Towards Effectiveness and Transparency in e-Business Transactions

-An Ontology for Customer Complaint Management

Mustafa Jarrar[1,2]

[1]HPCLab, University of Cyprus, Cyprus
[2]STARLab, Vrije Universiteit Brussel, Belgium
mjarrar@vub.ac.be | mjarrar@cs.ucy.ac.cy

Abstract

This chapter presents an ontology for customer complaint management, which has been developed in the CCFORM project. CCFORM is an EU funded project (IST-2001-38248) with the aim of studying the foundation of a central European customer complaint portal. The idea is that any consumer can register a complaint against any party about any problem, at one portal. This portal should: support 11 languages, be sensitive to cross-border business regulations, dynamic, and can be extended by companies. To manage this dynamicity and to control companies' extensions, a customer complaint ontology (CContology) has to be built to underpin the CC portal. In other words, the complaint forms are generated based on the ontology. The CContology comprises classifications of complaint problems, complaint resolutions, complaining parties, complaint-recipients, "best-practices", rules of complaint, etc. The main uses of this ontology are 1) to enable consistent implementation (and interoperation) of all software complaint management mechanisms based on a shared background vocabulary, which can be used by many stakeholders. 2) to play the role of a domain ontology that encompasses the core complaining elements and that can be extended by either individual or groups of firms; and 3) to generate CC-forms based on the ontological commitments and to enforce the validity (and/or integrity) of their population. At the end of this chapter, we outline our experience in applying the methodological principles (Double-Articulation and Modularization) and the tool (DogmaModeler) that we used in developing the CContology.

Keywords: e-Commerce, CRM, Customer Relationship management, Customer Complaints Forms, Ontology, Customer Complaint Ontology, Semantics, Domain Axiomatization, Multilingual Ontology, Ontology Engineering, Methodology, Double Articulation, Modularization Context, Gloss, Lexon, DogmaModeler.

1 Introduction and Motivation

1.1 Current Situation

The use of the Internet for cross-border business is growing rapidly. However, in many cases, the benefits of electronic commerce are not exploited fully by customers because of the frequent lack of trust and confidence in online cross-border purchases. To achieve fair trading and transparency in commercial communications and transactions, effective cross-border complaint platforms need to be established and involved in e-business activities (Claes, 1987) (Cho et al, 2002) (ABA, 2002).

The CCFORM project aims to study and *reach a consensus* about the foundation of online customer complaint mechanisms by developing a general but extensible form (called CC-form[1]) which has widespread industry and customer support. This CC-form must facilitate cross-language communication

---

[1] We refer to the project name as "CCFORM" and to a customer complaint portal as "CC-form". One may imagine a CC-form as several pages of web forms, which can be dynamic and filled in several steps.

to support cross-border e-commerce and should be easy to implement in software tools. The CC-form will raise the basic agreement about complaint handling, and should be extended in vertical markets (e.g. hotels, banks, factories, or even governments) to provide sector-wide solutions to allow service providers to gain competitive advantages (See Fig. 1).

1.2 Problem Statement

There are several challenges involved in establishing and agreeing on such a CC-form: (1) Legal bases: the sensitivity of cross-border business regulations and privacy issues. (2) The diversity of language and culture: controlling and agreeing on the semantics of the complaint terminology so that the intended meaning of the term gets across, even in the different languages. (3) Consumer sensitivity and business perspectives. (4) Extensibility: the flexibility of extending the CC-form (perhaps dynamically) according to market needs and standards. This would mean for example, extending the kinds of problems that a complainant can complain about, extending the kinds of complaint resolutions, managing who may extend what, etc.
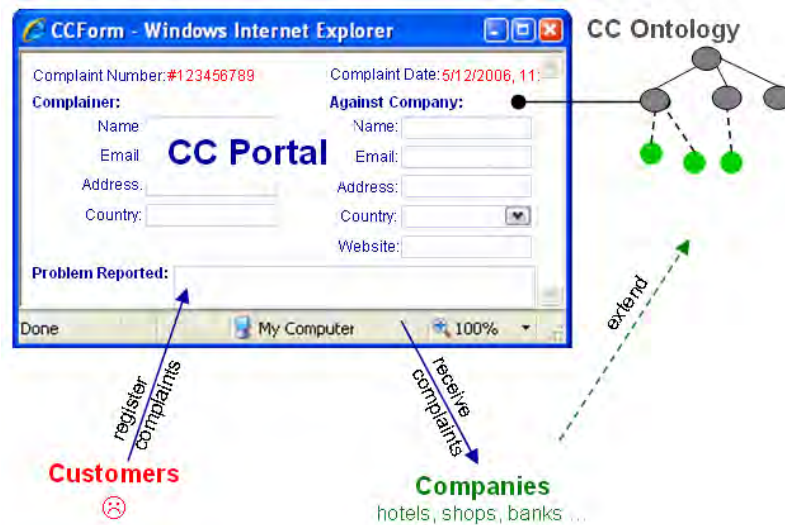

Fig. 1. Depiction of the CCform design.

1.3 Proposed Solution

In order to tackle such challenges and to perfect the reference model for a CC-form, the research has been divided into six interest groups, each consisting of 10-15 highly specialized members. Each group has been intensively discussing different issues: SIG1- Legal Affairs, SIG2- Consumer Affairs, SIG4 - Standards for SMEs, SIG5 -Alternative Dispute Resolution Systems, SIG6 - Ontology and Extensibility, SIG7 - Vertical markets.

The work presented in this chapter outlines our main achievements in the "Ontology and extensibility" group, including multilingual and cultural issues (Jarrar et al, 2003). The mission of this group, SIG6, is to undertake extensibility and multilingual demands. To approach this, a customer complaint ontology (CContology) has been developed and lexicalized in 11 European languages. *This CContology is developed and reviewed by the six interest groups, and is seen as a conceptual framework that is necessary to develop such a CC-form.*

In the following section, we present the CContology itself and the methodology we applied to engineer it. In section 3, we provide some lessons learnt and a discussion about our applied engineering solutions. Section 4 presents a multilingual lexicalization methodology. To end, section 6 presents our conclusions and directions for future work.

2 Customer Complaint ontology

In this section we introduce the *customer complaint ontology* (CContology) that is intended to capture the main concepts in the "customer complaint management" domain. Its core covers a semantic description of

complaints that could be issued by any legal person against any other legal person (NGO, company, natural person, etc.). The CContology comprises classifications of complaint problems, complaint resolutions, complainant, complaint-recipient, "best-practices", rules of complaint, etc.

The intended impact of this research is the future initiation of a European online complaint platform that will provide a trusted portal between consumers and business entities. In this respect, the ontology is intended to become the basis for a future *core ontology* in the domain of customer complaint management (for both humans and machines). Applying the CContology in such an European online complaint platform will facilitate further refinements of the CContology.

The main uses of such an ontology are 1) to enable consistent implementation (and interoperation) of all software complaint management mechanisms based on a shared background vocabulary, which can be used by many stakeholders. 2) to play the role of a *domain ontology* that encompasses the core complaining elements and that *can be extended by either individuals or groups of firms*; and 3) to generate CC-forms based on the *ontological commitments* and to enforce the validity (and/or integrity) of their population.

Although this CContology has been developed and reviewed by six specialized groups, in its current state, it can only be considered a proposal. The CCFORM community is representative of a sizable cross-section of the domain but is not a standardization body. Nor is it in the position to insist on a *de facto* enforcement of this ontology as a generally agreed semantic specification. However, the approach presented in this chapter is designed to initiate and drive such a process.

2.1   The Applied Engineering Methodology

The CContology is developed according to the two methodological principles that we developed in (Jarrar, 2005): 1) ontology double articulation, and 2) ontology modularization. See figure 2.

The ontology double articulation principle suggests, in nutshell, that: *an ontology is doubly articulated into: domain axiomatization and application axiomatizations. While a domain axiomatization is mainly concerned with characterizing the "intended models" of a vocabulary at the domain level (typically shared and public), an application axiomatization (typically local) is mainly concerned with the usability of this vocabulary according to certain application/usability perspectives. An application axiomatization is intended to specify the legal models (a subset of the intended models) of the application(s)' interest.*

To simplify the double articulation principle, in other words, one can imagine WordNet as a domain axiomatization, and an application axiomatization built in OWL (or RDF, ORM, UML, etc). The double articulation principle then suggests that all vocabulary in the OWL axiomatization should be linked with word-senses (i.e. concepts) in WordNet. In this way, we gain more consensus about application axiomatizations as it is rooted at the domain level; we improve the usability of application axiomatizations as they are specific, and the reusability of domain axiomatizations as the are generic; application axiomatizations that are built in the same way (i.e. commit to the same domain axiomatization) will be easier to integrate, and so forth. See (Jarrar, 2005), (Jarrar et al, 2008b), and (Jarrar, 2006) for more details.

The modularization principle suggests that application axiomatizations be built in a modular manner. Axiomatizations should be developed as a set of small modules and later composed to form, and be used as, one modular axiomatization. A composition operator is defined for a full automatic composition of modules (see (Jarrar, 2005) and (Jarrar, 2005b)). It combines all axioms introduced in the composed modules. As shall be discussed later, the idea of the modularization principle is that modules are easier build, maintain, use, and reuse.
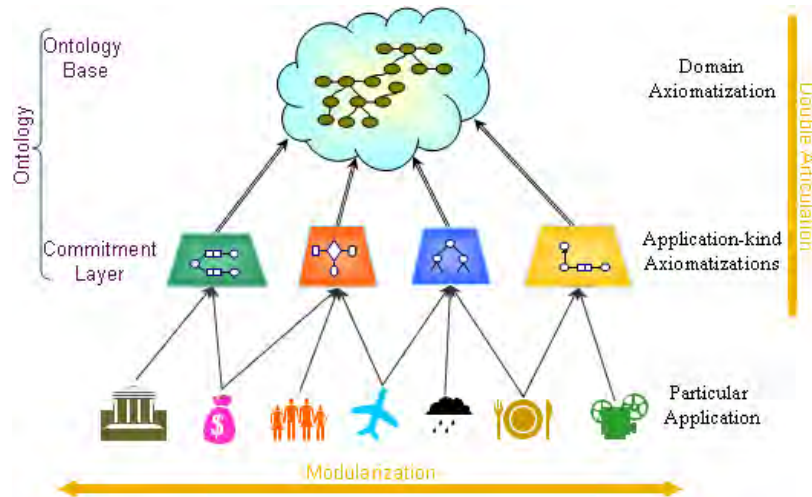
Fig. 2. Depiction of applied methodology (Jarrar, 2005) (Jarrar et al, 2003c).

The CContology is built according to the above methodological principles. It consists of a domain axiomatization (the lexons, context, and the term glossary) and seven application axiomatization modules, (Complaint Problems, Complaint Resolutions, Complaint, Complainant, Complaint-Recipient, Address, and Contract). See figure 3. Applications (such as the CCform) use the composition of these 7 modules. Notice that the CCform (as an application) is committing to the domain ontology through this composition (i.e. the 7 application axiomatizations). The advantages of this methodology will be discussed in later sections. In the following we present each component of the CContology. The full content of the CContology can be accessed from the CContology-webpage[2]. In the following we present some samples of this ontology.
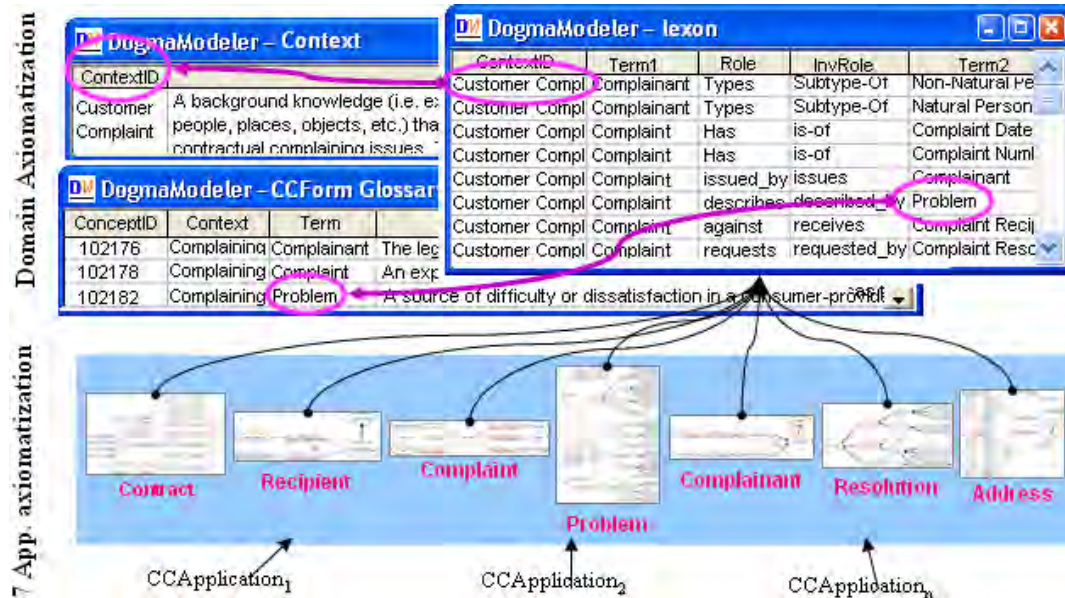

Fig. 3. Depiction of CContology components.

2.2 The Domain Axiomatization

This part of the CContology consists of three representation units: 1-the "customer complaint" context, 2- domain vocabularies and their glosses, and 3- the set of lexons. A lexon is a binary relationship between context-specific linguistic terms, or in other words, a lexical rendering of a binary conceptual relation.

2.2.1 The "Customer Complaint" Context

The notion of context is the first building block in our methodology for developing a domain axiomatization. It plays a *scoping role*, through which the interpretation of the intended meaning of the ontology terminology is bounded. We say a term *within* a context refers to a concept, or in other words, *that context is an abstract identifier that refers to implicit (or maybe tacit) assumptions, in which the interpretation of a term is bounded to a concept.* In practice, we define context by referring to a source (e.g. a set of documents, laws and regulations, informal description of "best practices", etc.), which, by *human understanding*, is assumed to "contain" those assumptions.

In the CContology, the "Content ID" is called the "*Customer Complaint*" context, or the *CCcontext in short*. The "Context Description" is defined in the following table :

Background knowledge (i.e. explicit, implicit, or tacit assumptions) about all (activities, communications, institutions, people, places, objects, etc.) that are involved in consumer-provider relationships, regarding contractual and non-contractual complaining issues.

These assumptions can be understood (i.e. can be found explicitly or intuitively) in the following sources:

- *European Distance Selling Directive (97/7/EC), on the promotion of consumers in respect of distance contracts.*

- *European e-Commerce Directive (2000/31/EC) on certain legal aspects of information society services, in particular, electronic commerce, in the Internal Market.*

- *European Data Protection Directives (95/46/EC and 97/66/EC) on the protection of individuals with regards to the processing of personal data and on the free movement of such data.*

- *European Directive (99/44/EC) on aspects of the sale of consumer goods and associated guarantees.*

- *European Directive (98/27EC) on Injunctions for the Protection of Consumers' Interests.*

- *CEN/TC331 Postal Services EN 14012:2002 Quality of Service – Measurement of complaints and redress procedures.*

- *"Best practice" guidelines, The Nordic Consumer Ombudsmen's position paper on trading and marketing on the Internet and other similar communication systems(http://econfidence.jrc.it, June 2002)*

- *CCFORM Annex 1, (IST-2001-34908, 5th framework).*

- *CCFORM Report On Copyright And Privacy Recommendations (Deliverable D.5.3).*

- *CCFORM user guide and business complaints (Deliverable D.5.1.1).*

- *CCFORM Company user guide (Deliverable D.5.1.2).*

- *CCFORM Web publication of CCform User Guides in 11 languages (Deliverable D6.11).*

- *Code of Conduct (CCFORM deliverable).*

Remark: For the sake of brevity, many resources (regulations at the European and national levels, best practices, existing online complaining (plat)forms, etc.) are not mentioned here. However, references to these resources can be found inside the resources listed above.

Table 1. The definition of the "Customer Complaint" context.

We have learned during the definition process that it is not an easy task to create such a context, and a context cannot be defined rigidly in the early phases of the development of such an ontology. As none of our team was an ontology expert, we provided a draft definition and investigated by providing many different examples of application scenarios that this context should cover. This investigation was done to prevent the CContology from being dependent on the CC-form application scenario which the team had in mind during the early phases. For example, we questioned whether the context should cover applications such as customer-relationship-management, market analyses, sales force automation and so forth; whether it should cover all consumer regulations in any country or only in Europe; whether it should cover all commercial activity, in any place and at any time; which documents, laws and regulations should be our main references, etc. Such questions led not only to the CCcontext definition

(which was achieved after several iterations), but also propelled the team to discuss deeply and even redefine the scope of the research goals.

2.2.2 Vocabularies and their Glosses

Within the "Customer Complaint" context, we define 220 terms. A sample of these terms and their glosses (Called the CCglossary) is provided in Table 2. This CCglossary was developed (and reviewed) over several iterations. The first iteration was accomplished by a few (selected) experts before the lexon modeling process was started (lexons are the third component of the CContology). Further iterations have been carried out in parallel with the lexon modeling process. The final draft was reviewed and approved by several groups. It is probably worth noting that intensive discussions were carried out (by legal experts, market experts, application-oriented experts) for almost every gloss. We have found that the gloss modeling process is a great mechanism for brainstorming, domain analyses, domain understanding and for reaching (and documenting) consensus. Furthermore, it allowed non-ontology experts to participate actively in the ontology modeling process. Some partners have remarked that the CCglossary is the most useful and reusable component in the CContology.

As shall be discussed later, this CCglossary, which has been developed in English, has played the role of *the key* reference for lexicalizing the CContology into 11 other European languages. The translators have acknowledged that it guided their understanding of the intended meanings of the terms and allowed them to achieve better translation quality. The following are the guidelines (Jarrar, 2006) that we used to deciding what should and should not be provided in a gloss.

1. It should start with the *principal/super type* of the concept being defined. For example, "Search engine: A computer program that ...", "Invoice: A business document that..."University: An institution of ...".

2. It should be written in the form of propositions, offering the reader inferential knowledge that helps him to construct the image of the concept. For example, instead of defining 'Search engine' as "*A computer program for searching the internet*", or "*One of the most useful aspects of the World Wide Web. Some of the major ones are Google, Galaxy... .*". One can also say "*A computer program that enables users to search and retrieve documents or data from a database or from a computer network...*".

3. More importantly, it should focus on distinguishing characteristics and intrinsic properties that differentiate the concept from other concepts. For example, compare the following two glosses of a 'Laptop computer': (1) "*A computer that is designed to do pretty much anything a desktop computer can do. It runs for a short time (usually two to five hours) on batteries*"; and (2) "*A portable computer small enough to use in your lap...*". Notice that according to the first gloss, a 'server computer' running on batteries can be seen as a laptop computer; also, a 'Portable computer' that is not running on batteries is not a 'Laptop computer'.

4. The use of supportive examples is strongly encouraged: (1) to clarify true cases that are commonly known to be false, or false cases that are known to be true; and (2) to strengthen and illustrate distinguishing characteristics (by using examples and counter-examples). The examples can be types and/or instances of the concept being defined. For example: "Legal Person: *An entity with legal recognition in accordance with law. It has the legal capacity to represent its own interests in its own name, before a court of law, to obtain rights or obligations for itself, to impose binding obligations, or to grant privileges to others, for example as a plaintiff or as a defendant. A legal person exists wherever the law recognizes, as a matter of policy, the personality of any entity, regardless of whether it is naturally considered to be a person. Recognized associations, relief agencies, committees and companies are examples of legal persons*".

5. It should be consistent with the formal axioms in the ontology. As glosses co-exist in parallel to the formal axioms, both should not contradict each other; any change on side should be reflected on the other.

6. It should be sufficient, clear, and easy to understand.

The CCglossary (*Sample*)	
Term	Gloss
Address	A construct describing the means by which contact may be taken with, or messages or physical objects may be delivered to; an address may contain indicators for a physical or virtual (i.e. accessed electronically) location or both.
Complainant	The legal person who issues a complaint.
Complaint	An expression of grievance or resentment issued by a complainant against a compliant-recipient, describing a problem(s) that needs to be resolved.

Complaint Recipient	A legal person to whom a complaint is addressed.
Complaint Resolution	A determination for settling or solving a problem in a consumer-provider relationship.
Contract	A binding agreement between two or more legal persons that is enforceable by law; an invoice can be a contract.
Contract Problem	Problem linked to a contract in a customer-provider relationship, it may occur before or after the contract effective date.
Contract Termination Problem	A problem concerned with the proper termination or completion of the contract.
Data Collection Problem	A privacy problem regarding all activities and purposes of private data collection
Delivery and Installation Problem	A purchase phase problem related to dissatisfaction regarding delivery or Installation of goods or services.
Delivery problem	A purchase phase problem related to dissatisfaction regarding the delivery of goods or services.
Evidence	(WordNet) all the means by which any alleged matter of fact whose truth is investigated at judicial trial is established or disproved
Guarantee Problem	An after sales service problem related to a legal or contractual guarantee; particularly a problem related to a responsibility on the recipient consequent to the guarantees directive.
Guarantee refused	Refusal to apply a legal or contractual guarantee.
Legal Person	An entity with legal recognition in accordance with law. It has the legal capacity to represent its own interests in its own name, before a court of law, to obtain rights or obligations for itself, to impose binding obligations, or to grant privileges to others, for example as a plaintiff or as a defendant. A legal person exists wherever the law recognizes (as a matter of policy). This includes the personality of any entity, regardless of whether it is naturally considered to be a person. Recognized associations, relief agencies, committees, and companies are examples of legal persons
Mailing Address	The address where a person or organization can be communicated with for providing physical objects. It is broadly equivalent to a postal address as described in standards CEN 14132 or UPU S42, but has different functional definition
Privacy Problem	A problem related to the collection, storage, handling, use or distribution of private data, violating the data protection directives.
Problem	A source of difficulty or dissatisfaction in a consumer-provider relationship.
Product delivery delayed	A delivery problem related to delay in product delivery.
Registration	A certification, issued by an administrative authority or an accredited registration agency, declaring the official enrolment of an entity. Typically, it includes the official name, mailing address, registration number, VAT number, legal bases, etc.
	….

Table 2. A sample of the "Customer Complaint" glossary.

2.2.3 Lexons

Stemming from the 220 terms within the "Customer Complaint" context, we have developed 300 lexons. A sample of these lexons can be found in Table 3. As we mentioned earlier, a lexon is a lexical rendering of a binary conceptual relation. A lexon (Jarrar, 2005) (Jarrar, 2002) (Meersman, 1999) is described as a tuple of the form: <*ContextID*: *Term₁*, *Role*, *InvRole*, *Term₂*>. Where *Term₁* and *Term₂* are linguistic terms from a language *L*; *Role* and *InvRole* are lexicalizations of the pair roles of a binary conceptual relationship R; *InvRole* is the inverse of the *Role*. For example, the pair roles of a *subsumption* relationship could be: "Is a type of" and "Has type"; the pair roles of a *parthood* relationship could be: "is a part of" and "has part", and so forth. See Table 3 for examples.

Table 3 shows a sample of lexons for the CContology[3]. Notice that "Customer Complaint" is defined in section 2.2.1, and that each term used in these lexons has a gloss as described in section 2.2.2. The first draft of the CC lexons has been developed based on presentations and discussions between the members of SIG 6 (Ontology and Extensibility). Most of these lexons represent taxonomies of complaint problems, complaint resolutions, complainant, complaint recipient, and addresses. One of the most important inputs, for the first draft, was the complaint categorization survey (Vassileva, 2003) that was performed by two of the group members. Further, refinements and investigations were performed during meetings and workshops that we organized in cooperation with other SIGs.

Context	Term1	Role	InvRole	Term2
Customer Complaint	Problem	Types	Subtype-Of	Contract Problem
Customer Complaint	Problem	Types	Subtype-Of	Non-Contract Problem
Customer Complaint	Problem	Types	Subtype-Of	Privacy Problem
Customer Complaint	Complaint	against	receives	Complaint Recipient
Customer Complaint	Complaint	describes	described_by	Problem
Customer Complaint	Complaint	issued_by	issues	Complainant
Customer Complaint	Complaint	requests	requested_by	Complaint Resolution
Customer Complaint	Contact Details	comprised_of	comprises	Address
Customer Complaint	Contact Details	Has	is-of	Name
Customer Complaint	Contract Problem	Types	Subtype-Of	Post-purchase Phase Problem
Customer Complaint	Contract Problem	Types	Subtype-Of	Pre-purchase Phase Problem
Customer Complaint	Contract Problem	Types	Subtype-Of	Purchase Phase Problem
Customer Complaint	Complaint Resolution	Types	Subtype-Of	Economic Resolution

---

[3]   The   full   content   of   the   CContology   can   be   downloaded   from http://www.starlab.vub.ac.be/staff/mustafa/CContology

Customer Complaint	Complaint Resolution	Types	Subtype-Of	Information Correction
Customer Complaint	Complaint Resolution	Types	Subtype-Of	Symbolic Resolution
Customer Complaint	Conduct	Types	Subtype-Of	Lewd or Immoral conduct
Customer Complaint	Conduct	Types	Subtype-Of	Untruthlness
		…		

Table 3. A sample of the "Customer Complaint" lexons.

2.3 The CC Application Axiomatizations

Given the previously presented "customer complaint" domain axiomatization, seven application axiomatization modules have been developed. *The intended meaning of the terminology used in these modules (i.e. application axiomatizations) is restricted to the terminology defined at the domain axiomatization level*. Figure 4 depicts this relationship between a concept used at the application level and its general meaning at the domain level (this way of expressing knowledge is called *double-articulation* (Jarrar, 2005)).
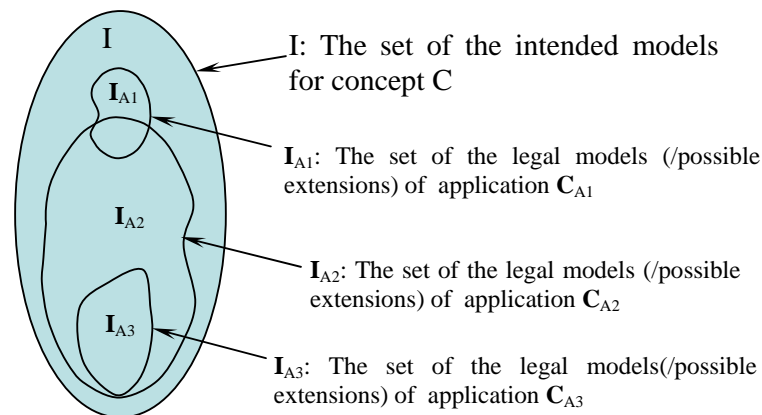

I: The set of the intended models for concept C

I_{A1}: The set of the legal models (/possible extensions) of application C_{A1}

I_{A2}: The set of the legal models (/possible extensions) of application C_{A2}

I_{A3}: The set of the legal models(/possible extensions) of application C_{A3}

Fig. 4. An example of three different applications specializing a domain concept.

Given a concept **C** as *a set of rules (e.g. described by lexons and glosses) in our mind about a certain thing in reality*, the set **I** of "all possible" instances that comply with these rules are called the *intended models* of the concept **C**. According to the ontology double articulation principle, such concepts are captured at the domain axiomatization level. An application A_i that is interested in a subset I_{Ai} of the set **I** (according to its usability perspectives), is supposed to provide some rules to specialize **I**. In other words, every instance in I_{Ai} must also be an instance in **I**. We call the subset I_{Ai}: the *legal models* (or extensions) of the application's concept C_{Ai}. Such application rules are captured at the application axiomatization level[4].

The application axiomatization modules are intended to play the role of conceptual data schema(s) for CC-forms development. Any CC-form, including its population, should be based on (i.e. commit to) the CContology through those modules. A CC-from can be constructed manually or generated automatically; nevertheless, the semantics of all elements in this CC-from (i.e. the data fields) is defined in the CContology. See (section 6.7.1 in (Jarrar, 2005)) on how to generate a web form automatically out of a given ORM schema

---

[4] The differences between the legal models of these application-types illustrate their different usability perspectives: (First) the intersection between the legal models of C_{A2} and the legal models C_{A3} shows that I_{A3} is a subset of I_{A2}. An example of this case could be the difference between notions of 'book' in an axiomatization of bookstores and libraries: all legal instances of the bookstores' notion are legal instances for the libraries, but not vice versa. For libraries, the instances of e.g. 'Manual' or 'Master Thesis' can be instances of a 'book'; however, they cannot be instances of 'book' for bookstores, unless they are published with an 'ISBN'. (Second) The difference between I_{A1} and I_{A3} shows an extreme case: two types of applications sharing the same concept C while their legal models are completely disjoint according to their usability perspectives. An example of this case could be the difference between notions of 'book' in an axiomatization of bookstores' and museums': Museums are interested in exhibiting and exchanging instances of old 'books', while bookstores are not interested in such 'books', unless for example, they are re-edited and published in a modern style.

As stated earlier in this chapter, the seven application axiomatization modules are: Complaint problems, Complaint resolutions, Contract, Complaint, Complainant, Complaint Recipient, and Address. Depending on an application's usability requirements, these modules can be used individually or composed to form a modular axiomatization(s). In the following, we provide a brief description of each module.

These application axiomatization modules below are represented using the ORM (Object Role Modeling) notation. We have found this notation is not only expressive, but also from a methodological viewpoint, ORM can be verbalized into natural language sentences (Jarrar et al, 2006b). This verbalization capabilities was a great help for our community (who are not IT savvy) to still be able to build and review the CContology without needing to know its underpinning logic or reasoning (Jarrar et al, 2006) (Jarrar et al, 2008). ORM has well-defined semantics (Halpin, 1989) (Halpin, 2001) in first order logic, and recently we have mapped ORM into the SHOIN/OWL (Jarrar, 2007b) (Jarrar et al, 2007) and the DLR (Jarrar, 2007) description logics.

2.3.1 Complaint Problems

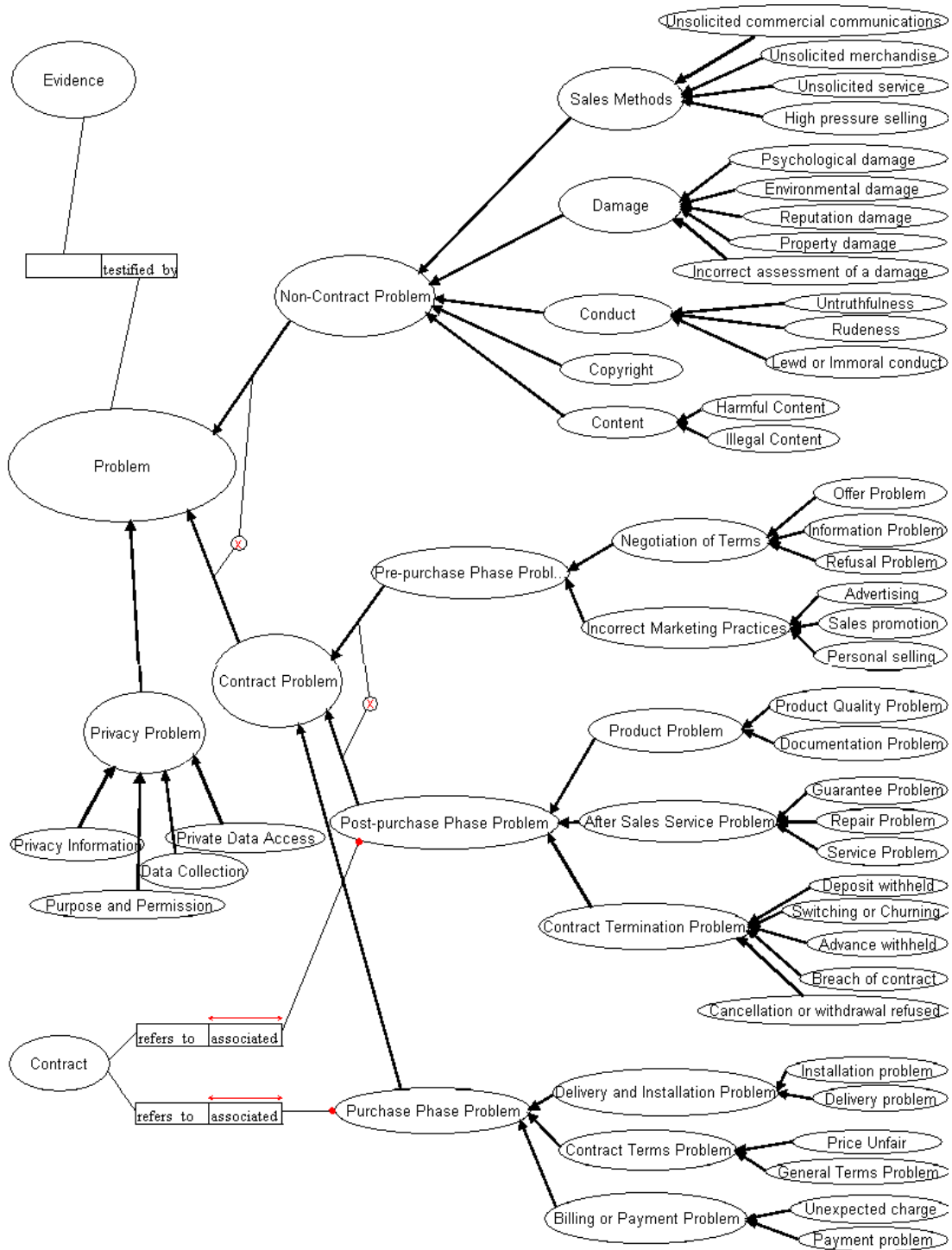Fig. 5 shows the "Complaint Problems" axiomatization module. It is a taxonomy of complaint problems.

Fig. 5. The "Complaint Problems" application axiomatization module, in ORM[5].

We distinguish between a 'Complaint' and a 'Problem'. A 'Complaint' *describes* one or more 'Problems'. While the concept 'Problem' is defined as "A source of difficulty or dissatisfaction", the concept 'Complaint' is defined as "An expression of grievance or resentment issued by a complainant against a compliant-recipient, describing a problem(s) that needs to be resolved".

Within the "customer complaint" domain, a 'Problem' can be a 'Privacy Problem', or either a 'Contract Problem' or a 'Non-contract Problem'. A 'Contract Problem' can be a 'Purchase Phase Problem', or either a 'Pre-purchase Phase Problem' or a 'Post-purchase Phase Problem'. It is mandatory for both 'Purchase Phase Problems' and 'Post-purchase Phase Problems' to be associated with a 'Contract'[6]. For any type of problem, 'Evidence' might be provided for investigation purposes.

Remark: In this "Complaint Problems" module, only four classification levels are presented, all of which are the popular categories in most CC-forms. Further classifications of complaint problems can be found at the ontology base level.

2.3.2 Complaint resolutions

Fig. 6 illustrates the "Complaint Resolution" module, which presents a taxonomy of 'Complaint Resolutions'. A 'Complaint Resolution' is defined in the CCglossary as "A determination for settling or solving a complaint problem(s)". It can be requested by a complainant or offered by a complaint-recipient. A 'Complaint Resolution' can be an 'Economic Resolution', a 'Symbolic Resolution', or an 'Information Correction'.
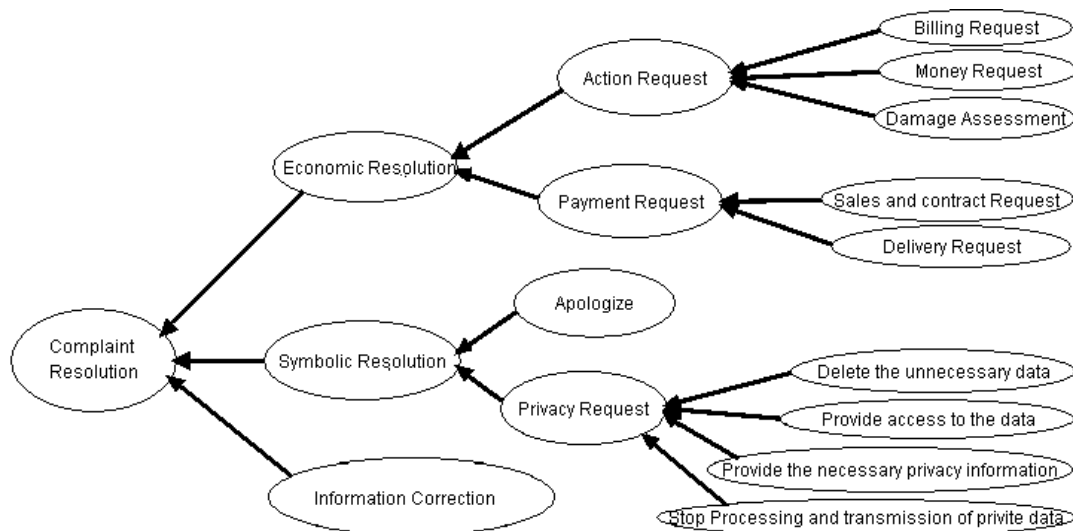

Fig. 6. The "Complaint Resolutions" application axiomatization module, in ORM.

2.3.3 Contract

A 'Contract' is defined in the CCglossary as "a binding agreement, between two or more legal persons, that is enforceable by law". Under this definition, an invoice can also be a contract. Fig. 7 illustrates the "Contract" module, which specifies the information that should be provided for a contract associated with a 'Purchase Phase Problem' or 'Post-purchase Phase Problem'. Notice that, for a CC-form, we speak of a 'Contract' from the moment there is a 'Contract Order Date'.

---

[6] Notice that the mandatory constraint cannot be generalized at the domain level, because there might be other types of applications where it is not mandatory to be associated with a contract, at least explicitly.
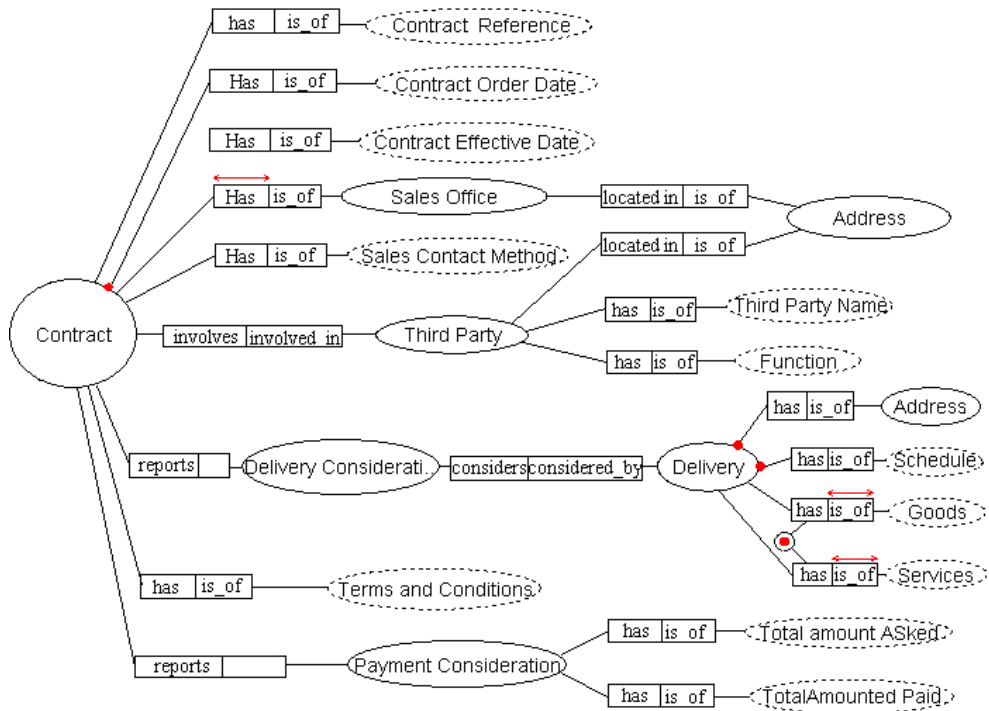
Fig. 7. The "Contract" axiomatization module, in ORM[7].

2.3.4 Complaint

A 'Complaint' is defined in the CCglossary as "An expression of grievance or resentment issued by a complainant against a compliant-recipient, describing a problem(s) that needs to be resolved".

Fig. 8 illustrates the "Complaint" axiomatization module, which specifies the main concepts that can be associated with the concept 'Complaint'. A 'Complaint' must be issued by a 'Complainant' against a 'Complaint-Recipient', on a certain 'Date'. It must describe at least one 'Problem', and may request one or more 'Complaint Resolutions'. A 'Complaint' *might be* identified by a 'Complaint Number', which is typically used as a unique reference in a court or a complaint system.
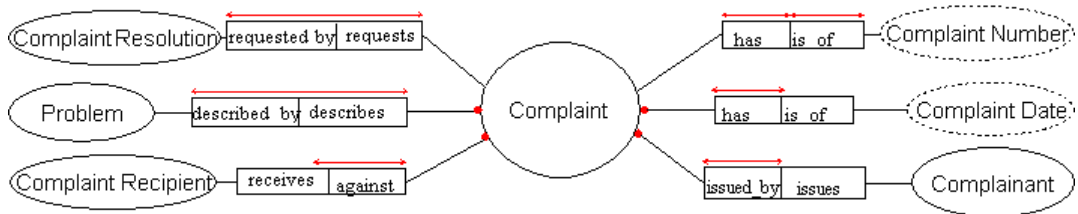

Fig. 8. The "Complaint" application axiomatization module, in ORM.

2.3.5 Complainant

Fig. 9 illustrates the 'Complainant' axiomatization module. A 'Complainant' is defined in the CCglossary as "A legal person who issues a complaint". In the customer complaint context, and as commonly understood in most consumer regulations, a complainant must either be a 'Natural Person Complainant' or a 'Non-Natural Person Complainant', each implying a different legal basis for the handling of the complaint.

---

[7] The ORM exclusion constraint ⊙ states that each *Delivery* should be at least *Goods* or *Services*, or both.
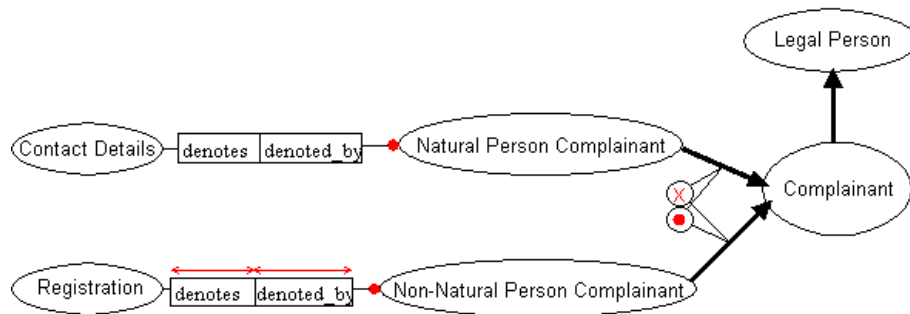
Fig. 9. The "Complainant" application axiomatization module, in ORM.

The distinction between natural and non-natural person complainants is not only based on the variation of their complaint handling regulations, but also on the legal preference (in any CC-from) for not obligating the inquiry of private information about the 'Natural Person Complainant', such as his/her 'Name', 'Birth Date', 'Mailing Address', 'Religion' etc. Each 'Natural Person Complainant' must have 'Contact Details'. The mandatory contact details (as agreed with the "customer complaint" community, but which cannot be generalized for all communities) are an 'eMail' and his/here 'Country' of residence. A 'Non-Natural Person Complainant' must be denoted by a certain 'Registration' that identifies him in a CC-form. See the definition of 'Registration' in the CCglossary.

2.3.6 Complaint recipient

Fig. 10 illustrates the "Complaint Recipient" axiomatization module. A 'Complaint Recipient' is any legal Person to whom a complaint is addressed. Typically, when a 'Complaint' is issued against a 'Complaint Recipient', the 'Contact Details' *or* the 'Registration' of this 'Complaint Recipient' should be denoted. Usually, all online customer complaint platforms provide a searchable database of many "Complaint Recipients", which enables complainants to easily find the official names and addresses of 'complaint recipients'.
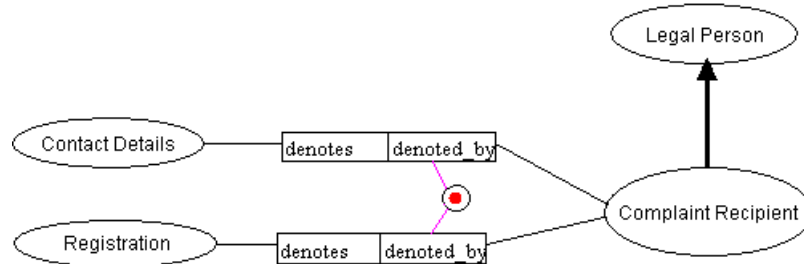

Fig. 10. The "Recipient" application axiomatization module, in ORM.

2.3.7 Address

Fig. 11 illustrates the "Address" axiomatization module. The concept 'Contact Details', which is a channel of communication, is attributed by both 'Name' and 'Address'. An 'Address' must be either an 'Electronic Address' or a 'Mailing Address'. An 'electronic Address' can be either a 'Web Site', 'Telephone', 'eMail', or 'Fax'. A 'Mailing Address' can have all the traditional information of postal addresses in the European Union.

Remark: Due to epistemological differences, the notion of 'Address' can be specified in many different ways, especially since each country has its own postal information structure. Hence, this "Address" axiomatization module is considered an "unsteady" module, and should be replaced by a more sophisticated module – one that does, for example, consider the compatibility with online national, European, or international address servers[8].

---

[8] Such address servers are: http://www.afd.co.uk/tryit/ (July 2007), http://www.postdirekt.de (July 2007), http://www.usps.com, (July 2004).
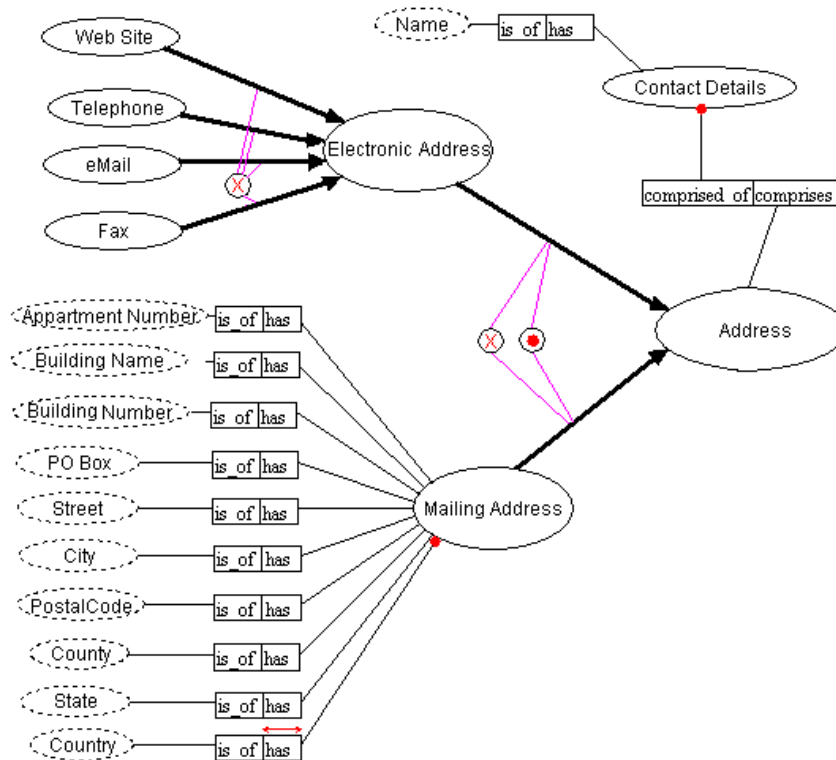
Fig. 11. The "Address" application axiomatization module, in ORM.

3 Discussion and Lessons Learned

This section provides a further discussion on the application of our methodological principles for the development and engineering of the CContology.

Extensibility is one of the main requirements (and one of the most challenging issues) for the development of any CC-form. As we have mentioned earlier, our main goal is to reach consensus about the foundation of a trusted customer complaint portal. Once such a portal is implemented as a centralized CC-form between customers and companies, companies may wish to extend "their" CC-form to inquire about more specific complaint details, e.g. delivery conditions, certain product attributes, or they might wish to offer the customer a particular resolution, etc[9]. Such extensions may be a necessity not only for individual companies but also in so called vertical markets applications (discussed by the "vertical market" group, SIG7). The idea is to allow companies to extend the CC-form content themselves. On the one hand, this will help to achieve a wider adoption of complaint mechanisms in e-commerce applications. On the other hand, this will create new challenges such as keeping the new extensions consistent with the existing CC-form and preventing the misuse of the CC-form. For example, a company might try to misuse the CC-form by inquiring about private information that violates the privacy regulations, or it may introduce new terminology and rules that are semantically inconsistent with the existing content terminology and rules.

As a solution, we propose that the CC-form not be altered directly. Instead, extensions should be introduced first into the CContology, the base of CC-form (see figure 1). Moreover, our modularization of the application axiomatization part of the CContology offers simplified methodologies for extending, maintaining, and managing the CC-form:

- *Extensions will not be allowed on all axiomatization modules*. For example, the "Complainant" and "Address" axiomatization modules may be "locked", so companies will be prevented from, for example, asking privacy-rule-violating questions. Or perhaps, we can only allow extensions to be

---

[9] One can imagine a company providing a link to the CC-form portal in their own webpage. When the link is clicked, the CC-form appears with the company's information filled and the details of the complaints (that are specific to this company) attached to the basic complaint questions.

made into the "Complaint Problems" and "Complaint Resolutions" modules. In this way, we can achieve a "relatively" systematic management of the kinds of extensions allowed.

- *Extensions can be made and treated as separate modules*. If a company wishes to extend one of the seven core modules to inquire details about, for example, a certain kind of product, a new module can be constructed to capture these details. Both the core module(s) and the new module can be composed automatically. Notice that the specification of our composition operator (see section 4.4.2 in (Jarrar, 2005)) guarantees that the constraints and the complaining details introduced in a core module will never be dismissed or weakened. In other words, the constraints and complaint details in the resultant composition will always imply the constraints and the complaint details in the core module[10].

- *Efficient maintenance and management*. A CC-form platform may need to manage a large number of extensions that target many different complaining issues. Releasing and treating these extensions as separate modules will make managing, maintaining, and indexing them more scalable.

- *The development of the modules can be distributed among ontology experts, domain experts and application-oriented experts*. In the case of a vertical market application where one wishes to develop a set of extensions (i.e. modules), the development and the review processes can be distributed according to the expertise of the developers and the subject of the modules. In the development of the seven core modules we have distributed the development and review between several specialized groups in accordance with their expertise. Bistra Vassilev acted as domain expert for the development of the complaint problem and resolutions modules, even though she was based several thousand kilometers away. Members from SIG1 (legal affairs) have contributed to the development and review of the "Complaint", "Complainant", "Complaint Recipient", "Address" and "Contract" modules. Members from SIG2 (consumer affairs) have similarly contributed to the development and review of the "Complaint", "Complainant", "Complaint Problem" and "Complaint Resolution" modules, etc.

- *Reusability of domain axiomatizations.* Notice that our methodology requires domain knowledge (lexons and glosses) to be developed first, then this knowledge can be used at the application level. In this way we improve the reusability of the domain knowledge because application developers will be forced to investigate and use what already exists at the domain level.

- *Reusability of the Module*. Modularizing the application axiomatization of the CContology indeed simplifies the reusability of this axiomatization. One may wish to reuse some of these axiomatization modules in application scenarios other than the CC-form. For example, the 'Address' module can easily be reused for tasks in other domains such as Mailing, Marketing and Sales Force Automation. The `Complaint Problems' module is in the domains of market analysis, qualitative statistics, etc.

3 Multilingual Lexicalization of the CContology

As our role SIG-6 was also to undertake the multilingual and cultural demands of customer complaint forms, a methodology for multilingual lexicalization of ontologies had to be developed. This methodology has been applied to lexicalize the CContology into several natural languages in order to support the development of a software platform providing cross-language CC-forms. For complaint platforms, this helps to systematize the translation of all terms in the generated and filled-in CC-forms that do not contain "free" text.

As shall be clear later in this section, we distinguish between a *multilingual ontology* and *multilingual lexicalization of an ontology*. The former refers either: 1) to different monolingual ontologies with an alignment layer to map between them. Such an alignment layer may include different kinds of relationships (e.g. 'equivalence', 'subtype-of', 'part-of', etc.) between concepts across the aligned ontologies. All of these ontologies, in addition to the alignment layer, form a multilingual ontology. A multilingual ontology can also be 2) a one ontology in which the terminology (i.e. concept labels) is a mixture of terms from different languages. For example, some concepts are lexicalized in language L_1, and others are lexicalized in language L_2, or maybe even in both L_1 and L_2. Yet other concepts may not have terms to lexicalize them. See (Kerremans et al, 2003) for a methodology (called "termontography") that supports such a process of multilingual ontology engineering. The processes of modeling, engineering, or using multilingual ontologies are still open (and difficult) research issues. Some related works can be found in (Lauser et al, 2002) (Agnesund, 1997) (Vossen, 1998) and (Bryan, 2001).

---

[10] This is in fact a clear illustrative application of our composition mechanism, especially in the legal domain. From a "legal" viewpoint, our composition operator means that when including a module into another module (that has a higher authority, or also called *legal weight*), all rules and fact-types in the included module will be inherited by (or applied in) the including module.

Multilingual lexicalization of an ontology is our aim in this section. It is an ontology lexicalized in a certain language (we call this the "native language") and a list of one-to-one translations of the ontology terms into other languages. *This list is not seen as part of the ontology itself*; rather, it belongs at the application level or to a group of users.

Our approach to the multilingual lexicalization of ontologies is motivated by the belief (Guarino, 1998) that *an ontology is language-dependent*, and by Avicenna's argument (980-1037 AC) (Qmair, 1991) that "*There is a strong relationship/dependence between concepts and their linguistic terms, change on linguistic aspects may affect the intended meaning... Therefore logicians should consider linguistic aspects 'as they are'. ...*". Indeed, Conceptual equivalence between terms in different languages is very difficult to find at the domain level. Hence, from an engineering viewpoint, multilingual lexicalization (i.e. one-to-one translation) of ontology terms should not be preserved or generalized at the domain level. Instead, such translations can be fairly established at the application level for a certain application (e.g. CC-form) or group of users.

The main goal of providing the multilingual lexicalization of an ontology is to *maximize the usability of this ontology for several cross-language applications*. We believe that this is of ever increasing importance in today's global, networked economy. In the following paragraphs, we describe our approach to the multilingual lexicalization of ontologies using the CContology as an illustrative example.

Our approach requires *an ontology to be built and lexicalized completely in one language*, namely, the ontology's *native language*. In the case of the CContology, English is chosen as the native language that then acts as *the* reference for translating ontology terms into other languages. Given the CCglossary (all the terms in the CContology and their glosses), and given the CC-form as a certain application scenario, the CContology has been lexicalized into 11 European languages[11]. Notice that changing this application scenario may yield different translations. In fig. 12, we provide a sample of these translations, illustrating one-to-one translation between terms in English, Dutch, and French languages. A CC-form can easily switch between different natural languages by substituting the terms and using the corresponding terms in such a translation list (Jarrar et al, 2003b).

It is important to note that the CCglossary has played a critical role during the translation process of the CContology. The CCglossary has been used as the principal reference, by the translators, for understanding the intended meaning of the terms, and thus achieving better quality translations. It is maybe worth mentioning that the translation process has been subcontracted to an a translation company whose personnel have been trained to follow our approach.

DogmaModeler - Multilingual Lexicalization				
Context	English (Native)	Dutch	French	
Customer Complaint	Complainant	Klager	Plaignant	
Customer Complaint	Complaint	Klacht	Réclamation	
Customer Complaint	Complaint Recipient	Ontvanger	Destinataire	
Customer Complaint	Complaint Number	Klachtnummer	Numéro de Réclamation	
Customer Complaint	Legal Person	Rechtspersoon	Personne Morale	

Fig. 12. An example of multilingual lexicalization of the CContology.

While it is a scalable, pragmatic, easy to use, and systemized approach, one-to-one translations are not as simple as they appear – they do sometimes yield imperfect translations. The translator needs to perform further searches in order to acquire more elegant translations. In the following, we present some issues and guidelines for greater convenience and accuracy in the multilingual lexicalization of ontologies:

- *Cultural issues*. There is a great interdependency between the language and culture (social activities, religion, region, weather, interests, etc.) of a people. Thus, within a community of people speaking the same language, we can find different usage of terms, even within the same context and situation. For example, within the "Customer Complaint" and CC-form application scenario, when translating the term "Complaint" into Arabic, there are two possible terms: "Mathaalem" and "Shakaoa". In Palestine, the most commonly used term is "Shakaoa", while in Saudi Arabia, people prefer the term "Mathaalem". Seemingly, the ideal solution for such a problem is to provide a set of rules for the usage of each term, considering *all* cultural issues (see (Chalabi, 1998)). However, this does not yield a scalable approach for our purposes. Thus, we

---

[11] These translations are not provided in this chapter as the distribution of the knowledge is restricted, and its intellectual property is owned by the CCFORM project.

advise that if such cultural variations are important for a certain application scenario, it is better to treat each variation as a distinct language e.g. English-UK, English-USA, Dutch-Belgium, Dutch-Netherlands, Old-Arabic, Modern-Arabic.

- *Word to word translation is not our goal*. Usually, the purpose of building an ontology is to formally represent an agreed conceptualization of a certain domain, and share it among a community of users. Thus, lexicalizing the concepts in an ontology into multiple languages is a way of maximizing the *usability* of this ontology. It does not result in a multilingual lexicon. In lexicons or dictionaries, the purpose is to list only the common *words* (e.g. based on the corpus of a language) with a description and some lexical information. In ontologies, it is normal to find a concept lexicalized by an expression. For example, "Total Amount Paid", "Trying to obtain data improperly", etc. Such concepts cannot, in general, be lexicalized into one word - at least not in English.

To conclude, with the methodology we have presented in this chapter, we aim to maximize the usability of an ontology over several cross-language applications. We believe this methodology would be useful and easily applicable in information systems that comprise forms, database schemes, XML and RDF tags, etc. However, this methodology might not be suitable for ontology-based information retrieval and natural language processing applications. For such application scenarios, multilingual ontologies might be more suitable. See (Gilarranz et al, 1997) (Bonino et al, 2004).

4   Conclusions and Future Directions

We have presented an ontology for customer complaint management, with the aim of improving the effectiveness and transparency in e-business transactions. Using ontologies as a foundation for cross-border online complaint management platforms can indeed improve the effectiveness, scope and extensibility of such platforms. While offering individual companies, organizations or associations the possibility of advanced customization (by including ontology extension capabilities) semantic consistency is maintained through the complaint management terminology. Furthermore, by restricting extensions to certain parts of the ontology, some legal constraints such as privacy regulations may be enforced systematically.

The proposed methodology for the multilingual lexicalization of ontologies is a pragmatic one. It offers a scalable way of offering multilingual services –a necessity for cross-border complaint management within the EU. An important goal in our future research is to develop a formal approach for developing multilingual ontologies which would for example, allow computers to interpret and disambiguate terms in different languages.

Remark: The ontology presented in this chapter has been implemented and applied in the CC-form demo portal[12]. Furthermore, the results of this research are being disseminated by FEDMA (the Federation of European Direct and Interactive Marketing) into its wide consortium. However, we are not aware of any realization of this portal *yet*.

References

ABA Task Force on Electronic Commerce and Alternative Dispute Resolution. Final Report. (2002)

Agnesund, M. (1997). Representing culture-specific knowledge in a multilingual ontology. Proceedings of the IJCAI-97 Workshop on Ontologies and Multilingual NLP.

Bonino, D., Corno, F., Farinetti, L., Ferrato, A. (2004). Multilingual Semantic Elaboration in the DOSE platform. ACM Symposium on Applied Computing, SAC'04. Nicosia, Cyprus. March.

Bryan, M. (eds.) (2001). MULECO -Multilingual Upper-Level Electronic Commerce Ontology. MULECO draft CWA. The CEN/ISSS Electronic Commerce Workshop.

---

[12] This portal (which was not an official deliverable in the project) is no longer available due to copy-right issues.

Chalabi, C. (1998). Sakhr Arabic-English Computer-Aided Translation System. AMTA'98. pp. 518–52.

Cho, Y., Im, I., Hiltz, S., Fjermestad, J. (2002). An Analysis of Online Customer Complaints: Implications for Web Complaint Management. Proceedings of the 35th Annual Hawaii Int. Conf. on System Sciences. Vol. 7.

Claes, F., Wernerfelt, B. (1987). Defensive Marketing Strategy by Customer Complaint Management: A Theoretical Analysis. Journal of Marketing Research, No. 24. November pp. 337–346

Gilarranz, J., Gonzalo, J., Verdejo, F. (1997). Language-independent text retrieval with the EuroWordNet multilingual semantic database. The 2nd WS on Multilinguality in the Software Industry.

Guarino, N. (1998). Formal Ontology in Information Systems. Proceedings of FOIS'98, IOS Press. pp. 3–15

Halpin, T. (1989). A logical analysis of information systems: static aspects of the data-oriented perspective. PhD thesis, University of Queensland, Brisbane. Australia.

Halpin, T. (2001). Information Modeling and Relational Databases. 3rd edn. Morgan-Kaufmann.

Jarrar, M., & Meersman, R. (2002). Formal Ontology Engineering in the DOGMA Approach. In proceedings of the International Conference on Ontologies, Databases, and Applications of Semantics (ODBase 2002). Volume 2519, LNCS, Pages: 1238-1254, Springer. ISBN: 3540001069.

Jarrar, M. (2005). Towards Methodological Principles for Ontology Engineering. PhD thesis, Vrije Universiteit Brussel.

Jarrar, M. (2005b). Modularization and Automatic Composition of Object-Role Modeling (ORM) Schemes. In OTM 2005 Workshops, proceedings of the International Workshop on Object-Role Modeling (ORM'05). Volume 3762, LNCS, Pages (613-625), Springer. ISBN: 3540297391.

Jarrar, M. (2006): Towards the Notion of Gloss, and the Adoption of Linguistic Resources in Formal Ontology Engineering. Proceedings of the 15th International World Wide Web Conference (WWW2006). Edinburgh, Scotland. Pages 497-503. ACM Press. ISBN: 1595933239.

(Jarrar, 2007) Jarrar, M. (2007). Towards Automated Reasoning on ORM Schemes. -Mapping ORM into the DLR_idf description logic. Proceedings of the 26th International Conference on Conceptual Modeling (ER 2007). Volume 4801, LNCS, Pages (181-197), Springer. ISBN:9783540755623. New Zealand.

Jarrar, M. (2007). Mapping ORM into the SHOIN/OWL Description Logic- Towards a Methodological and Expressive Graphical Notation for Ontology Engineering. OTM workshops (ORM'07). Portogal. Volume 4805, LNCS, Pages (729-741), Springer. ISBN: 9783540768890.

Jarrar, M., Verlinden, R., & Meersman, R. (2003). Ontology-based Customer Complaint Management. OTM 2003 Workshops, proceedings of the 1st International Workshop on Regulatory Ontologies and the Modeling of Complaint Regulations. Italy. Volume 2889, LNCS, pages: 594-606, Springer. ISBN: 3540204946.

Jarrar, M., Lisovoy, A., Verlinden, R., & Meersman, R. (2003b): OntoForm Ontology based CCForms Demo. Deliverable D6.8, The CCFORM Thematic Network (IST-2001-34908), Brussels.

Jarrar, M., Demey, J., & Meersman, R. (2003c). On Using Conceptual Data Modeling for Ontology Engineering. Journal on Data Semantics, Special issue on ""Best papers from the ER/ODBASE/COOPIS 2002 Conferences"", 2800(1):185-207. Springer, ISBN: 3540204075.

Jarrar, M., & Heymans, S. (2006). Unsatisfiability Reasoning in ORM Conceptual Schemes. Proceeding of International Conference on Semantics of a Networked World. Germany. Volume 4254, LNCS, Pages (517-534), Springer. ISBN: 3540467882.

Jarrar, M., Keet, M., & Dongilli, P. (2006b). Multilingual verbalization of ORM conceptual models and axiomatized ontologies. Technical report. STARLab, Vrije Universiteit Brussel.

Jarrar, M., & Eldammagh, M. (2007, August). Reasoning on ORM using Racer. Technical Report. STAR Lab, Vrije Universiteit Brussel, Belgium.

Jarrar, M., & Heymans, S. (2008, in press). Towards Pattern-based Reasoning for Friendly Ontology Debugging. Journal of Artificial Tools. World Scientific Publishing.

Jarrar, M. & Meersman, R. (2008b, in press). Ontology Engineering -The DOGMA Approach. (Chapter 3). Advances in Web Semantic. Volume 1, IFIP2.12. Springer.

Kerremans, K., Temmerman, R. and Tummers, J. (2003). Representing multilingual and culture-specific knowledge in a VAT regulatory ontology: support from the termontography approach. OTM 2003 Workshops.

Lauser, B., Wildemann, T., Poulos, A., Fisseha, F., Keizer, J., Katz, S. (2002). A Comprehensive Framework for Building Multilingual Domain Ontologies. Proceedings of the Dublin Core and Metadata.

Meersman R. (1999). Semantic Ontology Tools in Information System Design. Proceedings of the ISMIS 99 Conference, LNCS 1609, Springer Verlag. pp. 30–45

Qmair, Y. (1991). Foundations of Arabic Philosophy. Dar al-Shoroq. Bairut, ISBN 2-7214-8024-3.

Vossen, P. (eds.) (1998). EuroWordNet: A Multilingual Database with Lexical Semantic Networks. Kluwer Academic Publishers, Dordrecht.

Vassileva, B., Scoggins, P. (2003). Consumer Complaint Forms: An Assessment, Evaluation and Recommendations for Complaint Categorization. Technical report, CCForm Project (IST-2001-34908). Brussels.