.oasis-open.org/ubl/UBL-conformance-to-CCTS/v1.0/UBL-conformance-to-CCTS-v1.0.html. Last visited 2017-09-20.

Reference Collaborative Processes in eBIZ/TC Upstream (2013) http://www.moda-ml.org/moda-ml/imple/moda-ml-2013-1.asp?lingua=en&modo=&sito=Moda-ML&pag=2 Last visited 2017-09-20.

ROSPOCHER Marco, GHIDINI Chiara and SERAFINI Luciano (2014): An ontology for the Business Process Modelling Notation. In: 8th Int. Conf. on Formal Ontology in Inf. Systems (FOIS). (2014)

SMITH Barry et al. (2015) Basic Formal Ontology 2.0 SPECIFICATION AND USER'S GUIDE. http://purl.obolibrary.org/obo/bfo/Reference. Last visited 2017-06-30.

STOLZ, Alex, RODRIGUEZ-CASTRO, Bene, RADINGER, Andreas, HEPP, Martin (2014, May). PCS2OWL: A generic approach for deriving web ontologies from product classification systems. In European Semantic Web Conference (pp. 644-658). Springer International Publishing.

SUÁREZ-FIGUEROA, Mari Carmen et al. (2008). NeOn Deliverable D5.4.1. NeOn Methodology for Building Contextualized Ontology Networks. NeOn Project. http://www.neon-project.org. February 2008

USCHOLD, Mike and GRÜNINGER, Michael (1996) Ontologies: Principles, Methods, and Applications. Knowledge Engineering Review 11, 93–155

VON ROSING, Mark, LAURIER, Wim Paul Remi and POLOVINA, Simon (2015): The BPM Ontology. In: The Complete Business Process Handbook: Elsevier, 101-121, 2015

YARIMAGAN, Yalin and DOGAC, Asuman (2009): A Semantic-Based Solution for UBL Schema Interoperability. In IEEE Internet Comput. 13 (3), pp. 64–71. DOI: 10.1109/MIC.2009.50.

Annex A: Excerpt of Entities involved in Micuna Use Case

Resources involved in the childcare furniture scenario of these resources are:

- Partner profile: manufacturer of furniture, manufacturer of supplying goods, retailer, etc.
- Product specifications: quality, reliability, price, etc.
- Service specifications: delivery term, conditions, price, etc.
- Production processes: machining, painting, assembly, packaging, etc.
- Regulatory aspects: normative, legislation in markets, certifications, etc.
- Environmental issues: recyclability, material sustainability, etc.

Objects

Column named MANDATORY indicates whether or not the data field is mandatory (if checked) or optional, while PRIVATE column indicates if the field is susceptible of being private or not. If some field is checked as private, the company is free to set it as public (making it visible to all the members of the platform), or set it as private (making it visible only to specific companies at specific collaboration stages, such as during negotiation).

PRODUCT SUPPLIER DATASHEET

GENERAL INFORMATION				
		MANDATORY	PRIVATE	CORRESPONDING UBL ELEMENT
Name		X		cac:Party/cac:PartyName/cbc:Name
Code				cac:Party/cbc:IndustryClassificationCode
NIF/VAT		X		cac:Party/cac:PartyTaxScheme/cac:TaxScheme
Address		X		cac:Party/cac:PostalAddress
Zip code				cac:Party/cac:PostalAddress/cbc:PostalZone
Town		X		cac:Party/cac:PostalAddress/cbc:CityName
Region				cac:Party/cac:PostalAddress/cbc:Region
Contact person		X	X	cac:Party/cac:Person
Position of contact person				cac:Party/cac:Person/cbc:JobTitle
Email		X		cac:Party/cac:Contact/cbc:ElectronicMail
Telephone		X		cac:Party/cac:Contact/cbc:Telephone
Fax				cac:Party/cac/Contacy/cbc:Telefax
Classification				cac:Party/cbc:IndustryClassificationCode
Supplier facilities			X	cac:Party/cac:PhysicalLocation
ISO compliance				cac:Party/cac:PartyLegalEntity
OSHAS certification				cac:Party/cac:PartyLegalEntity
EFQM implementation				cac:Party/cac:PartyLegalEntity
Awards received				TBD

Penalty clauses for late delivery			X	cac:DeliveryTerms/cbc:Amount
Quality indicators				TBD
Technical advice				cac:ItemSpecificationDocumentReference
TRADE CONDITIONS				
Method of payment		X		cac:PaymentMeans
Payment conditions		X		cac:PaymentTerms
Payment days				cac:PaymentTerms/cac:SettlementPeriod /cbc:DurationMeasure
Bank entity		X	X	cac:FinancialAccount /cac:FinancialInstitituonBranch /cac:FinancialInstitution
Bank account number		X	X	cac:FinancialAccount
Discounts				cac:AllowanceCharge/cbc:Amout
	Trade discounts		X	cac:AllowanceCharge /cbc:AllowanceChargeReasonCode
	Rappel (variable)		X	cac:AllowanceCharge /cbc:AllowanceChargeReasonCode
	Cash discount		X	cac:AllowanceCharge /cbc:AllowanceChargeReasonCode
Delivery conditions				
	General comments			cac:DeliveryTerms/cbc:SpecialTerms
	Delivery period		X	cac:Delivery/cac:PromisedDeliveryPeriod
	Freight type			cac:Shipment/cac:ShipmentStage /cbc:TransportModeCode
	Insurance		X	cac:Delivery/cbc:InsuranceValueAmount
	Services			cac:Shipment/cbc:HandlingInstructions or cbc:DeliveryInstructions
	Other fees		X	cac:Shipment /cac:DeclaredForCarriageValueAmount

LOGISTICS SUPPLIER DATASHEET

GENERAL INFORMATION				
		MANDATORY	PRIVATE	CORRESPONDING UBL ELEMENT
Name		X		Same as above
Code				Same as above
NIF/VAT		X		Same as above
Address		X		Same as above
Zip code				Same as above
Town		X		Same as above

Region				Same as above
Contact person		X	X	Same as above
Position of contact person				Same as above
Email		X		Same as above
Telephone		X		Same as above
Fax				Same as above
Classification				Same as above
Supplier facilities			X	Same as above
ISO compliance				Same as above
Safety stock				cac:ItemManagementProfile /cbc:MinimumInventoryQuantity
Penalty clauses for late delivery			X	Same as above
Quality indicators				Same as above
Technical advice				Same as above
TRADE CONDITIONS				
Method of payment		X		Same as above
Payment conditions		X		Same as above
Payment days				Same as above
Bank entity		X	X	Same as above
Bank account number		X	X	Same as above
Discounts				Same as above
	Trade discounts		X	Same as above
	Rappel		X	Same as above
	Cash discount		X	Same as above
Service conditions				
	General comments			Same as above
	Cargo days			cac:Delivery/cac:PromisedDeliveryPeriod /cbc:DurationMeasure
	Freight type			Same as above
	Insurance		X	Same as above
	Services			Same as above
	Other fees		X	Same as above
SERVICE SPECIFICATIONS				
Delivery method		X		cac:Shipment/cac:ShipmentStage /cbc:TransportModeCode
Destination		X		cac:DeliveryCustomerParty
Insurance			X	cac:Shipment /cbc:InsuranceValueAmount
Rate			X	cac:Shipment /cac:DeclaredForCarriageValueAmount
Quality indicator				TBD

Certifications				cac:Item/cac:Certificate
General features			X	cac:Shipment/cbc:Information

PRODUCT SPECIFICATIONS

		MANDATORY	PRIVATE	
Name		X		cac:Item/cbc:Name
Code		X		cac:Item/cac:CommodityClassification
Price			X	cac:Price/cbc:PriceAmount
Product features		X		cac:Item/cac:AdditionalItemProperty
Minimum order quantity			X	cac:CatalogueLine/cbc:MinimumOrderQuantity
Materials				cac:Item/cac:ItemSpecificationDocumentReference
Certifications				cac:Item/cac:Certificate
Technical safety datasheet			X	cac:Item/cac:ItemSpecificationDocumentReference
Tolerance/Allowed measurement deviations			X	cac:Item/cac:Dimension/cbc:MinimumMeasure, cbc:MaximumMeasure
Packaging type			X	cac:GoodsItem/cac:ContainingPackage/cbc:PackagingTypeCode
Product image				cac:Item/cac:ItemSpecificationDocumentReference
Usual delivery period				cac:Delivery/cac:EstimatedDeliveryPeriod
Delivery period in non-conformity claims			X	TBD
Graduated prices by quantity			X	TBD
First product sample free of charge				cac:InvoiceLine/cbc:FreeOfChargeIndicator
Quantity per package			X	cac:GoodsItem/cac:ContainingPackage/cbc:Quantity
Technical product datasheet			X	cac:Item/cac:ItemSpecificationDocumentReference
Product guarantee			X	cac:CatalogueLine/cbc:WarrantyInformation
Safety stock			X	cac:ItemManagementProfile/cbc:MinimumInventoryQuantity

More in detail, special features can be defined depending on the type of material, which is especially useful for searching issues.

		MANDATORY	PRIVATE
Wooden boards or spare parts			
	Type of wood		
	Defined quality specifications		
	Forestry chain of custody certificate		
	Origin of wood		
Polishing			
	Technical datasheet		
	Product application		
	Technical safety datasheet		
Chipboards			
	Type of board		
	Composition		
	Tolerance/Allowed measurement deviations		
	Forestry chain of custody certificate		
Plastics	Technical datasheet		
	Composition		
	Tolerance/Allowed measurement deviations		
	Piece customization		
	Own mould making		
	R&D involvement		

The childcare furniture scenario considers the following Terms and Objects:

PARTY also called provider, which includes any type of NIMBLE member, such as:

- Product manufacturer
- Supplier (of components, materials or external operations)
- Logistics supplier
- Retailer

(Other kind of stakeholders such as AIDIMME, regulatory authorities, certification bodies, charitable organizations and end customers have not been included in the list above so because they would are not planned to be members directly involved in the platform at first)

PRODUCT: this entity represents any sellable element. In this regard, the furniture sector taxonomy includes a wide categorisation of products focusing on the furniture market, such as compositions, pieces of furniture, complements and related textile items.

CATALOGUE: this is the main class to represent the collection of products produced by a given manufacturer. This may have different product catalogues associated.

LEGISLATION: laws suggested by a government and approved by the corresponding parliaments with some relationship with the furniture industry are represented by this entity. This are usually laws about preventive measures and considerations about the use of specific substances and materials as well as safety conditions during the productions processes.

REGULATION: they refer to specific requirements that can take on various forms. In the furniture industry scope, they mainly address specific industry issues such as product safety, consumer protection and other factors in public interest. They are the way the legislation is enforced by regulators supporting the requirements of the legislation. Regulations may be developed either internally or externally through technical specifications or standards in the private sector.

CERTIFICATION: this entity represents the documents that prove that a product, service, system, process, person or company complies with the requirements established in a specific regulation. To obtain these documents, a certification process, which is usually a documental process, is needed. In the furniture scope, this process usually also involves a carousel of tests, trials and analysis to assess the compliance with the established requirements.

STANDARD: this represents those technical/legal documents which contain technical specifications eligible to be adopted and developed by consensus of the interested parts, such as manufacturers, users, administration, association and other parties. They result from the experience and the technical development and they are approved by standardisation bodies at regional, national or international level.

PRODUCTION PROCESS: the furniture sector taxonomy of AIDIMME covers productions process in furniture industry, such as the ones involved in the Micuna use case as for example:

- Machining
- Pre-assembly
- Finishing (lacquering, painting, polishing, etc.)
- Assembly
- Packaging

Annex B: Semantic Web Technologies in NIMBLE Platform

One main goal of the semantic web is to add meaning to the content on the web with the help of common metadata vocabularies, and to ensure that the content is machine readable and understandable. Revolving around the semantic web are technologies including: RDF, RDFS, SKOS, OWL, Ontology, Linked Data, SPARQL, RDF triple store etc. The following is a short introduction to semantic web technologies.

RDF[24]: Resource Description Framework. It is a framework for representing information in the Web. An RDF triple is in the form: (subject, property , object). The triple can be interpreted as the “subject” has “property” with value “object”. A set of such RDF triples is called RDF graph.

[24] https://www.w3.org/TR/2004/REC-rdf-concepts-20040210/

RDFS [25]: Resource Description Framework Schema. RDFS is a semantic extension of RDF. It provides mechanisms for describing groups of related resources and the relationships between these resources. In RDFS, predefined vocabularies rdfs:Class, rdfs:Resource and rdf:Property can be used to define classes, resources and properties, respectively.

SKOS [26]: Simple Knowledge Organization System. It provides a standard way to represent knowledge organization systems such as thesauri, classification schemes, taxonomies, subject-heading systems, or any other type of structured controlled vocabulary using the RDF. For example, the skos:broader and skos:narrower are two example core vocabularies defined to assert a direct hierarchical link between two concepts.

OWL [27]: Web Ontology Language. OWL is a computational logic-based language designed to represent rich and complex knowledge about things, groups of things, and relations between things. It extends RDF and RDFS and provides more concepts and constraints for the description of classes and properties. For example, the vocabulary "owl:someValueFrom" expresses restrictions on the value of some object property. The OWL recommendation provides three languages of increasing expressiveness: OWL-Lite, OWL-DL and OWL-Full.

Ontology: an ontology is a 'representation of a shared conceptualisation' of a specific domain in computer science (Gruber 1995). An ontology provides explicit formal specifications on common vocabularies including concepts, properties and constraints, used in a domain. It can be used to support the communication and semantic interoperability across distributed heterogeneous systems. Ontologies can be expressed in description logics of varying expressivity. RDFS provides basic elements for the description of ontologies. Compared to RDFS, OWL provides more expressive power and is recommended by W3C for the semantic Web. An ontology consists of a terminological part (TBox) that defines concepts and roles of a given domain, and an assertional part (ABox) that describe individuals (instances) and the values of their object properties using the roles and concepts defined in the TBox. By analogy with databases, the TBox is similar to the data schema, and the ABox can be regarded as the stored data set according to the schema.

Linked Data: Linked Data is a method for interlinking structured data in fashion that is compatible with the World-Wide-Web. It refers to data published in such a way that it is machine-readable, its meaning is explicitly defined, it is linked to other external data sets, and can in turn be linked to, from external data sets (Bizer et al. 2009).

SPARQL[28]: SPARQL Protocol and RDF Query Language**.** It is a protocol and query language to retrieve and manipulate data stored in RDF format.

RDF triple store[29]: an RDF triple store is a purpose-built graph database designed for storage and retrieval of RDF triples. Apache Marmotta [30] is an Open Platform for Linked Data. It provides methods to maintain data in the form of triples. It is easily possible to query the data by

[25] https://www.w3.org/TR/2004/REC-rdf-schema-20040210/
[26] https://www.w3.org/2004/02/skos/intro
[27] https://www.w3.org/OWL/
[28] https://www.w3.org/TR/rdf-sparql-query/
[29] https://www.w3.org/2001/sw/Europe/events/20031113-storage/positions/rusher.html
[30] http://marmotta.apache.org/

means of SPARQL or LDPath queries and also to identify new information by using the built-in rule-based reasoner.

For the purpose of facilitating communication and semantic interoperability across distributed heterogeneous systems in B2B e-commerce, it is necessary to have semantic models of the manufacturing collaboration assets for the representation of various types of tangible resources and intangible resources. The semantic models are in the form of ontologies. We have adopted a gradual approach to introducing ontologically sound concepts into the system, in order to maintain semantic structure within the vocabulary of the NIMBLE platform. All relevant shared common vocabularies and their properties and constraints in the ontology modules are maintained in the semantic metadata repository. In the NIMBLE platform, we are using an open source RDF triple store and Linked Data Server – Apache Marmotta – as semantic metadata repository. Apache Marmotta is able manage both the ontology TBox and the ontology ABox. On the NIMBLE platform, this will work in the following way: first, the ontology TBox, e.g. the catalogue ontology, is imported into an Apache Marmotta instance. Afterwards, individuals of data triples in the form of the ontology ABox are semantically annotated and published via the same Apache Marmotta instance. With the managed ontology's TBox and ABox, it allows the execution of SPARQL to semantically retrieve and manipulate data stored in Apache Marmotta. Details on how this works with the semantic web technologies to support semantic annotation and semantic search in the NIMBLE platform can be found in deliverable D3.2 "Catalogue Ingestion and Semantic Annotation" and D3.3 "Product and Service Search Engine and Search Mediator".

More detailed introductions related to ontologies, linked data and semantic web can be found in the following resources:

1) Semantic Web: https://en.wikipedia.org/wiki/Semantic_Web

2) Introduction to Ontologies and Semantic Web: https://www.obitko.com/tutorials/ontologies-semantic-web/

3) Introducing Linked Data And The Semantic Web: http://www.linkeddatatools.com/semantic-web-basics

4) Tutorial on Semantic Web: https://www.w3.org/Consortium/Offices/Presentations/RDFTutorial/

5) Semantic Integration & Interoperability Using RDF and OWL: https://www.w3.org/2001/sw/BestPractices/OEP/SemInt/

6) Web Data Management: http://webdam.inria.fr/Jorge/html/