THE TRUE COMPLEXITY OF PRODUCT REPRESENTATION IN THE SEMANTIC WEB

Hepp, Martin, Digital Enterprise Research Institute, University of Innsbruck, Technikerstrasse 21a, A-6020 Innsbruck, Austria, martin.hepp@deri.org

Abstract

The ontological representation of products and services is a core challenge on the road to business applications for the Semantic Web. This will not only help search engines provide more precise product search for human users, but can be expected to support a much higher degree of business process automation in general, especially in all tasks that involve content integration. In industrial data interchange between business partners, the state of the art is the use of common XML schema definitions (e.g. BMEcat) for the representation of structure and the use of classification schemes (e.g. UNSPSC or eCl@ss) for the representation of product semantics. This current practice, however, takes place in well-defined contexts known to both the publisher (or sender) of data and the recipient, which allows even the usage of the same standard with varying semantics in distinct settings. In a Semantic Web context in contrast, the same document must be machine-readable (1) by a huge number of different partners (2) for a multiplicity of purposes. In other words, the data recipient and the data usage are not predetermined, which makes it much more difficult to reach consensus about suitable product classes. This paper develops the requirements for product representation in the Semantic Web and evaluates existing alternatives.

Keywords: Semantic Web, Content Standards, Product Ontologies, eCl@ss, UNSPSC, eOTD, EGAS

1 INTRODUCTION

An effective communication between machines is a requirement for B2B e-commerce (Corcho and Gómez-Pérez 2001) and the Semantic Web promises to make the vast amount of data on the Web machine-readable and processable, by formalizing the semantics (Berners-Lee et al. 2001). As products and services are the core objects of trade, their machine-readable representation is a key challenge on the road to business applications for the Semantic Web (cf. Schulten et al. 2001, Hepp 2005a).

The most obvious business process that would gain from a machine-readable representation of products and their properties is the search for potential matches between buyers and sellers and finding the best one (even if there is no perfect match), which is known as *matchmaking* (Di Noia et al. 2003). However, there are far more tasks that will benefit from a machine-readable representation of products in the Semantic Web. Examples are (1) recommender systems, which require *points of reference for specific product categories or product models* in order to store their domain knowledge, (2) expressing, formalizing, transmitting and even trading of *relationship knowledge* about products (e.g. "charcoal sells well with grill sausages"), (3) analytical tasks like cost accounting, spend analysis, and benchmarking, (4) expressing a demand when querying a search engine (e.g. "I need a mounting part that is non-corrosive and can hold 500 kilograms"), and (5) multi-vendor catalog data integration from *distributed* sources (e.g. gaining missing information by inference operations from public Semantic Web documents).

The set of attribute names used to describe a product may be called a *product schema*, following the relational database terminology (Ng et al. 2000). One can further differentiate between a *local schema* (which is a vendor-specific set of attributes) and a *global product schema* (which is a standardized, commonly used schema) (Ng et al. 2000). The attribute lists in eCl@ss (http://www.eclass.de) are an example of such a global product schema.

Basically, there are two different ways to handle heterogeneous product descriptions: (1) build a standard or (2) build a layer that maps different standards (Ng et al. 2000, Schulten et al. 2001). Content standards that can be used for the representation of products in a machine-readable form are often referred to as *product ontologies* (Ding et al. 2002). As the degree of formality in many available standards for product description is comparatively low, the term "descriptive languages for products and services" has been proposed (Hepp 2004). Their scope is not limited to tangible products, as such ontologies (or descriptive languages) can also be used for the description of services and agents (Obrst et al. 2001). For an overview of the role of ontologies for B2B e-commerce see Obrst et al. (2001).

Schulten et al. (2001) called for a comprehensive, theoretically founded, and practically useful model of product description that would both address current B2B e-commerce needs as well as be compatible with the Semantic Web. However, despite numerous industrial and academic activities in the field, there is so far no common representation scheme nor consensus on the appropriate level of detail or granularity (Corcho and Gómez-Pérez 2001, Hepp 2004). Current descriptive languages for products and services, like the UNSPSC (http://www.unspsc.org), the German approach eCl@ss, and the eOTD (http://www.eotd.org) lack both the required coverage of concepts and semantic precision (Hepp 2004, Hepp et al. 2005a, Hepp et al. 2005b). It seems to be that replacing the human component in electronic buying processes is more complex than many had expected (cf. Schulten et al. 2001).

One core problem with those traditional classification-based approaches is that in a Semantic Web context, the same document must be *machine-readable by a huge number of different partners for a multiplicity of purposes.* In other words, the data recipient and the data usage are not predetermined, which makes it difficult to reach consensus about suitable product *classes that represent the nature and potential usages of a product in sufficient detail.*

This paper develops the requirements for product representation in the Semantic Web, evaluates alternatives and aims at contributing to a common understanding of the true complexity of product representation in the Semantic Web.

The structure of the paper is as follows. In section 2, we summarize the characteristics of the Semantic Web as an environment for product-related data. Section 3 contains an analysis of the complexity of products and services representation in this environment. Section 4 compares alternative approaches of product representation, and section 5 discusses the implications and summarizes our findings.

2 CHARACTERISTICS OF THE SEMANTIC WEB ENVIRONMENT

Most previous works on product data interoperability are related to catalog data exchange. For an overview of catalog data integration see Fensel et al. (2001), Agrawal and Srikant (2001), and Leukel et al. (2002a). Catalog integration is a straightforward task, as the sender, the recipient, and the intended use of the document is known. The catalog document is created individually for one specific recipient or at least for a well-defined set of recipients. It is usually self-contained in the sense that all data for one product comes from just one party, even though many standards support abbreviated documents for product or price updates that do not contain the unchanged part of the data (cf. Leukel et al. 2002a). Thus, in the context of catalog data exchange, there exists (1) a shared process context and (2) a limited need for inference operations.

Today, product data annotation is often done by a content management solution provider (Omelayenko and Fensel 2001). Those providers use mostly semi-automatic approaches that involve human interaction. Even, however, for the narrow application of catalog data exchange, the problem of machine-readable product description has so far not been solved successfully: Incompleteness and inaccuracy are two fundamental problems, and it is still often a manual job – solution providers have several hundred employees who manually process the data sets in content factories (Fensel et al. 2001).

In the Semantic Web, this is no longer a feasible approach, as (1) ad-hoc translation between different product representations and (2) inference operations about product data from multiple sources must be possible in real-time. In a Semantic Web environment, inferring additional knowledge from multiple external sources to overcome both incompleteness and inaccuracy will be the daily practice (see Figure 1 below), and chains of information processing with agents passing components to each other in order to yield the final product or service might be, too (Berners-Lee et al. 2001). This chaining of information from multiple sources means also that the inherent dynamics of concepts (e.g. new product categories, new features, etc.) from different semantic communities might multiply. Product representation is in this respect different to e.g. life sciences where the community consensus about a domain can be expected to be more explicit and more stable.

As a consequence, product representation in the Semantic will have to deal with three major challenges:

1. The data must be suitable for a broad audience and a variety of purposes, which might be unknown to the publisher.
2. The information about a product will be retrieved and assembled from multiple documents stored on many different systems.
3. Product concepts change over time and new concepts evolve rather quickly.

2.1 Unknown Data Recipient and Data Usage

In the Semantic Web, access to ontological data will often be in the form of crawling persistently published RDF data, very much like current search engine bots visiting a company's Web pages. It is beyond the data provider's control who will be accessing and interpreting that public data for which purpose. It is noteworthy that, as of today, even for B2B catalog data exchange, additional negotiations, agreements and adjustments regarding syntax, content, and quality of the data have to be made (cf. Leukel et al. 2002a). As the relationship between recipient and sender is explicit in that environment, this poses a much lesser problem, though – at least it is clear whom to ask for clarification and who to inform of past inconsistencies as soon as they are detected.

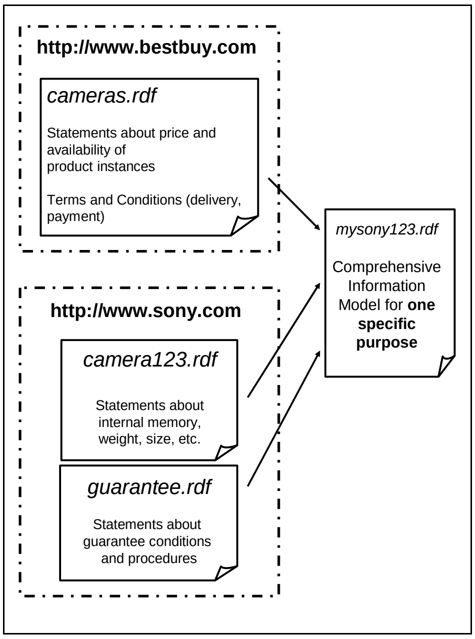
http://www.bestbuy.com
cameras.rdf
Statements about price and availability of product instances
Terms and Conditions (delivery, payment)
http://www.sony.com
camera123.rdf
Statements about internal memory, weight, size, etc.
guarantee.rdf
Statements about guarantee conditions and procedures
mysony123.rdf
Comprehensive Information Model for one specific purpose

Figure 1. A typical scenario of product data usage in the Semantic Web

When it comes to Web data integration, product schema integration has to deal with limited knowledge of local schemas, a large number of local schemas, and possible frequent changes to the local schemas (Ng et al. 2000). In P2P matchmaking, even what is the query and what is the data can be just a question of the perspective (Di Noia et al. 2003), and B2B marketplaces can be both providers and consumers of Semantic Web data. With regard to catalog data integration, it is very probable that at least parts of future catalogs will be assembled by bots that enrich input data with additional information.

This creates specific requirements that must be properly addressed in advance, in order to avoid incorrect inference operations with potentially negative consequences for the publisher of the data. For example, a statement made on the US Web site of Sony for the domestic market might not be valid outside the United States.

Additionally, it is in the business interest of any data publishing entity to contribute to the correctness of inferences based on its data, in order to avoid lost business, unhappy customers, extensive customer service, or similar disadvantageous situations. This is different from the situation in the classical World Wide Web, where it is seldom a disadvantage if search engines falsely list your page for a given search term, and the responsibility for false inferences is always solely on the side of the consumer of the data. As soon as the data consumers are machines, such false positive situations can be as negative as a false negative result.

2.2 Distributed Nature

A common situation will be that (1) manufacturers of products publish data about their products in Web documents, (2) dealers provide additional information and prices, (3) third-parties provide additional data, information, knowledge or services (e.g. recommended products for a given purpose), see also Figure 1. That means that we will have to deal with statements stored in multiple places by different business entities, for individual purposes and driven by different incentives.

This distributed environment means that any operation on the data must be able to (1) properly handle incomplete information, (2) distinguish missing information from negative information, and (3) identify exact matches, potential matches, and partial matches (cf. Di Noia et al. 2003) between a demand and a supply. For example, the absence of a characteristic in a description should be treated as a characteristic that could either be refined later or ignored (if irrelevant) (Di Noia et al. 2003). Incomplete information will be a very common situation in the Semantic Web. The refinement can be achieved by two ways: We either need a protocol for the request of additional characteristics, e.g. a Web Service; or we must obtain the missing piece of information by reasoning and / or using facts from additional sources (see Figure 1). In addition to that, it must be guaranteed that more specific descriptions are not inferior to unfairly generic or simply less specific descriptions (though the later might contain some contradictions to the counterparty's description) (Di Noia et al. 2003). Otherwise the strange situation would result that the less information you provide, the more often are you considered a potential match – ignoring the value of specificity for a potential business partner.

Another consequence of the distributed environment are agency problems (in the economic sense of agent theory) and generally the question of the trustworthiness of product-related data. A first approach to handle those two issues would be a simple *ranking* of the trustworthiness *of the resources* providing data for the current reasoning operation, and to use this ranking to resolve conflicting statements (e.g. a document from http://www.sony.com lists a different memory capacity as compared to the document from an Internet retailer → the statement made by the retailer might override the manufacturer's statement).

It is important to note that companies have an internal product representation, usually being stored in their ERP system, and that this will frequently be the most reliable source for product data. Thus, bridging the gap between the ERP-internal representation and the data published on the Semantic Web will be a core task for businesses.

2.3 Dynamics and Volatility of Concepts

Obrst et al. classify product ontologies as relatively stable as compared to vendor-specific data (Obrst et al. 2001). This is in general true, however, there is still significant dynamics in the product domain, as new product categories evolve and old product categories become irrelevant. Thus, the product terminology needed by buyers and sellers is dynamic (Guo and Sun 2003a). In other words, product-related ontologies may require changes over time. This creates a need for continuous maintenance (cf. Schulten et al. 2001, Hepp 2003).

Basically, the dynamics and volatility of concepts is directly related to their specificity. The higher the specificity of the concepts, the more dynamic is a vocabulary. A two-concept ontology „Thing“ and „Thought“ will work without change infinitely, and an overly complex ontology containing individual concepts for each phenomenon on earth at any given point in time („Florida on December 1, 2005“, „Florida on December 2, 2005“) would be the extreme opposite with infinite dynamics. Because the lowest level of semantic precision limits the quality of possible mappings, an ontology should provide a very detailed partitioning (Omelayenko and Fensel 2001). Thus, a proper representation of products in the Semantic Web will require a high-degree of semantic specificity, which goes far beyond the granularity of current descriptive languages like eCl@ss and UNSPSC. For example, none of us would ask a shopping bot to order the cheapest instance of "beverage", while this is likely the degree of specificity we can achieve when reusing industrial categorization schemes. In this respect it is important to note that industrial categorization schemes were created in the first place for (1) *aggregating* entities for (2) some purpose, and not for describing the nature of an entity. For example, for cost accounting purposes, an invoice for ice cubes can be treated as an invoice for beverages, but when we search for a beverage, we do not want to see instances of ice cubes in the result set (Hepp 2005a).

In a simulation approach (Hepp 2003), the conceptual dynamics in various industries was analyzed and set them into relation to various standards maintenance lags. One can see that there is significant concept dynamics in multiple terminological segments, which causes unsatisfying coverage rates. One approach to deal with that dynamics is to accelerate the maintenance of a content standard.

3 COMPLEXITY OF PRODUCT REPRESENTATION

The problem of product representation can be divided into three sub-problems:

1. providing *means for expressing* statements about a product (e.g. domain ontologies),
2. having or *gaining respective knowledge* about the product, and
3. *expressing and publishing* respective *statements* (assertions) about the product.

This differentiation is essential, as those three tasks need not to be performed by the same party, nor at the same time, and also not for the same reasons or driven by the same incentives. The only requirement is that they are performed in the order as listed above, as one step requires the results from the preceding one. This is an important observation because it implies that due to the delay in updating ontologies so that they include product innovations, new products cannot be immediately described in a class-centric product ontology.

It will be a realistic scenario that the industry interest group A will provide an ontology for the digital camera market, two manufacturers B and C of digital cameras will publish statements about their models (e.g. technical details), and an Internet shop D will provide price information and application-specific product recommendations in a machine-readable form. It is very important to note that not all knowledge about a product needs to be provided by its manufacturer. Even the statement that a product belongs to a specific class of some content standard (e.g. eCl@ss) can be made by someone else at some later point in time somewhere else. This is in sharp contrast to the situation in catalog data exchange between business partners, where catalog documents are usually self-contained (see section 2.2 above). An important observation in this respect is that the ontological commitment by the manufacturer or vendor is not required, i.e. even if the manufacturer does not support semantic annotations of its products that would ease automated product comparison, he cannot prevent anybody else from publishing such annotations, as soon as a unique identifier for the product model exists.

As ontologies represent a shared and common understanding of a domain, in such a way that it is accessible by both humans and machines (Ding et al. 2002, Fensel 2004), it is natural that there exist approaches to create ontologies for products and services based on product classification schemes, like the DAML+OIL and RDF-S versions of the UNSPSC (McGuinness 2001, Klein 2002). On the way to product ontologies, current product standards could be enriched using attributes, relations, and axioms, which would support reasoning (Corcho and Gómez-Pérez 2001). However, there are several problems when designing products and services ontologies that put the emphasis on product classes:

1. The amount of classes will be enormous, since the specificity required for product discovery will be much higher than that found in current classification schemes, and those already contain more classes (about 25,000 in the case of eCl@ss and UNSPSC) than a sophisticated English dictionary.
2. Classes usually group products that can be treated as substitutes. The intensions of the classes and the class hierarchy were defined for some specific purpose that reflects one particular

view on the space of products. It might be, however, that such a thing like a common product class does not exist at all in many areas, because the product space can be regarded as a continuum and product classes can be regarded as *context-bound, subjective judgments* therein. Two products that are perfect substitutes for one person can be completely different for a second person or in a different context. Also, whether a product or service is an instance of a specific category can be bound to a limited scope or timeframe.

3. Di Noia et al. stress the importance of imperfect matches, i.e. such supplies that can "*to some extent* fulfill a demand" (Di Noia et al. 2003). By its nature, a class-centric description method creates problems when it comes to flexibly comparing imperfect matches, because the candidate products might belong to different classes and are thus described using different properties.
4. Even if commonly accepted product classes theoretically exist, it might be that the inevitable maintenance lag (required to reach consensus about a new product class and to add it to the ontology) prevents us from formalizing volatile concepts within their lifespan.

Those two limitations might seem merely academic, but they are not. The lack of content coverage resulting from the maintenance lag on the one hand and the dynamics of product concepts on the other hand has been quantitatively analyzed for the computer market, the chemical industry, and pharmaceutical products (Hepp 2003), and the unsatisfying content quality of current standards has been documented in Hepp (2004) and Hepp et al. (2005b). Moreover, we can expect those limitations to become more serious in the future, because the concepts in current content standards have not yet reached the final degree of specificity, and it is obvious that the problem of content maintenance grows as semantic precision increases. In other words, the more specific the concepts for products are, the more of conceptual dynamics must be dealt with.

As a consequence, the key question when it comes to ontological support for the representation of products is: What are the *concept*s that we need a formal conceptualization for? The common opinion seems to be that *product classes*, similar to those contained in the UNSPSC, eCl@ss, or the EGIS in the eOTD, should be the concepts of a product ontology. As we will detail in section 4, we believe that *product properties* are much more promising as the core of product representation in the Semantic Web.

Whichever approach is taken, it is important that statements about products can refer to varying perspectives on a product. Basically, the description of a product consists of one or multiple assertions (e.g. "is a radio", "weight = 500 kilograms", "is non-corrosive", "may rust", "fits to a Volkswagen", etc.). This can refer to (1) the *nature* of a product, (2) its recommended or potential *usage*, or (3) both.

3.1 Statements about the Nature of a Product

The nature of a product, e.g. its raw material ("stainless steel"), the chemical components ("NaCl" – salt), or details about the production process ("ionized") is one out of multiple perspectives that can be used when describing a product. This is usually the preferred perspective of a manufacturer, because the respective facts are known, and, in addition to that, it is a straightforward approach. If a product classification scheme is based on this perspective (like the UNSPSC), each manufacturer can identify the one single correct entry. It is also comparatively easy to store this entry in the ERP system of that company. However, it might be difficult for customers to find products for a specific purpose or usage. Most buyers know what they want to use a product for, but they might not know what this means with regard to the constraints over the nature and characteristics of a product. The mediation between the intended usage and the product capabilities is, in traditional trade, usually done at the sales side.

3.2 Statements about the Product Usage

The second important perspective when describing a product is its recommended or possible usage (or, in Problem Solving terminology, the goal). For example, common salt (NaCl) can be used in the household, as a desiccant, or to melt snow and ice on the street. The nature of the product can be exactly the same, but those are three different usages. The difficult issue is that whether two products are substitutes or not frequently depends on the intended *usage.* As a desiccant, one can as well use potassium chloride (KaCl), while as table salt, one should not. eCl@ss, for example, is based on product market segments and provides many entries for product *usages.* Statements about the product usage are more helpful for a buyer, because the buyer wants a product for a specific use, not naturally of a specific nature. Statements about the potential use of a product might be difficult to make for manufacturers and vendors, because they might not know what their customers are doing with a product. The same wheel, for example, might be used for carts, toys, or plants and machinery.

Product usage categories and product property categories may sometimes match, but frequently they do not. However, if the Semantic Web is to support comprehensive reasoning about suitable products for various business processes, both types of statements must be part of the product representation. The existence of those two categories of statements indicates that class-centric product description faces serious limitations, because the number of required classes can be very big if they are to reflect all potential usages and product constitutions.

3.3 Scope of Product Properties

When part of a product description, assertions assign properties to products. Such properties can be (1) valid for every make and model that falls into a specific product category, (2) vendor- or model-specific, or (3) characteristics of single product instances.

In other words, the *scope* of a property must be observed, because this determines where and how respective statements shall be stored and how frequently the statements need to be updated. For example, model-specific properties will be assigned in a vendor-specific product model ontology, while properties of product categories will rather be part of the TBox part of a domain ontology. Additionally, some product properties may be dynamic, others may be static.

Properties of Product Categories: Prominent content standards like eCl@ss and ETIM provide attributes that help describe product properties. This allows parametric search, i.e. searching for a product that meets a set of property constraints (e.g. class=TV set with screen size < 11 inches and color=true).

Properties of Product Models: In this category of properties fall most of the product attributes that popular content standards (e.g. eCl@ss and the eOTD) offer for the description of products. Typical examples are weight, diameter, color, dimensions, etc. Both eCl@ss and the eOTD contain a relation between product classes and such attributes. The eOTD link table between classes (EGIS) and attributes (EGAS) is just a rather inconsistent recommendation (Hepp 2003), while the respective "attribute lists" in eCl@ss are well-defined sets of attributes with industry-wide consensus.

Properties of Product Instances: Some properties are valid only for a single instance of a specific product, e.g. the sales location of a specific car, the expiry date of a perishable food, a serial number, or, for a used car, the previous owner. Those properties can also be important for reasoning about alternative offers. For example, a reliable company instead of an unknown individual as the previous owner of a used car might fit to our preferences better, and whether a food product that will expire soon (but is available at a reduced price) is a good deal depends on the actual situation.

Multi-dependent Properties: Leukel et al. have already pointed out that there exist multi-dependencies with regard to catalog data (Leukel et al. 2002a), that means that a fact exchanged in a catalog document can depend on a combination of parameters. The price, for example, is not a static property of a product, neither of a generic product concept, nor of a product understood as a make and model, and not even of a product instance understood as one single item in the inventory, but instead an instance-related property that depends on multiple additional parameters, for example the recipient of the offer and the time-interval during which the offer will be valid. Another typical multi-dependent properties is e.g. the weight of a given volume of a gas (since it varies with temperature and pressure).

3.4 Relationship between Resources and Product Categories

A lot of semantics when representing the products and services domain is kept in the relation between a resource (e.g. a Web resource) and a product description. The simplest type of relationship that can exist between the two is *rdf:type*, which means that the resource is an instance of the respective class.

However, we do in fact not want to say that a specific Web resource is an instance of a specific product category. Much more, we want to represent a different type of relationship between a Web resource and a product definition, e.g.

> "This Web page contains an offer of product instances that meet the following specification",
>
> "The company identified by this URI repairs products of the following kind", or
>
> "This Web page contains a general offer to lease products of the following kind".

So we need an ontology that captures the subtle ways of relationship that can exist between a Web resources and a product definition. This is because a simple property *http://example.com/sell,* can mean a couple of things – e.g., "We offer, in a legally binding manner, to sell *this particular instance*" or "In general, we sell *instances of this type of goods*".

Thus, when making assertions about the properties of a product, it is important to make an assertion about the relationship of the publishing business entity with regard to that product. A very valuable semantic standard providing common concepts for this domain is the UNSPSC Business Function Identifier (BFI) (United Nations Development Programme 2003). It is a simple two-digit value that reflects the kind of relationship between the company and the product category. Table 1 shows the currently supported values.

Table 1. *UNSPSC Business Function Identifiers (United Nations Development Programme 2003)*

UNSPSC BFI	Meaning
10	Rental or Lease
11	Maintenance or Repair
12	Manufacturer
13	Wholesale
14	Retail
15	Recycle
16	Installation
17	Engineered
18	Outsource

4 ALTERNATIVES OF PRODUCT REPRESENTATION

Most product-related data that is currently available on the Web is in the form of pure text, i.e. in natural language. Depending on the progress with regard to natural language processing, this kind of data might be accessible to machine reasoning in the future. Currently, only probabilistic text similarity approaches are widely available; however, they can result in seriously incorrect interpretations. We have to take into account that the text similarity between e.g. “dogs, no smokers welcome” and “dogs: no, smokers welcome” in the description of an accommodation will be quite high, while the difference in meaning is significant (Di Noia et al. 2003). A special problem is that product description is frequently not in the form of full sentences where structural knowledge about the natural language could be exploited.

We currently see the following alternatives of representing products on the Internet in a machine-readable way.

4.1 Ontologies Based on Classification Schemes

The common schema model of a product is a rooted tree, with the tree being the product and the tree nodes being product attributes (Ng et al. 2000). More common is however a tree reflecting a product hierarchy and attributes assigned to the tree nodes. This structure is for example found in the eCl@ss approach and the now nonexistent UCEC attempt to add attributes to UNSPSC. The number of attributes per product category can be very large (Ng et al. 2000). Recent analysis of eClass 5.0 for example, shows that one class (“Bottom globevalve”, primary key AAD661001, eClass code 37-01-02-60) has 266 (!) attributes (Hepp 2004). For an overview of the architecture see Leukel et al. (2002b). Currently, there are multiple systems available, and it cannot be expected that a single product classification system will be used worldwide and spanning all industries (Leukel et al. 2002b).

Zhao and Lövdahl propose that classification standards like UNSPSC can be directly used to build a product ontology (Zhao and Lövdahl 2003). While the inherent consensus regarding product concepts is very valuable, a direct transformation into a product ontology is a problematic task, because the input standards were not created with the rigor of knowledge representation in mind and suffer from several inconsistencies. There has been progress in mechanizing this task in the form of the “gen/tax” methodology (Hepp 2005b) and, as a result, successful transformation of eCl@ss into a fully-fledged products and services ontology in OWL (see http://www.heppnetz.de/eclassowl).

While the results can be used for a variety of business applications of Semantic Web technology, they might not yet be the perfect means for the annotation of offerings on public Web pages. This is for two reasons:

1. The resulting ontologies are very big (between 25 and 50 MB file size) and can thus hardly be retrieved on demand.
2. The amount of inferencing support resulting from the ontologies is limited, since there is just a purpose specific hierarchy.
3. Despite the huge number of classes, the coverage of commonly used commodities is insufficient. So in many cases, there will not be specific classes for the description of a product or service. By their nature, classification approaches are difficult to maintain, because they combine dynamics from multiple spheres: (1) product innovation, (2) attribute innovation, (3) hierarchical order of products for analytical purposes, and they address multiple audiences.

All in all one can question that whether for the annotation of Web pages, the overhead caused by the serialization of the full standard using an ontology language is justified, or whether it would not be more feasible to take a different approach.

4.2 COPE: Combination of Local and Global Product Schemas

Agrawal and Srikant pointed out that even a proprietary categorization of products contains valuable implicit information about the degree of similarity between multiple products (Agrawal and Srikant 2001). Guo and Sun recently proposed a collaborative approach of product representation (COPE) which emphasizes the semantics in local product representation (Guo and Sun 2003a). Guo and Sun recommend to differentiate between *local concepts* of semantic communities and *common concepts* (Guo and Sun 2003b), which is attractive. If, however, the local representations are to be translated into common representations, this still requires commonly accepted concepts, i.e. ontologies. It seems to be that Guo and Sun assume that an ontology would be restricted to formalizing the semantics of *classes for objects* (here: products). However, there is no reason why we could not create *property ontologies,* which formalize the semantics of product properties. Without a commonly accepted definition of the meaning of properties used to locally describe a product, no automatic transformation into a machine-readable representation seems feasible.

Furthermore, the COPE approach does currently not explain how the local semantic communities are able to determine (1) whether a locally needed concept is already contained in the property terminology dictionary and (2) how they are enabled to determine whether a potentially applicable term really represents the desired concept. For example, the attributes „durable“ or „high quality“ might be perceived and used very differently among various local semantic communities.

As a preliminary summary, the COPE approach points to the dynamics of local semantics and contains the valuable idea to focus on properties for product representations, but is not as contradicting to an ontology-focused approach as the authors state. Furthermore, a large-scale proof of this proposal in an industrial setting is not available.

4.3 Description by Example

Instead of classes as abstraction from the instance space, we could make popular instances the core of representation. By pointing to instances of a specific product or service, we could establish community consensus over symbols for meanings. In the context of this paper, we could think of describing a product by referring to a similar product. That could be achieved by creating a very shallow ontology for relevant predicates, e.g. “similar product” and “is a supply for”. Figure 2 shows a simple example, in which a third-party toner cartridge is defined by referring to both the original part as well as the compatible original laser printer.

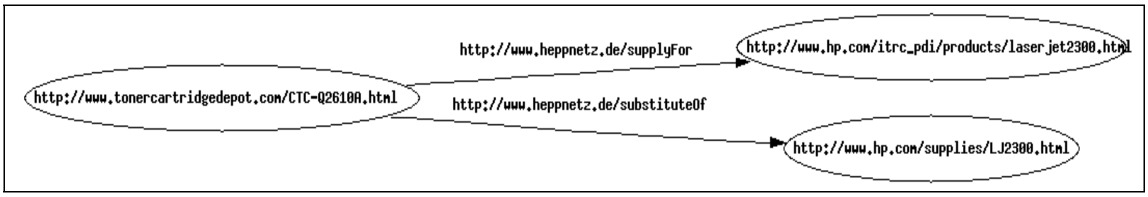
http://www.heppnetz.de/supplyFor
http://www.hp.com/itrc_pdi/products/laserjet2300.html
http://www.tonercartridgedepot.com/CTC-Q2610A.html
http://www.heppnetz.de/substituteOf
http://www.hp.com/supplies/LJ2300.html

Figure 2. Description by Example

The intuitive compatibility with the structure of the current World Wide Web, where similarity in context and content is mainly preserved by links between documents, makes this approach attractive, despite the fact that it does not provide a means to truly formalize the properties of the described product.

4.4 Property-Centric Product Description

Instead of making the classes of categorization schemes the center of product representation, we can also treat the identifiers of such classes as simple literal values and use them just as one specific property among others. This could, for example happen in the following way:

1. We represent the properties that are contained in the property library of an industrial categorization scheme (e.g. eCl@ss or eOTD) as properties in either OWL, RDF-S, or pure RDF. For some properties, there exists enumerative data typing. In OWL, these should

become *owl:ObjectProperties* and the values should be represented as instances. The other properties become simple *owl:DatatypeProperties.*

2. For the categories themselves, we just define one single *owl:DatatypeProperty*, e.g. *http://example.com/IsEClassType* with the range of a string, and use the category codes as the literal value.

This can happen in parallel for many categorization schemes. All this together can then be used to make statements about product models and product instances.

The approach has several advantages:

1. We do not need to replicate the whole standard and extensive class descriptions but just the actually used properties, and rely on the official standards documentation for the establishment of community consensus over the definition of product classes and properties.
2. We do not need to have a proper class before we can start making statements about a product, e.g. we can make a statement about the weight of a new product without having an appropriate ontology class other than “Thing” or “Product”.
3. Elements from multiple standards can be used as building blocks to add semi-formal semantics to product descriptions.

In the following, we give a brief example of this approach in plain RDF. The same can of course be expressed in OWL, too, by simply adding definitions for the used three properties. In the example, “AAB29200202” is the identifier of the category “Photo Camera” and “PMA10500301” is the identifier of the property “Number of Pictures” in eCl@ss 5.0. In UNSPSC, the closest category “Digital Cameras” has the identifier “45121504”.

```
<rdf:Description rdf:about="camera123.html">
<ex:IsEClassType>AAB29200202</ex:IsEClassType>
<!—eCl@ss category "Photo camera" -->
<!-- Number of pictures -->
<eclass50:PMA10500301>100</eclass50:PMA10500301>
<ex:IsUnspscType>45121504</ex: IsUnspscType >
<!--  UNSPSC "Digital cameras" -->
</rdf:Description>
</rdf:RDF>
```

The example shows nicely how we can reuse the concepts from UNSPSC and eCl@ss for adding metadata to the product description.

5 DISCUSSION

For the reasons given above, it is unlikely that product and service ontologies being, at the same time, comprehensive, complete, current and expressive will be available in the near future. However, it is desirable to empower businesses around the world to start *now* with the process of adding machine-readable descriptions to their product-related Web resources. Manufacturers, wholesalers, and retailers should be able to describe what they offer and what they know about their products, at a time of their choice. The collection of statements as a whole will never be complete and might never be consistent, which renders it questionable to *wait* for industry-spanning product ontologies. They might never come, due to the concept dynamics, a lack of resources, or the lack of economic incentives.

We have proposed to use the consensus contained in product classification systems, attribute libraries, and other semantic standards, for the machine-readable description of products in the Semantic Web. Instead of creating a full ontology covering a specific product, simple properties are used to establish references to the concepts contained in the external standards. The downside of this approach is that it does not support any inferences. However, it is fairly easy to later merge the information with semantically richer constructs as soon as they exist. This way, it is also possible to create application-specific, rich ontologies for a limited product range when needed.

One open question in ontology research that is of special importance for products and services is how the underlying consensus is reached (cf. Ding et al. 2002). It has already been proposed that existing e-commerce standards should be reused as building blocks for e-commerce ontologies, because of the domain knowledge therein pertained, inherent consensus and support of user communities (Corcho and Gómez-Pérez 2001, Zhao and Lövdahl 2003). Additionally, an ontology based on the reuse of existing standards will be more likely to gain acceptance (Zhao and Lövdahl 2003). In other words, it might inherit the consensus and authority from the social process that led to the development of the underlying standard. This is the case in our proposal.

One might argue that the pure reference to flat concepts in external standards provides little help for reasoning about products in the Semantic Web. However, our approach eases the development of intelligent applications significantly, because it decouples the two spheres of product data annotation and the development of richly axiomatized vertical ontologies for small product domains. Also, useful lightweight applications can be developed right away. For example, it will be fairly simple to create a service that can determine whether a specific product (e.g. a camera model) fits into a specific bag. This would only require an ontology that integrates the concepts “width”, “height”, and “length” between various attribute libraries (e.g. eCl@ss ↔ eOTD-EGAS).

The key advantage of our lightweight approach is that the tremendous effort required to develop a fully-fledged ontology can be limited to the narrow scope of a specific use. The economics of ontology development and maintenance have not been sufficiently analyzed yet, and it might be promising to allow market forces to find out which ontologies justify the cost of their development. From an economic perspective, it is not likely that formalizing the whole world will be an efficient allocation of resources. Yet today, some eBay users earn money by bridging semantic gaps: They explicitely search for offers that contain misspelled product terms („labtops“, „camras“, „dimonds“) and assume that the seller will accept a lower price because the number of potential buyers who find his or her offer is limited (o.V. 2004). Such forces must be unleashed to help build the Semantic Web.

6 CONCLUSION

As of today, companies can only use either natural language or classification systems to describe their products and services, and as the classification systems frequently do not cover their needs, they do not start with the important task of semantically enriching their data, which will make it hard for them to engage in the Semantic Web. It is especially disadvantageous that class-centric product description usually limits the range of choices to two options: Either the standard contains a suitable class and one can employ this, or it does not. If there is no suitable class available, one cannot use existing components to create a representation for a new or very specific type of product

The Semantic Web will require that companies need a solution soon and cannot wait for a global standard to be available (cf. Schulten et al. 2001). We have shown that a property-centric approach of product description offers several advantages. Especially, a company will not have to wait for a new class to be added and can instead use existing attributes. For example, the attributes for width, length and height from the eCl@ss attribute library can be used for the description of any tangible, rectangular product, even if no suitable product class exists. A “bag finder” application could easily reason about those attributes and determine a suitable bag for this product.

ACKNOWLEDGEMENTS

The work presented in this paper is partly funded by the European Commission under the project DIP (FP6-507483), the TransIT Entwicklungs- und Transfercenter at the University of Innsbruck, and Florida Gulf Coast University.

REFERENCES

Agrawal, R. and R. Srikant (2001). On integrating catalogs. The tenth international World Wide Web conference on World Wide Web, Hong Kong, ACM Press.

Berners-Lee, T., J. Hendler and O. Lassila (2001). The Semantic Web. Scientific American 284 (5), 28-37.

Corcho, O. and A. Gómez-Pérez (2001). Solving Integration Problems of E-commerce Standards and Initiatives through Ontological Mappings. Workshop on E-Business and Intelligent Web at the Seventeenth International Joint Conference on Artificial Intelligence (IJCAI-2001), Seattle, USA.

Di Noia, T., E. Di Sciascio, F. M. Donini and M. Mongiello (2003). A System for Principled Matchmaking in an Electronic Marketplace. Twelfth International World Wide Web Conference (WWW2003), Budapest, Hungary.

Ding, Y., D. Fensel, M. Klein and B. Omelayenko (2002). The Semantic Web - Yet Another Hip? Data & Knowledge Engineering 41 (2-3 (June 2002)), 205-227.

Fensel, D. (2004). Ontologies: A Silver Bullet for Knowledge Management and Electronic Commerce. Berlin etc., Springer.

Fensel, D., Y. Ding, B. Omelayenko, E. Schulten, G. Botquin, M. Brown and A. Flett (2001). Product Data Integration in B2B E-Commerce. IEEE Intelligent Systems 16 (4), 54-59.

Guo, J. and C. Sun (2003a). Collaborative Product Representation for Emergent Electronic Marketplace. 16th Bled eCommerce Conference eTransformation, Bled (Slovenia).

Guo, J. and C. Sun (2003b). Context Representation, Transformation and Comparison for Ad Hoc Product Data Exchange. DocEng'03: 2003 ACM Symposium on Document Engineering, Grenoble, France, ACM Press.

Hepp, M. (2003). Güterklassifikation als semantisches Standardisierungsproblem.

Hepp, M. (2004). Measuring the Quality of Descriptive Languages for Products and Services. E-Business - Standardisierung und Integration. Tagungsband zur Multikonferenz Wirtschaftsinformatik 2004. F.-D. Dorloff, J. Leukel and V. Schmitz. Göttingen, Cuvillier, 157-168.

Hepp, M. (2005a). A Methodology for Deriving OWL Ontologies from Products and Services Categorization Standards. 13th European Conference on Information Systems (ECIS2005), Regensburg, Germany.

Hepp, M. (2005b). Representing the Hierarchy of Industrial Taxonomies in OWL: The gen/tax Approach. ISWC Workshop Semantic Web Case Studies and Best Practices for eBusiness (SWCASE05), Galway, Irland.

Hepp, M., J. Leukel and V. Schmitz (2005a). Content Metrics for Products and Services Categorization Standards. IEEE International Conference on e-Technology, e-Commerce and e-Service (EEE-05), Hong Kong, IEEE.

Hepp, M., J. Leukel and V. Schmitz (2005b). A Quantitative Analysis of eCl@ss, UNSPSC, eOTD, and RNTD Content, Coverage, and Maintenance. IEEE ICEBE 2005, Beijing, China, IEEE.

Klein, M. (2002). DAML+OIL and RDF Schema representation of UNSPSC. Available at: http://www.cs.vu.nl/~mcaklein/unspsc/ (Retrieved April 23).

Leukel, J., V. Schmitz and F.-D. Dorloff (2002a). Exchange of Catalog Data in B2B Relationships - Analysis and Improvement. Proceedings of IADIS International Conference WWW/Internet 2002 (ICWI 2002). Lisbon, Portugal, 403-410.

Leukel, J., V. Schmitz and F.-D. Dorloff (2002b). A Modeling Approach for Product Classification Systems. 13th International Workshop on Database and Expert Systems Applications (DEXA'02), Aix-en-Provence, France, IEEE Computer Society.

McGuinness, D. L. (2001). UNSPSC Ontology in DAML+OIL. Available at: http://www.ksl.stanford.edu/projects/DAML/UNSPSC.daml (Retrieved November 5).

Ng, W. K., G. Yan and E.-P. Lim (2000). Heterogeneous Product Description in Electronic Commerce. ACM SIGEcom Exchanges 1 (1), 7-13.

o.V. (2004). Awkshun Serchs. Communications of the ACM 47 (4), 10.

Obrst, L., R. E. Wray and H. Liu (2001). Ontological Engineering for B2B E-Commerce. International Conference on Formal Ontology in Information Systems (FOIS'01), Ogunquit, Maine, USA, ACM Press.

Omelayenko, B. and D. Fensel (2001). A Two-Layered Integration Approach for Product Information in B2B E-commerce. Second International Conference on Electronic Commerce and Web Technologies (EC WEB-2001), Munich, Germany, Springer.

Schulten, E., H. Akkermans, G. Botquin, M. Dörr, N. Guarino, N. Lopes and N. Sadeh (2001). The E-Commerce Product Classification Challenge. IEEE Intelligent Systems 16 (4), 86-89.

United Nations Development Programme (2003). Business Function Identifiers (BFI). Available at: www.un-spsc.org/AdminFolder/documents/BFI.doc (Retrieved 19.04.2003).

Zhao, Y. and J. Lövdahl (2003). A Reuse-Based Method of Developing the Ontology for E-Procurement. Nordic Conference on Web Services (NCWS), Växjö, Sweden.