Workshop Notes

The 4th International Workshop on

Acquisition, Representation and Reasoning with Contextualized Knowledge

ARCOE-12

August 28, 2012
Montpellier, France

held at the

European Conference on Artificial Intelligence

Editors

Michael Fink
Martin Homola
Alessandra Mileo
Ivan Varzinczak

http://www.arcoe.org

Preface

Dealing with context is one of the most interesting and important problems faced in Artificial Intelligence (AI). Traditional AI applications often require to model, store, retrieve and reason about knowledge that holds within certain circumstances—the context. Without considering this contextual information, reasoning can easily run into problems such as: inconsistency, when considering knowledge in the wrong context; inefficiency, by considering knowledge irrelevant for a certain context; incompleteness, since an inference may depend on knowledge assumed to hold for a context but which is not explicitly stated. Contextual information is also relevant in knowledge representation and reasoning and it represents a strategic aspect to deal with inconsistency, ambiguity, uncertainty, knowledge base evolution, and commonsense reasoning, among others.

In recent years, research in context-aware knowledge representation and reasoning became more relevant in the areas of Semantic Web and Intelligent Systems, where knowledge is not considered a monolithic and static asset, but it is distributed in a network of interconnected heterogeneous and evolving knowledge resources. The ARCOE workshop aims to provide a dedicated forum for researchers interested in these topics to discuss recent developments, important open issues, and future directions.

Submissions to ARCOE-12 have been reviewed by at least two and in most cases three PC members and ranked on relevance and quality. Fifty percent of the submissions have been selected for presentation at the workshop and for inclusion in these Workshop Notes.

Thanks to the invaluable and much appreciated contributions of the authors and the Programme Committee, ARCOE-12 provides participants with an opportunity to position various approaches with respect to one another. Hopefully, though the workshop and these Notes will also start a process of cross-pollination and consolidate the constitution of a truly interdisciplinary research-community dedicated to acquisition, representation and reasoning with contextualized knowledge.

(Vienna, Bratislava, Galway, Pretoria – August 2012)

Michael Fink, Martin Homola, Alessandra Mileo, Ivan Varzinczak

Table of Contents

Workshop Organization

ARCOE-12 Co-Chairs

- Michael Fink (Vienna University of Technology, Austria)
- Martin Homola (Comenius University, Slovakia)
- Alessandra Mileo (Digital Enterprise Research Institute, NUI Galway, Ireland)
- Ivan Varzinczak (CSIR Meraka Institute and University of KwaZulu-Natal, South Africa)

ARCOE-12 Steering Committee

- Alan Bundy (University of Edinburgh, UK)
- Thomas Eiter (Vienna University of Technology, Austria)
- Luciano Serafini (Fondazione Bruno Kessler, Italy)

ARCOE-12 Invited Speakers

- Alan Bundy (University of Edinburgh, UK)
- Luciano Serafini (Fondazione Bruno Kessler, Italy)

ARCOE-12 Program Committee

- Grigoris Antoniou (FORTH, Greece)
- Christoph Benzmüller (Free University of Berlin, Germany)
- Leopoldo Bertossi (Carleton University, Canada)
- Katarina Britz (CSIR Meraka Institute, South Africa)
- Loris Bozzato (Fondazione Bruno Kessler, Italy)
- Patrick Blackburn (University of Roskilde, Denmark)
- Gerhard Brewka (University of Leipzig, Germany)
- Alan Bundy (University of Edinburgh, UK)
- Henning Christiansen (University of Roskilde, Denmark)
- Thomas Eiter (Vienna University of Technology, Austria)

- Michael Fink (Vienna University of Technology, Austria)
- Chiara Ghidini (Fondazione Bruno Kessler, Italy)
- Víctor Gutiérrez Basulto (University of Bremen, Germany)
- Siegfried Handschuh (Digital Enterprise Research Institute, Ireland)
- Martin Homola (Comenius University, Slovakia)
- Szymon Klarman (Free University of Amsterdam, Netherlands)
- Oliver Kutz (University of Bremen, Germany)
- João Leite (New University of Lisbon, Portugal)
- Vincenzo Malteze (University of Trento, Italy)
- Stefan Mandl (University of Augsburg, Germany)
- Tommie Meyer (CSIR Meraka Institute, South Africa)
- Alessandra Mileo (Digital Enterprise Research Institute, NUI Galway, Ireland)
- Valeria de Paiva (University of Birmingham, UK)
- Rafael Peñaloza (TU Dresden, Germany)
- Guilin Qi (Southeast University, China)
- Torsten Schaub (University of Potsdam, Germany)
- Luciano Serafini (Fondazione Bruno Kessler, Italy)
- Ivan Varzinczak (CSIR Meraka Institute and University of KwaZulu-Natal, South Africa)
- Renata Wassermann (University of São Paulo, Brazil)
- Antoine Zimmermann (École Nationale Supérieure des Mines de Saint-Étienne, France)

The Role of Context in Controlling Inconsistency

Alan Bundy

University of Edinburgh, UK
bundy@inf.ed.ac.uk

In 2011 the Opera Collaboration at Gran Sasso Laboratory reported measurements of the velocity of neutrinos in a CERN experiment as being faster than the speed of light. This announcement created a storm of media attention because faster than light travel is in violation of Einstein's theory of special relativity. Opera thus invited us to consider two mutually contradictory assertions: faster than light travel is/isn't possible, together with some evidence for each of them. Such paradoxical situations are commonplace in science, but also frequently occur in everyday experience. In the Galileo project we have been automating mechanisms for proposing resolutions of such contradictions, mainly by changes in the representation language. Contexts have played a key role in Galileo as a mechanism for quarantining these contradictory assertions while they are diagnosed and repaired. Contexts also allow us to prove that the repairs meet an appropriately modified definition of conservative extension, thus providing an assurance that the proposed repairs are minimal.

Multi context logics: a formal framework for structuring knowledge

Luciano Serafini

Fondazione Bruno Kessler
via Sommarive 18, I-38123, Trento, Italy
serafini@fbk.eu

In many knowledge representation tasks one is put in front of the problem of integrating the knowledge encoded into a set of logical theories each of which describes a portion of the world. Let us suppose for simplicity that each theory is written in the same logic, but possibly using different signatures, with different intuitive meaning, and suppose also that each theory describes some (and not all) of the aspects of the world portion. The problem is how can we build an integrated theory which represents all the knowledge contained in the single theories? Multi context logics (MCL) is a formalism that allow to integrate multiple logical theories (contexts) in a more complex structure called multi context system. In the past 20 years MCLs have been developed for contexts in propositional logics, first order logics, description logics and temporal logic. The two principles MCL are *locality* and *compatibility*. The principle of locality states that a context axiomatizes in a logical theory a portion of the world, and that every statement entailed by such a theory is intended to hold within such a portion of the world. The principle of compatibility instead states that, since different contexts can describe overlapping portions of world, the theories they contain must be constrained so that they describe compatible situations. Following these two principles, the formal semantics of an MCL is the result of a suitable composition of the semantics associated to each single context. This takes the name of *Local models semantics*. The effects of the two principles above on the inference engines that can be defined on a multi contextual knowledge base are the following: Locality principle implies that inference rules applied to knowledge inside a context (aka, local inference rules) allow to infer local truths; Compatibility principle instead implies that certain facts in a context can be inferred on the base of other facts present on other compatible contexts. This information propagation is formalized via a special type of inter contextual inference rules called *bridge rules*

In general terms, a multi context logic is defined on a family of logical languages $\{L_i\}_{i \in I}$ where each L_i is used to specify what holds in the i-th context. The set I of context indexes (aka context names) can be either a simple set, or a set equipped with some algebraic structure, like total or partial order, and operations on context indices. The relations and functions defined on I can be used to specify the organization of contexts in terms of an algebraic structure. For instance a partial order $\prec$ no I, can be used to represent that a context is wider (more general) than another context, e.g., $\mathsf{football} \prec \mathsf{sport}$, means that the context of sport is more specific that the context of football. To represent what is true in a scene in different time stamps, we can enrich I with a total order $\prec$ so that CV_Luciano_2010 $\prec$ CV_Luciano_2011 represents the fact that the context describing Luciano's curriculum vitæat 2010 precedes the context describing his CV at 2011.

A *model* for a multi context logic $\{L_i\}_{i\in I}$ is a pair $\langle M_I, C\rangle$ composed of a family of local models $M_I = \{M_i\}_{i\in I}$, where each M_i is a model of L_i, and a *compatibility relation* C among the local models. The formal structure of C can vary depending on the type of local models and the type of constraints it is necessary to impose on the local models of different contexts. For this reason we don't give a general definition of C, which will be completely defined for each specific multi context logic. Satisfiability of formulas in L_i is defined w.r.t1. the local models. Namely If ϕ is a formula of the language L_i, then the a multi context model satisfies $i : \phi$ iff $M_i \models_i \phi$, where $\models_i$ is the satisfiability relation associated to the local logic L_i.

A *Multi context theory* in a multi context logic $L_I = \{L_i\}_{i\in I}$ is a family of theories $\{T_i\}_{i\in I}$, where each T_i is a set of statements in the logics L_i, and a set BR of *bridge rules*. Intuitively each T_i axiomatizes the constraints on the local models M_i, while the bridge rules BR axiomatizes the compatibility relations. Bridge rules are cross logical axioms and their syntactic form depends on the local logics, so as in the case of the compatibility relation their syntactic depends on the syntax of each L_i.

In my invited talk, I will go through the many possible examples of multi context logics, starting from the simplest one, the propositional multi context logics, going through hierarchical meta logics, multi context logics for beliefs and propositional attitudes, non-monotonic propositional context logics, distributed first order logics, distributed description logics, and logics for semantic import and contextualized knowledge repository for the semantic web.

Context on the Semantic Web: Why and How

Loris Bozzato[1], Martin Homola[2], and Luciano Serafini[1]

[1] Fondazione Bruno Kessler, Via Sommarive 18, 38123 Trento, Italy
[2] FMFI, Comenius University, Mlynská dolina, 84248 Bratislava, Slovakia

{bozzato,serafini}@fbk.eu, homola@fmph.uniba.sk

Abstract. It is becoming increasingly apparent that knowledge published via the Semantic Web (SW) and Linked Open Data (LOD) resources is typically valid w.r.t. some assumed context. The contextual information, however, is often left implicit and not explicitly indicated. What is more, the means offered by SW technologies to represent this type of knowledge and link it to the resource itself are rather limited. In this position paper we argue that more advanced means of treatment of context in the SW and LOD resources are needed. Contextual meta knowledge has to be explicitly represented and logically treated. We propose a set of properties that we think such a representation should have and finally we review the known existing approaches to contextual representation on the SW.

1 Introduction

An increasing number of ontologies and data sets are being published using the SW languages such as RDF and OWL. Especially under the more recent LOD initiative, large knowledge sources such as DBpedia and Freebase but also many others were conceived and populated. However, rarely these large portions of knowledge are absolutely valid. They are typically qualified with respect to some *context*, i.e., they are assumed to hold under certain circumstances – relative to certain time period, a geo-political or geo-cultural region, certain specific topics, etc. By context we will therefore mean the situation that limits the validity of information and by contextual information (or meta information) we will mean any kind of description of this situation.

On the other hand, there is a lack of a widely accepted standard mechanism to qualify knowledge with the context in which it is supposed to hold. Sometimes the contextual meta data is mixed directly with other data; more often this meta information is crafted in annotation properties like rdfs:comment or owl:AnnotationProperty which do not affect reasoning at all, or it is even left implicit in many cases.

Recognizing this problem, a number of extensions to SW languages have been proposed with the aim to handle context. Among other proposals [4, 12, 5, 1, 15, 8], there are approaches such as aRDF [18], Metaview [17], two-dimensional description logics of context [10, 9] and CKR [14, 6]. All these approaches offer some solution for this problem, however, a recognized and widely accepted consensus has not yet been reached by far.

In this position paper, we argue that contextual representation of knowledge is one of the most needed ingredient that is still missing in the standardized SW technology family. We underline the relevance of this topic and propose to the community to try to come up with a set of standard and universally acknowledged features of such a representation for the SW. In order to foster the discussion about such features, we then present a number of properties that we believe a contextualized representation suitable for the SW should satisfy.

It is to be stressed that these properties mirror our ideas and intuitions. We leave it open for the community to discuss and evaluate the proposed properties, and to come up with modifications or with other proposals. Towards the end of the paper we give a brief overview of the existing approaches and conclude with discussion on future directions.

2 Properties of Contextual Representation for the Semantic Web

In the search for a suitable knowledge representation framework it is a relevant question what properties it should have in order to serve best its purpose, in this case the purpose of enabling context to be explicitly represented and reasoned about within the SW.

According to our opinion the following properties should be considered. It is not, however, our stance that this is the canonical set of properties to be applied in context-enabled SW knowledge representation from this moment. This is something the community needs to discuss and anticipate. The following list should therefore not be considered as final, but rather as a starting point in this discussion.

Property 1 (Encapsulation). Knowledge that shares the same context should be encapsulated, easily identified and accessed.

A good example for encapsulation can be drawn from the *context as a box* metaphor known from the previous works on context [3]. Under this metaphor a context is perceived as a "box" whose boundaries are given by a set of contextual attributes, i.e., all knowledge that fits into common boundaries should be grouped within the same box. In accordance with this paradigm, these "boxes", that is, collections of knowledge that shares the same contextual qualification, are often themselves referred to as *contexts*, as we also do in this paper. Encapsulation does not automatically imply that knowledge bases need to be physically split into subsets for each context. On the other hand, the part of knowledge bound to the same context has to be easily retrieved and manipulated as needed. In this sense, also, the context should serve as navigation axis when working with the knowledge base.

Property 2 (Explicit meta knowledge). Knowledge about contexts should be explicitly represented in a logical language.

The knowledge inside the context, that is, the knowledge represented in the first place, will be called *object* knowledge. We assume that this knowledge has a logical representation, as it normally is in the SW. The property implies that in addition the contextual information that qualifies the object knowledge should also be explicitly represented in a logical language. This information will be called *meta* knowledge.

There are multiple ways how to represent meta knowledge. In the context as a box approach [3] and also other works [11] it is formalized using a set of contextual dimensions together with their possible values.

For example, given some knowledge about the current US presidential election, the meta data that we want to possibly represent may be that this information is relative to the location USA, that it is relative to a certain time period, e.g., year 2012, and to a specific topic, e.g., "presidential election". This information should be explicitly stated in the representation language, for instance, in OWL[3] using properties $\mathsf{location}$, time and topic and individuals such as USA, $\mathsf{2012}$ and $\mathsf{presidential\_election}$. Assuming that the individual $\mathsf{c42}$ acts as the context identifier for our context, we may assert axioms such as $\mathsf{location}(\mathsf{c42}, \mathsf{USA})$, $\mathsf{time}(\mathsf{c42}, \mathsf{2012})$, and $\mathsf{topic}(\mathsf{c42}, \mathsf{presidential\_election})$. There might be more knowledge that we want to formalize in relation with the meta data. For instance we may want to assert that USA is a country, and that every country has a capital. To do this we may add more axioms, e.g., $\mathsf{Country}(\mathsf{USA})$ and $\mathsf{Country} \sqsubseteq \exists \mathsf{hasCapital}.\mathsf{City}$. The important point in this example is not this particular formalization, but instead the fact that the meta knowledge is represented in some logical language on top of which reasoning can be done.

Property 3 (Separation of meta knowledge and object knowledge). Meta knowledge is clearly distinguished from object knowledge.

By this property we mean that one can immediately tell which statements belong to object knowledge and which belong to meta knowledge, i.e., it should be apparent that the properties $\mathsf{location}$, time and topic as used above belong to meta knowledge, and that the statements formed using these properties represent meta information. Again, the meta knowledge does not have to be physically separated from the object knowledge, but at least the two vocabularies used for each one respectively should be disjoint. One good reason to stick to this rule is simply to avoid the user to confuse object and meta statements, since each type of statements has different purpose and influences the knowledge base in different ways.

Property 4 (Relations between contexts). If one context is related to another, there should be a way how to represent this within the framework.

Studying contexts in separation makes little sense, as if there was just one context, representation of meta data and contextual reasoning would not be needed in fact. Hence contextual representation must be able to deal with multiple contexts and with the implications that the knowledge present in a context has on the other contexts. Therefore the literature of contextual reasoning has dedicated significant attention to possible relations between contexts [3, 2, 11]. A distinctive attention was given to the context coverage relation, that enables to organize the contexts w.r.t. the specificity–generality axis [11, 2]. Other possible relations are for instance that of neighboring context (in space) or consecutive contexts (in time). Clearly such relations have significant influence of how much and which part of knowledge should be carried over from one context to another during reasoning. Therefore they need to be considered also in the case of contextual knowledge representation for the SW.

For example, given the context of US presidential election 2012, and given another context, say, one of US politics of early 21st century, the second context is clearly

[3] For convenience we use description logic-like syntax for OWL expressions.

broader then the first one – it contains the first context in some sense. This also means that the first context is narrower then the second one. Similarly, considering a third context, this time of US presidential administration 2013–2016, we can see not only that this context is also narrower then the second context, but also that it immediately succeeds the first context in time. We discuss later on how such relations should be taken into account in reasoning (see Property 7), which, however, may only be possible if the relations between contexts are explicitly represented.

Property 5 (Contextual reasoning). Reasoning should take into the account as well the contextual meta knowledge and relations between contexts.

This property states that both the contextual meta knowledge and the context structure should be taken into the account in reasoning. Continuing our example, we should be able to assert a constraint requiring that if the same individual is an instance of $\mathsf{President}$ in two consecutive US administration contexts, then it cannot be an instance of $\mathsf{Candidate}$ in the following presidential election context. Clearly, the reasoner needs to be able to access and process the meta knowledge and it must further consider also the relations between contexts in order to reason with such a constraint.

In our previous work [14] we have argued that while meta knowledge may (and in fact should) influence the object knowledge, it should not be the case vice versa. This was relevant in order to maintain low complexity of reasoning. This desideratum is a bit too strict, however, if satisfiable complexity can be maintained (see Property 9) of course also more complex meta-reasoning patterns can be exploited.

One example of such a complex modeling pattern is the case in which we would organize the context structure differently depending on the assertions in the object knowledge. Theoretically such use cases are appealing, however, their practical use and especially the implications on the computational costs have to be thoroughly investigated.

Property 6 (Locality of knowledge). In each context we should be able to state axioms with local effect, that do not affect other contexts, and are not affected by other contexts.

For instance, in any context related to US presidential administration there will be a concept $\mathsf{President}$ but in each of these contexts this concept may have different instances. Furthermore, in all of these context we may want to add an axiom which would guarantee that the concept has only one instance (i.e., there is always only one president). On the other hand, in the broader context related to early 21st century US politics, there is also a concept $\mathsf{President}$, in this case however we do not want to have such an axiom – there may be multiple presidents in this broader period.

Property 7 (Knowledge lifting). If needed, specific knowledge can be lifted from one context and reused by another.

Locality of knowledge, on the other hand, should not imply opacity. If needed, knowledge from one context should be accessible to another context, especially if the contexts are related in a favourable fashion. Such propagation of knowledge between contexts is commonly referred to as *knowledge lifting*, and it has been intensively studied in the literature [13, 11, 2]. It is usually implemented by specific lifting axioms, or possibly by some automated lifting mechanism.

In our running example we might want to assure that the president in the administration of 2013–2016 is featured as an instance of the concept $\mathsf{President}$ in the context of early 21st century US politics. This is an example of knowledge lifting and it could be asserted using a specific lifting axiom that would lift $\mathsf{President}$ from the administration 2013–2016 into the latter context as $\mathsf{President2013\text{-}2016}$ and then asserting a local axiom $\mathsf{President2013\text{-}2016} \sqsubseteq \mathsf{President}$.

Consider also the case of two consecutive contexts, e.g., the one of US presidential election 2012 and the one of the US presidential administration of 2013–2016. With specific lifting axioms, we should be able to assert that the individual which represents the elected president in the first context is the same as the one which represents the actual president in the latter context. The amount of knowledge reuse may possibly be constrained by relations between contexts – for example, if the contexts are unrelated the lifting may not be possible, or may at least be very limited.

Note that Properties 6 and 7 are not mutually conflicting. Both cases should be possible at the same time: one approach is to have knowledge with local meaning, if no lifting axioms were specified. Knowledge lifting can then be implemented by giving the user the option to specify lifting axioms in specific cases as needed.

Property 8 (Overlapping and varying domains). Objects can be present in multiple contexts, but not necessarily in all contexts.

The RDF semantics relies on the fact that the same URI should always have the same meaning in every RDF document. Accordingly, we believe constants should have the same meaning, at least in related contexts, and therefore the interpretation domains should partly overlap. We refrain from this requirement when contexts are completely unrelated. Also, we need not require to the existence of the interpretation for all constants of the language within a given context, as it is usual in DL-like OWL semantics. Therefore constants should naturally appear (and be interpreted) in those contexts for which they are relevant.

In this respect, in our example the constant obama which is present in the context related to US presidential administration of 2009–2012 need not necessarily appear in the context of the next administration. On the other hand obama should certainly appear in the early 21st century US politics context, as given the relation of the latter context to the one of the 2009–2012 administration, it is clearly a relevant constant there.

Property 9 (Complexity invariance). The contextual layer should not increase the complexity of reasoning.

The last property is concerned with the computational cost that we have to pay to add the contextual dimension to the SW. Given the fact that some of the SW languages already exhibit quite high complexity (i.e., OWL 2, based on the $\mathcal{SROIQ}$ DL, is $\mathrm{2NExpTime}$-complete [7]), we believe that the contextual layer needs to be added without any increase in the complexity – if only possible. Indeed, a minor increase in complexity, especially with more tractable OWL fragments, may not be as harmful. Therefore we could more generally require that the complexity of the resulting contextualized formalism should be acceptable.

3 Existing Approaches and Proposals

Perceiving the need for some mean of representing context in the SW, both aRDF [18] and Context Description Framework [8] extend RDF triples with an n-tuple of qualification attributes with partially ordered domains. In such an approach, each formula is annotated separately, seemingly violating the encapsulation property. However, this can also be seen as an optimization issue easy to resolve during the implementation.

Straccia et al. [16] enable RDFS graphs to be annotated with values from a lattice. The semantics of the framework is based on an interpretation structure that is common in multi-valued logics. This effectively restricts the dimensional structure to a complete lattice: for every two contexts there exists a meet ($\wedge$) and a join ($\vee$) and also the global bottom ($\perp$) and top ($\top$). This permits to study relations between contexts, e.g., looking for least common super context and greatest common sub context in some sense.

Another extension of RDFS to cope with context was proposed by Guha et al. [5] and further developed in Bao et al. [1]. A new predicate $\mathsf{isin}(c, \phi)$ is used to assert that the triple ϕ occurs in the context c. A set of operators to combine contexts ($c_1 \wedge c_1$, $c_1 \vee c_2$, $\neg c$) and to relate contexts ($c \Rightarrow c_2$, $c \rightarrow c_2$) is defined, making the approach particularly suited for manipulating contexts. Unfortunately, to our best knowledge no decision procedure is known so far.

The Metaview approach [17] enriches OWL ontologies with logically treated annotations and it can be used to model contextual meta data, albeit on per-axiom basis. In the Metaview framework, however, the contextual level has no direct implications on ontology reasoning, but it makes possible to reason about the ontology or even data. Also a contextually sensitive query language MQL is provided. Presented examples concentrate especially on modeling provenance of data and associated confidence, and the framework seems well suited for this purpose.

Two-dimensional description logics of context [10] allow for a multi-modal extension of one description logic by another. In this fashion, a combination of some object language $\mathcal{L}_O$ and another context language $\mathcal{L}_C$ results into the language $\mathfrak{C}_{\mathcal{L}_O}^{\mathcal{L}_C}$. This permits a structured knowledge base composed of a number of contexts in $\mathcal{L}_O$ with contextual meta data expressed in $\mathcal{L}_C$. For instance $\mathfrak{C}_{\mathcal{ALC}}^{\mathcal{ALC}}$ (previously called $\mathcal{ALC}_{\mathcal{ALC}}$) was studied [9]. An interesting feature of this logic are the contextual modal operators such as $[C]_r A$ representing "all objects of type A in all contexts of type C reachable from the current context via relation r". The language also provides rich representation possibilities for meta knowledge using a separate knowledge base in $\mathcal{L}_C$.

Finally, Contextualized Knowledge Repository (CKR) [14, 6] allows for contextualized knowledge bases structured into contexts. Local language of the contexts can be as expressive as $\mathcal{SROIQ}$ (i.e., OWL 2) or any of its sublanguages. Contextual meta data is assigned to contexts in form of dimensional attributes, which are formalized in a separate meta knowledge base. A set of attributes assigned to some context is called a dimensional vector. To access information across contexts so called qualified symbols are used: the concept $A_\mathbf{c}$ represents the meaning of concept A in the context identified by the dimensional vector $\mathbf{c}$. The choice to implement knowledge lifting with qualified symbols is one of the main differences between CKR and the previous approach.

4 Discussion

Coming back to the existing approaches that we discussed above, we first have to say that CKR was indeed modeled based on similar desiderata [14] as proposed in this paper, being proposed by the same authors. Remarkably, also the two-dimensional description logics of context seem to be in accord with most of the properties. Both these frameworks provide good encapsulation of contexts, rich representation of meta knowledge, and contextual reasoning. We stop the comparisons here; a more complex evaluation of all of the formalisms is beyond the scope of this position paper.

Considering the increasing importance of contextual knowledge representation for the SW and LOD resources, we have proposed in this paper a set of properties that we believe such a representation should satisfy. Our intention here is not to propose the canonical set of properties that should be immediately adopted: instead, we intend this just as the first step in a longer process, in which the community should review these properties, try to reach consensus in discussion, and work towards a standardization. Only then a well suited solution can be found, which can be adopted by a number of SW and LOD data sources to the benefit of the users.

Acknowledgements. This research was supported from the LiveMemories project. Martin Homola is also supported from the Slovak national project VEGA no. 1/1333/12.

References

1. Bao, J., Tao, J., McGuinness, D.: Context representation for the semantic web. In: Procs. of WebSci10 (2010)
2. Benerecetti, M., Bouquet, P., Ghidini, C.: Contextual Reasoning Distilled. Experimental and Theoretical AI 12(3), 279–305 (2000)
3. Benerecetti, M., Bouquet, P., Ghidini, C.: On the Dimensions of Context Dependence. In: Perspectives on Contexts. CSLI (2007)
4. Ding, L., Finin, T., Peng, Y., da Silva, P.P., McGuinness, D.: Tracking RDF graph provenance using RDF molecules. In: ISWC (2005)
5. Guha, R., McCool, R., Fikes, R.: Contexts for the semantic web. In: ISWC (2004)
6. Homola, M., Serafini, L.: Contextual representation and reasoning with description logics. In: DL (2011)
7. Kazakov, Y.: $\mathcal{RIQ}$ and $\mathcal{SROIQ}$ are harder than $\mathcal{SHOIQ}$. In: KR (2008)
8. Khriyenko, O., Terziyan, V.: A framework for context sensitive metadata description. IJSMO 1(2), 154–164 (2006)
9. Klarman, S., Gutiérrez-Basulto, V.: $\mathcal{ALC}_{\mathcal{ALC}}$: a context description logic. In: JELIA (2010)
10. Klarman, S., Gutiérrez-Basulto, V.: Two-dimensional description logics of context. In: DL (2011)
11. Lenat, D.: The Dimensions of Context Space. Tech. rep., CYCorp (1998)
12. Liao, H.C., Tu, C.C.: A RDF and OWL-based temporal context reasoning model for smart home. Inform. Tech. J. 6, 1130–1138 (2007)
13. McCarthy, J.: Notes on formalizing context. In: IJCAI (1993)
14. Serafini, L., Homola, M.: Contextualized knowledge repositories for the semantic web. Journal of Web Semantics, Special Issue: Reasoning with context in the Semantic Web 12 (2012)
15. Stoermer, H.: Introducing context into semantic web knowledge bases. In: CAiSE DC (2006)

16. Straccia, U., Lopes, N., Lukácsy, G., Polleres, A.: A general framework for representing and reasoning with annotated semantic web data. In: AAAI (2010)
17. Tran, T., Haase, P., Motik, B., Cuenca Grau, B., Horrocks, I.: Metalevel information in ontology-based applications. In: AAAI (2008)
18. Udrea, O., Recupero, D., Subrahmanian, V.S.: Annotated RDF. ACM Trans. Comput. Log. 11(2), 1–41 (2010)

Introducing Conviviality as a property of Multi-Context Systems*

Antonis Bikakis[1] and Vasileios Efthymiou[2] and Patrice Caire[2] and Yves Le Traon[2]

[1] Department of Information Studies, University College London
a.bikakis@ucl.ac.uk
[2] University of Luxembourg, Interdisciplinary Center for Security, Reliability and Trust (SnT)
firstname.lastname@uni.lu

Abstract. Multi-Context Systems (MCS) are rule-based representation models for distributed, heterogeneous knowledge sources, called *contexts*, such as ambient intelligence devices and agents. Contexts interact with each other through the sharing of their local knowledge, or parts thereof, using so-called *bridge rules* to enable the cooperation among different contexts. On the other hand, the concept of *conviviality*, introduced as a social science concept for multiagent systems to highlight soft qualitative requirements like user friendliness of systems, was recently proposed to model and measure cooperation among agents in multiagent systems. In this paper, we introduce conviviality as a property to model and evaluate cooperation in MCS. We first introduce a formal model, then we propose conviviality measures, and finally we suggest an application consisting in a conviviality-driven method for inconsistency resolution.

1 Introduction

Multi-Context Systems (*MCS*) [1–3] are logical formalizations of distributed context theories connected through a set of bridge rules, which enable information flow between contexts. A *context* can be thought of as a logical theory - a set of axioms and inference rules - that models local knowledge. Intuitively, MCS can be used as a representation model for any information system that involves distributed, heterogeneous knowledge agents including peer-to-peer systems, distributed ontologies (e.g. Linked Open Data) or Ambient Intelligence systems. In fact, several applications have already been developed on top of MCS or other similar formal models of context including (*a*) the CYC common sense knowledge base [4], (*b*) contextualized ontology languages, such as Distributed Description Logics [5] and C-OWL [6], (*c*) context-based agent architectures [7, 8], and (*d*) distributed reasoning algorithms for Mobile Social Networks [9] and Ambient Intelligence systems [10].

While designing a real system based on the MCS model, comparing MCSs in order to select the most appropriate configuration, evaluations and measures are needed. For example, consider a house environment, which consists of various devices, sensors and appliances connected through a wireless network. The role of an Ambient Intelligence system is to transform this network of devices into a smart home environment by enabling devices to share and reason with their context knowledge. However, techniques

* Thanks to: National Research Fund, Luxembourg (I2R-SER-PFN-11COPA) CoPAInS project

to allow sharing of knowledge among the contexts could still be improved, to make Ambient Intelligence sustainable in the long term. A MCS may be used as the representation and reasoning model for such a system. *Bridge rules* are used to enable this sharing, by allowing each context to access the knowledge acquired by the other contexts. For example, consider two devices in our smart-house, that share their knowledge about the user's location within the house, to reason and optimize their service to this user's needs and desires. But, how can we then evaluate the ways in which the system enables this cooperation? How can we characterise a MCS based on the opportunities for information exchange that it provides to its contexts? To answer such questions, we introduce in MCS the notion of *conviviality*.

Defined by Illich as "individual freedom realized in personal interdependence" [11], conviviality has been introduced as a social science concept for multiagent systems to highlight soft qualitative requirements like user friendliness of systems. Multiagent systems technology can be used to realize tools for conviviality when we interpret "freedom" as choice [12]. Tools for conviviality are concerned in particular with dynamic aspects of conviviality, such as the emergence of conviviality from the sharing of properties or behaviors whereby each member's perception is that their personal needs are taken care of [11]. We measure conviviality by counting the possible ways to cooperate, indicating degree of choice or freedom to engage in coalitions. Our coalitional theory is based on dependence networks [13, 14]; labeled directed graphs where the nodes are agents, and each labeled edge represents that the former agent depends on the latter one to achieve some goal. Here, we draw a parallel between, on the one hand an agent and a context, and on the other hand between a goal and a bridge rule. Particularly, we use a context to encode an agent's knowledge in some logic language, and a bridge rule to describe how an agent achieves its goal, namely to acquire knowledge from other agents, as illustrated in Figure 1. The focus on dependence networks and more specifically on their cycles, is a reasonable way of formalizing conviviality as something related to the freedom of choice of individuals plus the subsidiary relations –interdependence for task achievement– among fellow members of a social system.

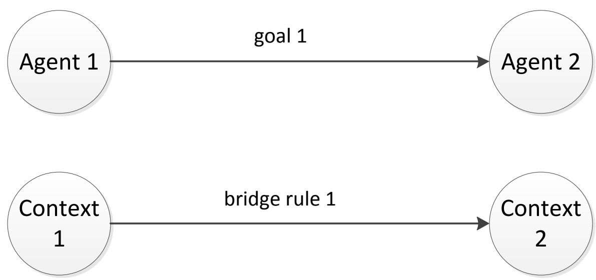

Fig. 1. The dependence network parallelism of contexts as agents, and bridge rules as goals. A labeled arrow, representing a goal, from a to b means that a depends on b to achieve this goal.

In distributed information systems, individual freedom is linked with the choice to keep personal knowledge and beliefs at the local level, while interdependence is understood as reciprocity, i.e. cooperation. Participating entities depend on each other to achieve the enrichment of their local knowledge.

Considering the potential applications of MCS and the notion of conviviality as described above, our main research question is the following:

How to introduce the concept of conviviality to Multi-Context Systems?

This main research question breaks into the following questions:

1. How to define and model conviviality for Multi-Context Systems?
2. How to measure the conviviality of Multi-Context Systems?
3. How to use conviviality as a property of Multi-Context Systems?

Building on the ideas of [15], where we first identified ways in which conviviality tools, and specifically dependence networks and conviviality measures can be used to evaluate cooperation in Contextual Defeasible Logic, we propose:

1. a formal model for representing *information dependencies* in MCS based on dependence networks,
2. conviviality measures for MCS, and
3. a potential application of these tools (model and measures) for the problem of inconsistency resolution in MCS.

So far, most approaches for inconsistency (such as occurrence of α, $\neg\alpha$) resolution in MCS have been based on the *invalidation* or *unconditional application* of a subset of the bridge rules that cause inconsistency [16–19]. They differ in the preference criterion that is applied for choosing among two or more candidate solutions. Here, we propose using the conviviality of the system as a preference criterion. This is based on the idea that removing (or applying unconditionally) a bridge rule affects the information dependency between the connected contexts, and, as a result, the conviviality of the system. We suggest that the optimal solution is the one that minimally affects conviviality.

The rest of the paper is structured as follows: Section 2 presents formal definitions for MCS, as these were originally proposed in [3]. Section 3 proposes a model and measures for conviviality in MCS. Section 4 describes a potential use of conviviality as a property of MCS for the problem of inconsistency resolution. Last section summarizes and presents directions for future work in the field.

2 Multi-Context Systems - Formal Definitions

For the needs of this paper we will use the definition of heterogeneous nonmonotonic MCS given in [3], according to which a MCS is a set of contexts, each composed of a knowledge base with an underlying logic, and a set of bridge rules which control the information flow between contexts. A logic L = ($\mathbf{KB}_L$, $\mathbf{BS}_L$, $\mathbf{ACC}_L$) consists of the following components:

– $\mathbf{KB}_L$ is the set of well-formed knowledge bases of L. We assume each element of $\mathbf{KB}_L$ is a set of "formulas".
– $\mathbf{BS}_L$ is the set of possible belief sets, where the elements of a belief set is a set of "formulas".
– $\mathbf{ACC}_L$: $\mathbf{KB}_L \rightarrow 2^{\mathrm{BS}_L}$ is a function describing the semantics of the logic by assigning to each knowledge base a set of acceptable belief sets.

A *bridge rule* can add information to a context, depending on the belief sets which are accepted at other contexts. Let $L = (L_1, \ldots, L_n)$ be a sequence of logics. An L_k-bridge rule r over L is of the form

$$r = (k : s) \leftarrow (c_1 : p_1), \ldots, (c_j : p_j), \mathbf{not}(c_{j+1} : p_{j+1}), \ldots, \mathbf{not}(c_m : p_m). \quad (1)$$

where $1 \leq c_i \leq n$, p_i is an element of some belief set of L_{c_i}, k refers to the context receiving information s. We denote by $h_b(r)$ the belief formula s in the head of r.

An *MCS* $M = (C_1, \ldots, C_n)$ is a collection of contexts $C_i = (L_i, kb_i, br_i)$, $1 \leq c_i \leq n$, where $L_i = (\mathbf{KB}_i, \mathbf{BS}_i, \mathbf{ACC}_i)$ is a logic, $kb_i \in \mathbf{KB}_i$ a knowledge base, and br_i a set of L_i-bridge rules over $(L_1, \ldots, L_n)$. For each $H \subseteq \{h_b(r) | r \in br_i\}$ it holds that $kb_i \cup H \in \mathbf{KB}_{L_i}$, i.e., bridge rule heads are compatible with knowledge bases.

A *belief state* of an MCS $M = (C_1, \ldots, C_n)$ is a sequence $S = (S_1, \ldots, S_n)$ such that $S_i \in \mathbf{BS}_i$. A bridge rule of form (1) is applicable in a belief state S iff for $1 \leq i \leq j$: $p_i \in S_{c_i}$ and for $j < l \leq m$: $p_l \notin S_{c_l}$. By $br_M = \bigcup_{i=1}^{n} br_i$ we denote the set of all bridge rules of M.

The above definitions are exemplified below. It is not in the scope of this paper to illustrate the many different logics that can be used in MCS. Furthermore, for the sake of clarity, our example is built on propositional logics only.

Example 1. Consider an MCS M, through which the software agents of three research students exchange information and classify research articles that they access in online databases. M contains contexts $C_1 - C_3$, each of which encodes the knowledge of each of the three agents. The knowledge bases for the three contexts are:

$$kb_1 = \{sensors, corba, centralizedComputing \leftrightarrow \neg distributedComputing\}$$
$$kb_2 = \{profA\}$$
$$kb_3 = \{ubiquitousComputing \subseteq ambientComputing\}$$

C_1 collects information about the keywords of the articles and encodes this information in propositional logic. In this case, the article under examination is about $sensors$ and $corba$ (Common Object Request Broker Architecture). C_1 also possesses the knowledge that centralized computing and distributed computing are two complementary concepts. C_2 uses propositional logic to encode additional information about articles, including the names of their authors; in this case $profA$ is the author of the article under examination. Finally, C_3 is an ontology of computing-related concepts, according to which $ubiquitousComputing$ is a type of $ambientComputing$.

The bridge rules that the three agents use to exchange information and collectively decide about the classification of the article are as follows:

$r_1 = (1 : centralizedComputing) \leftarrow (2 : middleware)$
$r_2 = (1 : distributedComputing) \leftarrow (3 : ambientComputing)$
$r_3 = (2 : middleware) \leftarrow (1 : corba)$
$r_4 = (3 : ubiquitousComputing) \leftarrow (1 : sensors), (2 : profB)$

Rule r_1 links the concept of $middleware$ used by C_2 to the concept of $centralized$-$Computing$ of C_1. r_2 expresses that $ambientComputing$ (a term used by C_3) implies $distributedComputing$ (a term used by C_1). r_3 expresses that $corba$ is a type of $middleware$, while r_4 expresses the belief of the third agent (C_3) that articles that are written by $profB$ and that contain $sensors$ among their keywords are about $ubiquitousComputing$.

Equilibrium semantics selects certain belief states of an MCS $M = (C_1, \ldots, C_n)$ as acceptable. Intuitively, an equilibrium is a belief state $S = (S_1, \ldots, S_n)$ where each context C_i respects all bridge rules applicable in S and accepts S_i. Formally, S is an equilibrium of M, iff for $1 \leq i \leq n$,

$$S_i \in \mathbf{ACC}_i(kb_i \cup \{h_b(r) | r \in br_i \text{ applicable in } S\}).$$

Example 2. In the example given above, $S = (S_1, S_2, S_3)$ is the only equilibrium of the system:

$$S = (\{sensors, corba, centralizedComputing\}, \{profA, middleware\}, \emptyset).$$

S_3 is an empty set, since kb_3, the knowledge base of context C_3, is an empty set, $br_3 = \{r_4\}$, namely the set of bridge rules for context C_3 only consists of bridge rule r_4, and r_4 is not applicable in S, because $profB \notin S_2$.

3 Modelling and measuring conviviality in MCS

We mentioned in the introduction that dependence networks have been proposed as a model for representing social dependencies among the agents of a multiagent system. They have also been used as the underlying model for formalizing and measuring conviviality in such systems. In this section, we describe how dependence networks can be used to model the information dependencies among the contexts of a MCS and how conviviality measures can then be applied to measure conviviality in MCS. Our approach is based on the following ideas: (*a*) cooperation in MCS can be understood as information sharing among the contexts; (*b*) it is enabled by the bridge rules of the system; (*c*) therefore, bridge rules actually represent information dependencies among the contexts. Intuitively, that means conviviality will be captured through the different bridge rules that link the contexts.

3.1 Dependence Networks Model for MCS

According to [20], conviviality may be modeled by the reciprocity-based coalitions that can be formed. Some coalitions, however, provide more opportunities for their participants to cooperate with each other than others, being thereby more convivial. To represent the interdependencies among agents in the coalitions, [20] use dependence networks.

In this subsection, we first present Definition 1 from [20], which abstracts from tasks and plans. Then, building on [20]'s definition, we introduce our definition for a dependence network corresponding to a MCS.

A dependence network is defined by [20] as follows:

Definition 1 (Dependence networks). *A dependence network (DN) is a tuple $\langle A, G, dep, \geq \rangle$ where: A is a set of agents, G is a set of goals, $dep : A \times A \rightarrow 2^G$ is a function that relates with each pair of agents, the sets of goals on which the first agent depends on the second, and $\geq: A \rightarrow 2^G \times 2^G$ is for each agent a total pre-order on sets of goals occurring in its dependencies: $G_1 >_{(a)} G_2$.*

To capture the notions of *contexts* and *bridge rules*, we now introduce our definition, Definition 2, for a dependence network corresponding to a MCS, as follows:

Definition 2 (Dependence networks for MCS). *A dependence network corresponding to a MCS M, denoted as $DN(M)$, is a tuple $\langle C, R, dep, \geq \rangle$ where: C is the set of contexts in M, R is the set of bridge rules in M, $dep : C \times C \rightarrow 2^R$ is a function that is constructed as follows: for each bridge rule r (in the form of (1)) in R add the following dependencies: $dep(k, c_i) = \{r\}$ where k is the context appearing in the head of r and c_i stands for each distinct context appearing in the body of r, and $\geq: C \rightarrow 2^R \times 2^R$ is for each context a total pre-order on sets of bridge rules that the context appears in their heads.*

In other words, a bridge rule r creates one dependency between context k, which appears in the head of r, and each of contexts c_i that appear in the body of r. The intuition behind this is that k depends on the information it receives from each of the contexts c_i to achieve its goal, which is to apply r in order to infer s. It follows from Definition 2 that we can have two or more dependencies labeled by the same rule. The application of this rule relies upon all dependencies labeled with this rule. An alternative way to label dependencies would be to use the heads of the rules that these dependencies are derived from, instead of the rules themselves. This is based on the intuition that, when using a rule, a context has actually the goal to derive the conclusion that labels the head of the rule. In that case, however, a new definition of dependence networks may be needed to support both conjunctions and disjunctions of dependencies.

We should also note here that the total preorder that each context defines on the sets of bridge rules may reflect the local preferences of a context, e.g. in the way that these are defined and used in Contextual Defeasible Logic [18, 10]. For sake of simplicity, we do not use this feature in the conviviality model that we describe below. However, it is among our plans to integrate it in future extensions of this work.

To graphically represent dependence networks, we use nodes for contexts and labeled arrows for dependencies among the contexts that the arrows connect. An arrow from context a to context b, labeled as r, means that a depends on b to apply bridge rule r.

In our running example, the dependence network that corresponds to MCS M is the one depicted in Figure 2.

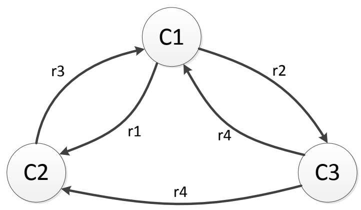

Fig. 2. The dependence network $DN(M)$ of MCS M of the running example. Nodes represent contexts and arrows represent dependencies. An arrow from context a to context b, labeled as r, means that a depends on b to apply bridge rule r.

In this graph, each node corresponds to one of the contexts in M. Dependencies are derived from the four bridge rules of M. For example, there are two dependencies labeled by r_4: each of them connects C_3, which appears in the head of r_4, to one of the contexts C_1 and C_2, which appear in the body of r_4. This actually means that to apply rule r_4 in order to prove that the paper under examination is about ubiquitous computing, C_4 depends on information about the keywords of the paper that it imports from C_1 and information about the authors of the paper that it imports from C_2.

3.2 Conviviality Measures

Conviviality measures have been introduced to compare the conviviality of multi agent systems [20], for example before and after, making a change such as adding a new norm, or policy. Furthermore, to evaluate conviviality in a more precise way, [20] introduce formal conviviality measures for dependence networks using coalitional game theoretic framework. Based on Illich's definition of conviviality as "individual freedom realized in personal interdependency", the notions of interdependency and choice, if we interpret freedom as choice, are stressed. Such measures provide insights into the type of properties that may be measured in convivial systems and thus reveal the quality of the system.

The conviviality measures presented in this work reflect the following Hypotheses:

- H1 the cycles identified in a dependence network are considered as coalitions. These coalitions are used to evaluate conviviality in the network. Cycles are the smallest graph topology expressing interdependence, thereby conviviality, and are therefore considered atomic relations of interdependence. When referring to *cycles*, we are implicitly signifying *simple cycles*, i.e., where all nodes are distinct [21]; we also discard self-loops. When referring to conviviality, we always refer to potential interaction not actual interaction.
- H2 conviviality in a dependence network is evaluated in a bounded domain, i.e., over a $[min, max]$ interval. This allows the comparison of different systems in terms of conviviality.
- H3 there is more conviviality in larger coalitions than in smaller ones.
- H4 the more coalitions in the dependence network, the higher the conviviality measure (ceteris paribus).

Hypothesis H1 is consistent with Definition 2, according to which each bridge rule is mapped to a set of potential dependencies between MCS contexts. The intuition for Hypothesis H3, is that a greater number of collaborating contexts in a MCS offers a greater source of knowledge. This means that each context participating in a large coalition has more available data, than the data it would have in a smaller coalition. Hypothesis H4 is motivated by the fact that a large number of coalitions indicates more interactions among contexts, which is positive in term of conviviality.

Our top goal is to maximize conviviality in the MCS. Some coalitions provide more opportunities for their participating contexts to cooperate than others, being thereby more convivial. Our two sub-goals (or Requirements) are thus:

R1 maximize the size of the contexts' coalitions, i.e. to maximize the number of contexts involved in the coalitions,
R2 maximize the number of these coalitions.

Following the definition of the *conviviality of a dependence network* [20], we define the *conviviality of a dependence network of a MCS M* as

$$\mathrm{Conv}(DN(M)) = \frac{\sum\limits_{c_i, c_j \in C, i \neq j} \mathrm{coal}(c_i, c_j)}{\Omega}, \tag{2}$$

$$\Omega = |C|(|C| - 1) \times \Theta, \tag{3}$$

$$\Theta = \sum_{L=2}^{L=|C|} P(|C| - 2, L - 2) \times |R|^L, \tag{4}$$

where $|C|$ is the number of contexts in M, $|R|$ is the number of bridge rules in M, L is the cycle length, P is the permutation defined in combinatorics, $\mathrm{coal}(c_i, c_j)$ for any distinct $c_i, c_j \in C$ is the number of cycles that contain both c_i and c_j in $DN(M)$ and Ω denotes the maximal number of pairs of contexts in cycles (which produces the normalization mentioned in Hypothesis H2).

This way, the conviviality measurement of a dependence network, which is a rational number in [0,1], can be used to compare different dependence networks, with 0 being the conviviality of an acyclic dependence network and 1 the conviviality of a fully-connected dependence network.

Example 3. Following Equation 2 and the dependence network of M, which is graphically represented in Figure 2, we calculate the conviviality of $DN(M)$ of our running example, as:

$$\mathrm{Conv}(DN(M)) = \frac{10}{\Omega} = 0.208,$$
$$\text{where } \Omega = 480.$$

The result of Example 3 is just a way of comparing the conviviality of different systems. By itself it cannot be used to classify the conviviality of a MCS.

4 Use of conviviality as a property of MCS: Inconsistency Resolution

As we previously argued, conviviality is a property that characterizes the cooperativeness of a MCS, namely the alternative ways in which the contexts of a MCS can share information in order to derive new knowledge. By evaluating conviviality, the system may propose the different ways in which it can be increased, e.g. by suggesting new connections (bridge rules) between the system contexts.

Consider, for example, a MCS, in which a context does not import any information from other contexts. Recommending other contexts that this context could import information from would be a way to increase the conviviality of the system, which would in turn lead to enriching the local knowledge of the context but also the knowledge of the whole system.

4.1 Problem Description

Another way of using conviviality as a property of MCS, which we describe in more detail in this section, is for the problem of inconsistency resolution. In an MCS, even if contexts are locally consistent, their bridge rules may render the whole system inconsistent. This is formally described in [3] as a *lack of an equilibrium*. All techniques that have been proposed so far for inconsistency resolution are based on the same intuition: a subset of the bridge rules that cause inconsistency must be invalidated and another subset must be unconditionally applied, so that the entire system becomes consistent again. For nonmonotonic MCS, this has been formally defined in [16] as diagnosis:

"Given an MCS *M*, a diagnosis of *M* is a pair (D_1, D_2), $D_1, D_2 \subseteq br_M$, s.t. $M[br_M \backslash D_1 \cup heads(D_2)] \not\models \bot$". $D^{\pm}(M)$ is the set of all such diagnoses, while $M[br_M \backslash D_1 \cup heads(D_2)]$ is the MCS obtained from M by removing the rules in D_1 and adding the heads of the rules in D_2.

In other words, if we deactivate the rules in D_1 and apply the rules in D_2 in unconditional form, M will become consistent. As it is obvious, in a MCS it is possible that there is more than one diagnosis that can be applied to restore consistency.

Example 4. In our running example, consider the case that $profB$ is also identified by C_2 as one of the authors of the paper under examination. In this case kb_2 would also contain $profB$:

$$kb_2 = \{profA, profB\}$$

This addition would result in an inconsistency in kb_1, caused by the activation of rules r_4 and r_2. Specifically, rule r_4 would become applicable, $ubiquitousComputing$ and $ambientComputing$ would become true in C_3, r_2 would then become applicable too, and $distributedComputing$ would become $true$ in C_1 causing an inconsistency with $centralizedComputing$, which has also been evaluated as $true$. To resolve this conflict, one of the four bridge rules r_1-r_4 must be invalidated. Using the definition of diagnosis that we presented above, this is formally described as:

$$D^{\pm}(M) = \{(\{r_1\}, \emptyset), (\{r_2\}, \emptyset), (\{r_3\}, \emptyset), (\{r_4\}, \emptyset)\}.$$

Various criteria have been proposed for choosing a diagnosis including:

- the number of bridge rules contained in the diagnosis - specifically in [16] subset-minimal diagnoses are preferred,
- local preferences on diagnoses proposed in [19], and
- local preferences on contexts and provenance information, which have been proposed for Contextual Defeasible Logic [18, 10].

4.2 Proposed Solution

Our approach is to use the conviviality of the resulted system as a criterion for choosing a diagnosis. This actually means that for each candidate solution (diagnosis), we measure the conviviality of the system that is derived after applying the diagnosis, and we choose the diagnosis that minimally decreases the conviviality of the system. The intuition behind this approach is that the system should remain as *cooperative* as possible, and this is achieved by enabling the maximum number of agents to both *contribute* to and *benefit* from this cooperation.

Diagnoses contain two types of changes that one can apply in the bridge rules: invalidation (removal) of a rule; and applying a bridge rule unconditionally, which actually means removing the body of the rule. These changes affect the dependencies of the system as follows: When invalidating or adding unconditionally rule r (as this is defined in (1)) in a MCS M, all the dependencies that are labeled by r are removed from the dependence network of M.

Assuming that $DN(M, D_i)$ is the dependence network that corresponds to MCS M after applying diagnosis D_1, the optimal diagnosis is the one that maximizes the conviviality of $DN(M, D_i)$:

$$D_{opt} = \{D_i : \text{Conv}(DN(M, D_i)) = max\}$$

Example 5. In the running example, there are four diagnoses that we can choose from: D_1-D_4. Each of them requires invalidating one of the four bridge rules r_1 to r_4, respectively. Figures 3 to 6 depict the four dependence networks $DN(M, D_i)$, which are derived after applying D_i. Dashed arrows in Figures 3-6 represent the dependencies that are dropped in each $DN(M, D_i)$, by applying diagnosis D_i.

Following Equation 2 and the four dependence networks, which are graphically represented in Figures 3-6, the conviviality of each $DN(M, D_i)$ is:

$$\text{Conv}(DN(M, D_1)) = \frac{8}{\Omega} = 0.037 \text{ and}$$

$$\text{Conv}(DN(M, D_2)) = \text{Conv}(DN(M, D_3)) = \text{Conv}(DN(M, D_4)) = \frac{2}{\Omega} = 0.009,$$

$$\text{with } \Omega = 216.$$

Since the number of goals $|G|$ is now 3, instead of 4, Ω has a different value than in $DN(M)$. By applying D_1 (Figure 3), only one cycle (C_1, C_2) is removed from the initial dependence network $DN(M)$, illustrated in Figure 2. However, by applying any

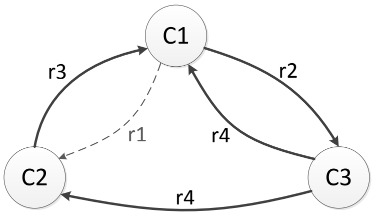

Fig. 3. $DN(M, D_1)$

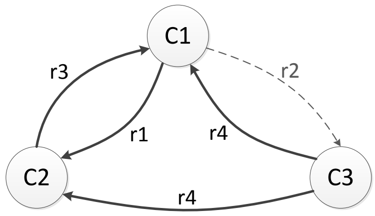

Fig. 4. $DN(M, D_2)$

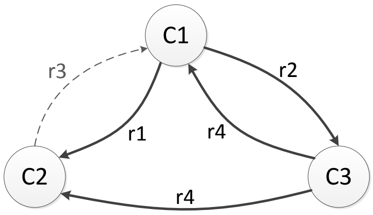

Fig. 5. $DN(M, D_3)$

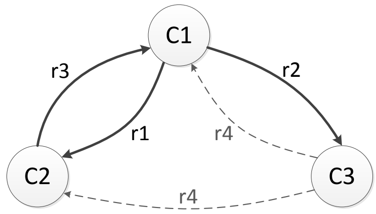

Fig. 6. $DN(M, D_4)$

of the remaining diagnoses D_2-D_4, two cycles are removed from $DN(M)$. Specifically, by applying D_2 (Figure 4), we remove the cycles (C_1, C_3) and (C_1, C_3, C_2). By applying D_3 (Figure 5), we remove the cycles (C_1, C_2) and (C_1, C_3, C_2). Finally, by applying D_4 (Figure 6), we remove the cycles (C_1, C_3) and (C_1, C_3, C_2).

Therefore the optimal diagnosis is D_1. By applying D_1 the system will have the following equilibrium S':

$$S' = (\{sensors, corba, distributedComputing\}, \{profA, profB, middleware\}, \{ubiquitousComputing, ambientComputing\})$$

This approach can also be combined with any of the approaches that have been proposed so far for inconsistency resolution. For example, one may choose to apply the conviviality-based approach only to those diagnoses that comply with some constraints representing user-defined criteria, as suggested in [19]. It can also be combined with preferences on diagnoses proposed by [19] or preferences on contexts suggested by [18, 10]. A study of such combined approaches will be part of our future work in the field.

5 Conclusion

Today, with the rise of systems in which knowledge is distributed in a network of interconnected heterogeneous and evolving knowledge resources, such as Semantic Web, Linked Open Data, and Ambient Intelligence, research in contextual knowledge representation and reasoning has become particularly relevant. Multi-Context Systems

(MCS) are logical formalizations of distributed context theories connected through a set of bridge rules, which enable information flow between contexts. The individual entities, that such systems consist of, cooperate by sharing information through their bridge rules. By reasoning on the information they import, they are able to derive new knowledge. Evaluating the ways in which the system enables cooperations, and characterizing a MCS based on the opportunities for information exchange that it provides to its contexts are. therefore, key issues. The social science concept of conviviality has recently been proposed to model and measure the potential cooperation among agents in multiagent systems. Furthermore, formal conviviality measures for dependence networks using a coalitional game theoretic framework, have been introduced. Roughly, more opportunities to work with other agents increase the conviviality of the system.

This paper is a step toward extending the concept of conviviality, modeled with dependence networks, to Multi-Context Systems. First, we describe how conviviality can be used to model cooperation in MCS. Based on the intuition that contexts depend on the information they receive from other contexts to achieve their goals, i.e., apply specific bridge rules to infer particular information, we define dependence networks for MCS. Furthermore, the aim is for MCSs to be as cooperative as possible, and for contexts to have as many choices as possible to cooperate with other contexts. This results in MCS being as convivial as possible. In order to evaluate the conviviality of a MCS, we apply pairwise conviviality measures and allow for comparisons among MCS. Finally we propose a potential use of conviviality as a property of MCS for the problem of inconsistency resolution. Indeed, without considering contextual information, reasoning can easily encounter inconsistency problems, for example, when considering knowledge in the wrong context. Our approach in this case is based on the idea that the optimal solution is the one that minimally decreases the conviviality of the system.

In further research, we contemplate the need to study alternative ways in which a MCS can be modeled as a dependence network, for example by labeling dependencies with the heads of the rules that they are derived from. We also plan to study the relation between the preference order on goals, which is included in the definition of dependence networks, and preferences on rules, contexts or diagnoses that the system contexts may have. Furthermore, we plan to combine the conviviality-based approach for inconsistency resolution with the preference-based approaches proposed by [19] and [18, 10]. Finally, we want to study how the concept of conviviality and the tools for conviviality can be used in other distributed knowledge models, such as Linked Open Data, Distributed Description Logics [5], E-connections [22] and managed MCS [17].

References

1. Giunchiglia, F., Serafini, L.: Multilanguage hierarchical logics, or: how we can do without modal logics. Artificial Intelligence **65**(1) (1994)
2. Ghidini, C., Giunchiglia, F.: Local Models Semantics, or contextual reasoning=locality+compatibility. Artificial Intelligence **127**(2) (2001) 221–259
3. Brewka, G., Roelofsen, F., Serafini, L.: Contextual Default Reasoning. In: IJCAI. (2007) 268–273
4. Lenat, D.B., Guha, R.V.: Building Large Knowledge-Based Systems; Representation and Inference in the Cyc Project. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA (1989)

5. Borgida, A., Serafini, L.: Distributed Description Logics: Assimilating Information from Peer Sources. Journal of Data Semantics **1** (2003) 153–184
6. Bouquet, P., Giunchiglia, F., van Harmelen, F., Serafini, L., Stuckenschmidt, H.: C-OWL: Contextualizing Ontologies. In: International Semantic Web Conference. (2003) 164–179
7. Parsons, S., Sierra, C., Jennings, N.R.: Agents that reason and negotiate by arguing. Journal of Logic and Computation **8**(3) (1998) 261–292
8. Sabater, J., Sierra, C., Parsons, S., Jennings, N.R.: Engineering Executable Agents using Multi-context Systems. Journal of Logic and Computation **12**(3) (2002) 413–442
9. Antoniou, G., Papatheodorou, C., Bikakis, A.: Reasoning about Context in Ambient Intelligence Environments: A Report from the Field. In: KR, AAAI Press (2010) 557–559
10. Bikakis, A., Antoniou, G., Hassapis, P.: Strategies for contextual reasoning with conflicts in Ambient Intelligence. Knowledge and Information Systems **27**(1) (2011) 45–84
11. Illich, I.: Tools for Conviviality. Marion Boyars Publishers, London (1974)
12. Caire, P., Villata, S., Boella, G., van der Torre, L.: Conviviality masks in multiagent systems. In: 7th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2008), Estoril, Portugal, May 12-16, 2008, Volume 3. (2008) 1265–1268
13. Castelfranchi, C.: The micro-macro constitution of power. Protosociology **18** (2003) 208–269
14. Sichman, J.S., Conte, R.: Multi-agent dependence by dependence graphs. In: Procs. of The First Int. Joint Conference on Autonomous Agents & Multiagent Systems, AAMAS 2002, ACM (2002) 483–490
15. Caire, P., Bikakis, A.: Enhancing Cooperation in Distributed Information Systems Using Conviviality and Multi-Context Systems. In: Multi-disciplinary Trends in Artificial Intelligence - 5th International Workshop, MIWAI 2011, Hyderabad, India, December 7-9, 2011. Proceedings. Volume 7080 of Lecture Notes in Computer Science., Springer (2011) 14–25
16. Eiter, T., Fink, M., Schüller, P., Weinzierl, A.: Finding Explanations of Inconsistency in Multi-Context Systems. In: Principles of Knowledge Representation and Reasoning: Proceedings of the Twelfth International Conference, KR 2010, Toronto, Ontario, Canada, May 9-13, 2010, AAAI Press (2010)
17. Brewka, G., Eiter, T., Fink, M., Weinzierl, A.: Managed Multi-Context Systems. In: IJCAI. (2011) 786–791
18. Bikakis, A., Antoniou, G.: Defeasible Contextual Reasoning with Arguments in Ambient Intelligence. IEEE Trans. on Knowledge and Data Engineering **22**(11) (2010) 1492–1506
19. Eiter, T., Fink, M., Weinzierl, A.: Preference-Based Inconsistency Assessment in Multi-Context Systems. In: Logics in Artificial Intelligence - 12th European Conference, JELIA 2010, Helsinki, Finland, September 13-15, 2010. Proceedings. Volume 6341 of Lecture Notes in Computer Science., Springer (2010) 143–155
20. Caire, P., Alcade, B., van der Torre, L., Sombattheera, C.: Conviviality measures. In: 10th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2011), Taipei, Taiwan, May 2-6, 2011. (2011)
21. Cormen, T.H., Leiserson, C.E., Rivest, R.L., Stein, C.: Introduction to Algorithms. 2nd edn. The MIT Press (2001)
22. Kutz, O., Lutz, C., Wolter, F., Zakharyaschev, M.: E-connections of abstract description systems. Artificial Intelligence **156**(1) (2004) 1–73

A step forward the elicitation of context: application in healthcare.

Nathalie BRICON-SOUF[1] & Emmanuel CONCHON[2]

Université de Toulouse
[1]IRIT-UPS, [2]IRIT-ISIS
à ISIS Rue F. Oulès 81100 Castres France.
{nathalie.souf, emmanuel.conchon}@irit.fr

Abstract. An accurate contextualisation has become relevant in medical applications for numerous data are available which makes personalized care a great expectation of ours. Through this work we will spot on how context is dealt with research and we will present you some pieces of work linked to the capture, the management and the representation of context. Henceforth, we will have a closer look on the problem raised by the elicitation of context and thus explain why it is a major issue in healthcare. It brings us to our own proposition of genuine elicitation model that already has helped experts on their way to determine valid contextual elements. Yet a discussion involves our future challenges in research.

1 Introduction

For many years, the improvement of computer science, as well as medical informatics, has allowed to build health information systems more reliable, better fed, more and more interactive. The computerized patient records, care information systems and especially hospital information systems are nowadays pretty mature. Personal health records (such as the French DMP), records of specialties, shared care records in case of care networks and medical websites, are building and enriching the amount of medical data. Once information exists, it becomes highly desirable to analyze and to process this information. We are entering an era when the medical world is filled up with a lot of information and exchange. But large databases still remain for most part under-used and new challenges arise such as the growing complexity of moving in this large information space and of providing appropriate services to healthcare professionals. Using context should help for instance when focusing on the information in relation to the current activity or by personalizing the interactions for "*Context seems to be particularly relevant in medical applications, where inter-patients variability is extremely high, and where diagnostic and therapeutic decision always need to be properly tailored to the single patient's peculiar situation*"(1).

Finding what is context in the use of such information systems remains a difficult exercise. In this paper, we underline the need for context elicitation and propose a model as a first step towards its computerization.

Before introducing the elicitation problem, we will briefly describe some of the work performed on context. This topic of research has been investigated for several years. Looking back in 1993, McCarthy was already interested in *formalizing context* (2). In this paper, our purpose is not to give an exhaustive view of the research on context but just to have a position on the elicitation problem towards other axes of research on context.

Dey (3) has proposed a definition of context which is commonly used "*Context is any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves.*" The use of context in healthcare has to be improved having said that "*The issue of reaching context awareness and of realizing context-based reasoning are being routinely addressed in mobile and ubiquitous computing[...], but are now recognized as key aspects also for a wide range of other areas, among which patient care*"(1). But even though there are a lot of systems proposed, most of them are still on prototype stage as stated by Orwat(4).

The consideration of contextual information was investigated since the 90's. In (5), the authors reviewed 237 journal articles about context-aware applications and propose a five-leveled classification framework for these papers : *concept and research layer, network layer, middleware layer, application layer and infrastructure layer*, these layers are closed from a classification of context-aware architectures as depicted in figure 1.

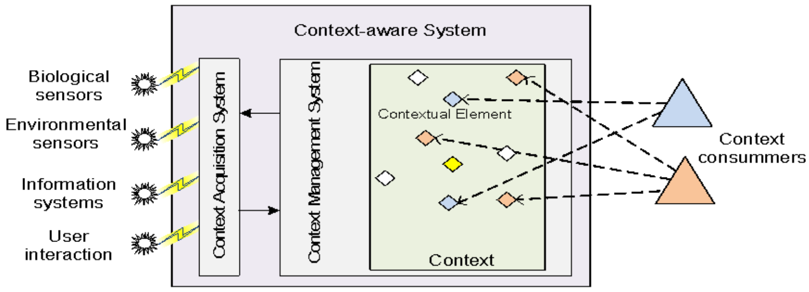

Figure 1: An example of Context-aware architecture

Context-aware architectures are mainly composed of three paradigms: capture of the context, management of the context and representation of the context (see Table 1).

Paradigms	*deals with*
Capture context	**How to get and gather contextual information**?
Manage context	**How to use the context**?
Represent context	**How to represent the contextual knowledge**?

Table 1 : different paradigms for context.

The context acquisition system is in charge of the context capture. This is particularly critical in mobile environments which need to focus on the ability to continuously capture external data (ie information) and to integrate this knowledge into the systems. External data can be acquired from physical sensors, information systems or even from the user interaction (graphical interface, type of device under use, voice, camera and so on). Once the context is captured, it is then provided to the context management system. For example, in healthcare environment, ubiquitous systems are proposed by (6).

Management of context answers the need for processing, storage and presentation of contextual information. Adapting the context to evolving conditions such as mobility or user interaction is also done by context management systems (a.k.a context manager). These systems compose an intermediate layer between context sources and context consumers. Zhang et al.(7) determined a layered context management framework used to provide contextual information to context-awareness systems. Different modules are used: context-source, context query engine, context knowledge base, context discoverer, context application. Wang et al.(8), Soylu et al (9) are some of the authors who present how to manage low level context (issued from captors) towards high level of context (more related to the activities). But to manage the context, these systems need a representation of the context.

Modeling and representing context is obviously a major issue. Zimmerman et al. (10) propose an operational definition of context, introducing a formal extension to the definition through fundamental categories of context (individuality, time, location, activity, relations) and an operational extension through the description of different uses of context. Soylu et al. (9) who work for pervasive learning with a context-awareness system propose a hierarchical representation of context. Two main roots are defined: user and environment; 8 entities are modeled: user, external devices, application, environment, time, history and relation. They also propose a formal representation of context according to its type (the 8 entities), its dynamism, and its level of adaptation (from macro to micro). Brézillon et al. (10) introduce the notion of proceduralized context built from pertinent knowledge in a given situation. Vieira & al. (12) propose a context meta-model in order to design Context Sensitive Systems. This meta-model uses context which appears during enactment, contextual knowledge, and contextual elements which are defined during design.

Different context model features are proposed as shown in a data-oriented survey of context models (13), or in a survey of context awareness in healthcare (14). Among the different context representations one can underline conceptual graphs or C-OWL contextualizing ontologies (15). Description Logic should be used too, as for the Contextualized Knowledge Repository proposed by (16). Context-aware systems aim at providing a context representation, as generic as possible, which is composed of a large number of contextual elements (we will also call them context information). Based on these context-aware systems, context consumers are supposed to be able to find the context elements relevant for their needs. Indeed, the consumers can be very different ranging from a human user to a service deployed in an information system. Therefore, a consumer needs a small subset of contextual elements from the overall representation.

2 The need for context elicitation

Sometimes, it can be rather easy to identify what could be used as contextual element as for instance *localization* which is often a useful contextual element in a mobile situation, or *noise* or *light* which are interesting ones to infer about good user interfaces. Frequently, determining what should be used as context is a major problem. This has been discussed since the beginning of the research on context: Winograd underlined that "*Something is context because of the way it is used in interpretation, not due to its inherent properties"* (17), Schmidt et al. (18) even showed that the feasibility of context acquisition impacts the use and the understanding of context. We bear in mind the following definition for context : "*a set of elements surrounding a domain entity of interest which are considered relevant in a specific situation during some time interval*" (19), and of *Contextual Element* used during the design of the system and referring to pieces of data, information or knowledge that can be used to define the context, in accordance to Vieira et al.(12). These specifications are domain dependent and difficult to build. So we choose to focus not only on how to model the context but mainly on how to elicitate the relevant contextual elements.

Paradigms	*deals with*
Capture context	How to get and gather contextual information:
Manage context	How to use the context
Represent context	How to represent the contextual knowledge:
Elicitate context	**What are the relevant contextual elements for an application?**

Table2 : integration of the elicitation problem

Referring to the literature, we have found few explanations on how this knowledge is built. One of the more detailed processes was found in Souza et al. (19). The authors propose to get the domain entities taxonomy and the contextual elements of a context ontology for data integration from an empirical methodology based on face to face meetings with data integration experts and literature reviews. Mei (20) also proposed a framework of cognitive context modeling. The process of elicitation and modeling is thus based on a spiral lifecycle which involves four phases: identification of technical constraint, stakeholder and end-users; building of cognitive context model; analyzing and optimizing context space and detection of context change. Yang et al. (21) propose a JESS-enabled (Java Expert System Shell) context elicitation system. It involves three phases – form-filling : contextual information is acquired directly from requesters' inputs, context detection (such as positioning), and context extraction to derive contextual information from ontologies. In that case, the system focuses more on how to use useful contextual information rather than on what are the relevant contextual information to capture.

In the medical domain, a lot of information can be used by the application. Some of the latter can be considered as context, surrounding the application domain of interest and relevant for it. Context is very often needed in healthcare. An alert system can be different according to the medical units (eg geriatric unit has not the same needs as pediatric one). A proposed care is not the same according to the availability of specific device (eg:Magnetic Resonance Imaging in a development country or not). Normality of a measure has not always the same value (eg for an infant or for an adult). Contextual situations are frequent and for each situation it is difficult to determine what are the contextual elements.

We propose a model to come across this elicitation process.

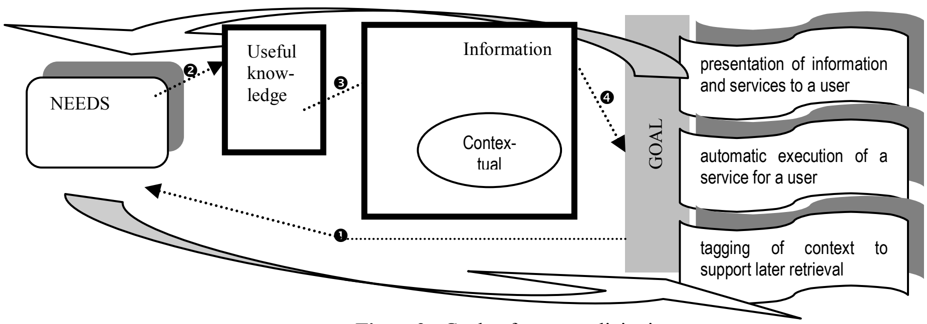

Figure2 : Cycle of context elicitation

A virtuous cycle (see figure 2) is proposed, in order to fix what is the context used for and how it is possible to reach contextual information among the informational space of the application

A first step is to state what is the context used for, we propose the 3 features mentioned by Dey (2) "*presentation of information and services to a user; automatic execution of a service for a user; and tagging of context to support later retrieval*.". It enables us ❶ to specify the application needs. At this stage,❷ we determine the useful knowledge needed to compute the application. Then,❸ we focus on the information used to build this knowledge, which is a difficult part of the work (see figure 3).

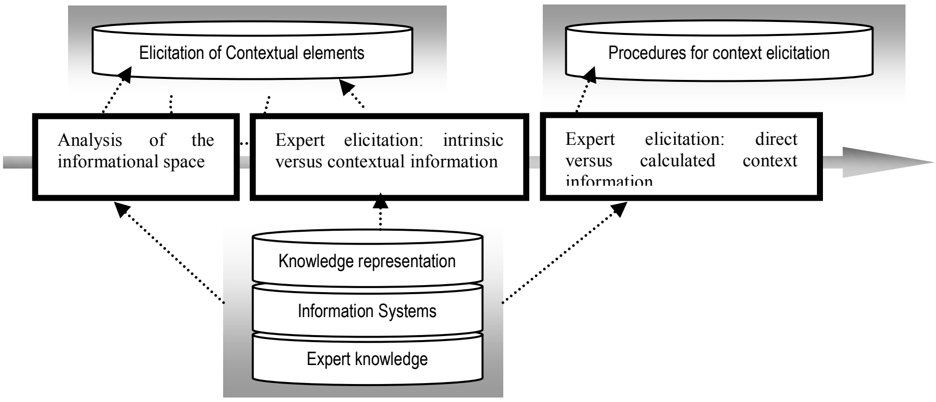

Figure3: Elaboration of the context elicitation

To build the informational space, we dispose of numerous resources of information such as knowledge representations (ontologies, classifications, terminologies for example), data models from information systems (Hospital Information System, medical record for example) and various expert knowledge (habits of a department, prevalence of disease for example). This space must be structured. Discussions about the way information is used are performed to decide if it is contextual or not. This classification is given through the expert analysis of the application needs. Once the useful contextual elements are known, we are able to represent the procedures used to provide it (such as age is useful and is provided by date_of_birth, name as found in HIS). The very last step of our cycle ❹ is the acquisition of the information. This organization of context elicitation in terms of "Goals" - "Needs" - "Useful knowledge", "Intrinsic information" and "Contextual information" helps the discussion with experts and highlights the use of contextual knowledge.

3 Discussion

This model for context elicitation has been used in two different medical situations: one was dedicated to the creation of pedagogical information and the other to the proposition of alerts during prescription(22). Each time, a lot of data were available. A difficult process was to decide what should be retained as relevant context, and the proposed model has helped the different participants to know what was usable and what was relevant as context information. We are now working on a national project (PlasOSoin ANR-2010) about the coordination of healthcare providers during homecare. It is important to understand which information should impact the whole organization of care (for instance, in the context of a heavy patient, maybe two care professionals should come together at home so that they can sit him in the bed). Before using contextual information, it is important to be able to focus on it. This part of work is still at its early stage. Our contribution focus onto the detection of such information, more particularly when they are gathered on large databases (Health care system for instance). The computerization part of the modeled process will be performed during the next step of our work. It can provide new user interfaces to represent and discuss about the useful knowledge and classify some of them as contextual elements, when the mining of the information should provide new ways of reducing the amount of possible elements.

4 Reference

1. Montani S. How to use contextual knowledge in medical case-based reasoning systems: a survey on very recent trends. Artificial intelligence in medicine 2011 Feb;51(2):125–31.
2. McCarthy J. Notes on Formalizing Context. In: Bajcsy R, editor. IJCAI. Morgan Kaufmann; 1993.555–62.
3. Dey AK. Understanding and Using Context. Personal and Ubiquitous Computing. 2001 Feb 28;5(1):4–7.
4. Orwat C, Graefe A, Faulwasser T. Towards pervasive computing in health care – A literature review. BMC Medical Informatics and Decision Making. 2008;8(1):26.
5. yi Hong J, ho Suh E, Kim S-J, Hong J-yi, Suh E-ho. Context-aware systems: A literature review and classification. Expert Systems with Applications . 2009 May;36(4):8509–22.
6. Shin C-sun, Ahn D-in, Kim D-seok, Joo S-chong. Ubiquitous Healthcare System Using Context Information Based on the DOGF. Context. 2007;447–57.
7. Zhang D, Yu Z, Chin C-Y. Context-aware infrastructure for personalized healthcare. Studies in health technology and informatics. 2005 Jan;117:154–63.
8. Wang XH, Zhang DQ, Pung TG an. HK. Ontology Based Context Modelling and Reasoning using OWL. In: CoMoRea, the Second IEEE International Conference on Pervasive Computing and Communications. Orlando, Florida, USA: 2004.
9. Soylu A, Causmaecker PD, Desmet P. Context and Adaptivity in Pervasive Computing Environments: Links with Software Engineering and Ontological Engineering. Journal of Software. 2009;4(9).
10. Zimmermann A, Lorenz A, Oppermann R, Augustin S. An Operational Definition of Context. Context. 2007;:558–71.
11. Brézillon P, Zaraté P, Université LIP, Curie M. Group Decision Oriented View Making : Context-. Decision Support Systems. 2005;X:1–16.
12. Vieira V, Tedesco P, Salgado AC. Designing context-sensitive systems: An integrated approach. Expert Systems with Applications 2011;38(2):1119–38
13. Bolchini C, Curino CA, Quintarelli E, Schreiber FA, Tanca L. A data-oriented survey of context models. ACM SIGMOD Record. 2007 Dec 1;36(4):19
14. Bricon-Souf N, Newman CR. Context awareness in health care: a review. International Journal of Medical Informatics 2007 Jan;76(1):2–12

15. Bouquet P, Giunchiglia F, Harmelen FV. C-OWL : Contextualizing Ontologies. 2003:164–79.
16. Homola M, Serafini L, Tamilin A. Modeling Contextualized Knowledge. 6th Workshop on Semantic Web Applications and Perspectives (SWAP2010). 2010.
17. Winograd T. Architectures for context. Human-Computer Interaction 2001;16(2):401–19.
18. Schmidt A. Ubiquitous Computing – Computing in Context. Lancaster University; 2002.
19. Souza D, Belian R, Salgado AC, Tedesco P. Towards a Context Ontology to Enhance Data Integration Processes. 2008;
20. Mei L. Cognitive Context Elicitation and modeling. Toronto; 2011.
21. Yang SJH, Zhang J, Chen IYL. A JESS-enabled context elicitation system for providing context-aware Web services. Expert Syst. Appl. 2008;34(4):2254–66.
22. Bricon-souf N, Messai R, Bernonville S. A Framework for Context Elicitation in Medicine. 2011;:7–9.

Influence of Context on Decision Making during Requirements Elicitation

Corentin BURNAY , Ivan JURETA and Stéphane FAULKNER

PReCISE Research Center & Department of Business Administration
University of Namur

Abstract. Requirements engineers should strive to get a better insight into decision making processes. During elicitation of requirements, decision making influences how stakeholders communicate with engineers, thereby affecting the engineers' understanding of requirements for the future information system. Empirical studies issued from Artificial Intelligence offer an adequate groundwork to understand how decision making is influenced by some particular contextual factors. However, no research has gone into the validation of such empirical studies in the process of collecting needs of the future system's users. As an answer, the paper empirically studies factors, initially identified by AI literature, that influence decision making and communication during requirements elicitation. We argue that the context's structure of the decision should be considered as a cornerstone to adequately study how stakeholders decide to communicate or not a requirement. The paper proposes a context framework to categorize former factors into specific families, and support the engineers during the elicitation process.

1 Introduction

The process of analyzing the goals of a system to be and the needs of its future users is commonly referred to as Requirements Engineering (RE). Its first step is requirements elicitation, which consists in collecting information about the expectations and needs of stakeholders, in order to identify the problems that need to be solved by the future information system [1].

Collecting requirements implies extensive communication between the requirements engineers and the stakeholders. In this communication process, engineers expect stakeholders to state their requirements on the future system. When communicating requirements, stakeholders choose requirements to state on the basis of implicit assumptions on, e.g., the conditions that will hold in the future, when the information system will be operational. The implicit assumptions remain obscure to the requirements engineer, potentially leading to incomplete and conflicting requirements. It is therefore relevant for RE to study how stakeholders decide which requirements to communicate, and what implicit assumptions they make when doing so.

Classical reasoning theories offer a relevant groundwork for this purpose. Their mathematical approach of reasoning makes it easier to relate it to requirements engineering's formalisms. Some of these theories are based on classical logic, which we consider inadequate to relevently model decision making. As an alternative, we focus on non-monotonic reasoning (NMR) theories, which enable a conclusion to be withdrawn after new information invalidating the decision appears. NMR is similar in many ways to a stakeholder's decision making. Indeed, background knowledge in a decision situation is hardly ever complete. NMR consequently assumes that the stakeholder makes normality assumptions when drawing conclusions: anything the stakeholder does not know is assumed to be as expected [2]. Yet, it may happen these assumptions turn out to be incorrect, which may in turn invalidate the conclusions drawn from the assumptions. Such reasoning is called common-sense reasoning, and is considered to be non-monotonic. Reiter [3] proposed default logic, an influential model of non-monotonic reasoning. In default logic, the normality assumption states that, in absence of evidence to the contrary, default rules hold.

We assume that default logic can relevantly model reasoning that occurs in stakeholder's decision making during requirements elicitation. For example, concluding from the sentences "The stakeholder wants her accounting software to have the feature to import data from the software which keeps track of wages levels" that "The software can actually import data from wages software" involves for a stakeholder to consider a default assumption such as "The two software programs can be connected and exchange data". This is a typical example of a default assumption made by the stakeholder, that is not explicitly communicated to engineers but which constitutes an important requirement. In fact, the default assumption - or the default requirement - may be untrue and invalidate the initial requirement, e.g. the two software programs are proprietary applications closed to connection with other software. During elicitation, the engineer should know what default requirements underline a communicated requirement, since knowing the default requirement helps to decide if the basis requirement that the stakeholder communicated is correctly defined.

Identifying the default assumptions is not a simple task. There exist many NMR studies that demonstrate the impact of contextual factors on default reasoning [4–9]. Therefore, engineers should consider that many parameters play a role in the process of adopting or not a default requirement. In other words, there is a variability in decision making which is inherent to the context. Such variability is likely to lead to some bias between real stakeholders' requirements and what is stated in the result of the elicitation process. We argue the use of a context framework - a list of the different categories of information that forms context - is of great help to identify default requirements of the stakeholders under such circumstances, i.e. information that they implicitly assume in a given context, but do not say explicitly. The requirements engineers could use the framework as a tool to determine which questions to ask during the elicitation in order to identify the default requirements, thereby verifying the completeness and precision of requirements that the stakeholders have provided.

The paper proposes a preliminary discussion and study of factors identified in AI and applied to the context of Requirements Engineering. The paper also reports results from preliminary experiments that suggest the influence of context on requirements elicitation and confirm the need for a new experimental framework to consider different factors and their influence on decision making. The paper then proposes a context framework to categorize factors that influence communication during the requirements elicitation step.

2 Internal and External Domains

We see two distinct categories of factors that are likely to influence stakeholder's decision making. This distinction is important to the requirements engineer, because the two categories do not have the same impact for requirements identification. The engineers should consider such distinction during the elicitation.

The first category is on factors related to human cognition and factors that depend on the way an individual uses knowledge and heuristics in reasoning. It is typically this kind of factors that have been studied by NMR community [4–7]. These are referred to as *internal factors*, since they depend on the individual, and not on the situation in which the individual draws conclusions. Consequently, their implications for the requirements engineers is limited.

The second category includes factors which are not, strictly speaking, specific to the person. They are consequently referred to as *external factors*. These factors also influence how a stakeholder makes the decision to communicate a requirement. As a requirements engineer, it is interesting to understand such factors' influence, because they apply to any stakeholders. It is important to understand how factors influence the choice of default requirements used by the stakeholder, in order to account for such influence in further requirements treatment. Only a few papers [8–10] deal with the influence of external factors on NMR. None of them focuses on the case of elicitation.

External factors are different from internal factors because they may be relevant for one individual, and irrelevant for another: this observation is particularly important in the scope of the RE process. Requirements engineers aiming to design a new political forecasting system should consider factors that influence the entire set of stakeholders, namely the political candidates. To do so, they should establish a clear distinction between internal and external domains. If they do not consider this distinction, they would collect default requirements related to actual needs of one candidate (due to internal factors), but which do not apply to other candidates. For instance, beliefs a candidate has are an internal factor that is likely to influence her expectations regarding the system-to-be and consequently lead to potential bias in the requirements elicitation: the candidate believes that charisma is a key to be elected, and therefore wants the future forecasting system to account for such element in the forecasting function. Yet charisma is maybe not considered as relevant for a second candidate, thereby invalidating the previous requirement. This issue is due to the internality of beliefs: what is true for one is maybe not true for the other, and the factor is consequently considered as internal. On the other hand, the factor "Size of Group" identified by Elio and Pelletier [9] is a relevant factor for any candidate, i.e. politician with largest group of potential voters is likely to be predicted as the winner of elections in a democratic political system. This factor is related to the environment of candidates, and not to their own perception of this environment. "Size of Group" is therefore classified as an external factor. Such external factors should be considered by engineers as a real and unbiased source of default requirement.

In short, the elicitation of requirements is based on the decision of stakeholders to communicate or not some requirements to engineers. The decision process can be adequately modeled by default logic, in which anything the stakeholder does not know is assumed to be as expected. Such an assumption is called a default, and we will call it in RE a default requirement. Such defaults are opaque to the engineers, making it necessary to attempt to uncover them during elicitation. We argue this can be done using a definition of context, discussed later in this paper. This definition categorizes external factors, that we consider to be acceptable sources of defaults. As an answer to the limited attention to external factors influencing default reasoning, we report results of our preliminary experiments, with the aim to replicate Elio and Pelletier's research in the scope of managerial decision making and RE. Conclusions of this replication lead to a discussion about the limitations of such empirical approach in RE, and suggest the need for a broader experiment design.

3 Experiment

3.1 Design

So far, few external factors have been identified in the literature, and none of them have been validated in the particular case of RE. The first step toward understanding of NMR in RE decision making during elicitation is to validate the previously identified external factors in relation to RE decision making problems. To test the influence of these factors, we ask subjects to consider a basic problem and to provide a first conclusion. We then introduce similar problems with additional potentially influencing factors. Questions in the questionnaires are benchmark problems requiring "basic default reasoning". A benchmark problem always introduces at least two objects that are supposed to respect a rule, then informs the subject one object does not respect the rule, i.e. there is an exception to the rule, and finally asks the subject whether the remaining object respects that rule.

In our experiment, problems deal with classification of two objects that are described using one or more default rules. Problems then inform subjects that there is at least one exception to the default rules. Finally, subjects are asked to provide a conclusion about the remaining object. The benchmark answer that is expected is that no exception object for a default rule should have impact on the conclusions drawn about any other object for which that rule applies. The different problems submitted to subjects are highly contextual due to the RE orientation of the experiment. Consequently, exercises are not presented under the typical benchmark form [11], but rather under the form of a story. The intrinsic structure of problems are always similar. The goal is to give a plausible organizational context to subjects. Consider the following problem as an example of problems proposed in our questionnaire:

> "An engineer collects requirements for a system to be used in a factory. The engineer typically considers information about employees of the factory to establish a list of requirements *(Default Rule)*. Information about wages *(Object 1)* and number of working hours *(Object 2)* are examples of information about employees. However, information about wages has not been considered by the engineer *(Exception)*. What do you think about engineer's behavior toward working hours? *(Benchmark Question)*"

In the case of the above problem, we expect subjects to select a conclusion from a list of four different answers regarding the remaining working hours information *(Object 2)*:

1. Benchmark - *Exception* has no bearing on the *Benchmark Question*: e.g. *Object 2* have been considered by the engineer unlike *Object 1*;
2. Exception - *Exception* also applies to the *Benchmark Question*: e.g. *Object 2* have not been considered by the engineer like *Object 1*;
3. Other - *Exception* does imply another exception, but with different characteristics: e.g. a wage per hour measure have been preferred by the engineer;
4. Can't Say - The subject cannot choose one of the former proposition.

In order to decrease the chance of finding out the pattern of good answers, the four solutions are randomly ordered, i.e. the benchmark answer will not always be the first answer. Our experiment is designed to confirm the influence of five external factors: Specificity, Similarity, Size of the Group, Nature of the Group, Perspective. The experiment only aims to determine whether already known factors can be confirmed in a RE context. The purpose is not to understand how people are reasoning in terms of the steps that they may be taking, but what may influence them while reasoning.

Similarity refers to the availability of information regarding the similitude between objects. It describes the set of commonalities that are shared by objects, and has two levels, i.e., low or high. For instance, high similarity would inform subjects that *Object 1* and *Object 2* are both provided by the accounting department. The low similarity would state that the two objects come from different departments. Such factor should impact reasoning since high similarity suggests that objects may have been subject to the same set of intentions [9].

Specificity refers to the availability of information about the way the exception is violating the default rule. It is any piece of information that acts as a justification of the exception, and has two levels, i.e., vague or specific. The specific level would state taht *Object 1* is not following *Default Rule* because it is violating privacy rules. Low Specificity level would only explain that *Object 1* was not considered by the engineer, without any explanation about the way the exception happened.

Nature of the Group refers to whether the considered entities are real or artifact objects. Wason and Shapiro [10] argue that the difficulty to reason about abstract objects may be due to "the failure to generate alternatives in order to derive the correct solution". Facing something she does not think real, a subject will have difficulties to reason about it, and this can impact her final conclusion. The artifact level would state that Object 1 is an artificial one, e.g. a number of winks. Note that we interpret the word artificial as "out of the plausible set of possibility for a given context".

Size of the Group refers to the relative importance of classes that are compared in the benchmark problem. The factor has two levels: with or without size information. The former would state that only 20% of wages information is accessible against 95% of working hours information. The later would avoid any reference to the relative size of objects.

Perspective refers to the point of view adopted by subjects. Elio and Pelletier tested subjects answering to questions while taking the "perspective" of a human or of a robot. They observe that robots should be cautious [8], and thereby that people tend to privilege the "Can't Say answer" when they are asked to adopt the robot perspective. We replicate the "who is answering?" question with different actors more suitable to a RE approach. Our experiment proposes "For Me" and "For Other" answers.

3.2 Questionnaires

In our experiment, we test interactions between two factors. Considering the definition of factors, we focus on the interaction between similarity and specificity on the one hand, and nature and size on the other hand. Perspective is considered for both combination of factors. Questionnaires consist of 21 problems, which are designed to test previous factors. Three questions have been created in order to cover the possible routine of the questionnaire and to make subjects more careful about their way of thinking and taking information into account. These three additional questions are constructed with exactly the same structure as the 18 relevant exercises, but they differ in number of default rules and objects to which rules apply. This makes the reasoning even more difficult, but does not influence the individual in other exercises. The answers to these latter problems will not be taken into account for the final analysis and are designed to avoid bias while entertaining volunteers.

3.3 Subjects

The first questionnaire ("For Other") have been submitted to a group of 68 management science students at the Department of Business Administration, University of Namur. All subjects were bachelor students and were asked to answer within a 50 minutes time period. The second questionnaire ("For Me") has been submitted to the same group, at a later date. Both sessions took place at the same place and under the same circumstances. Subjects were asked to answer during class time, and were not compensated for participating in the study. The sample is considered to be representative of the population. Firstly, we study how human makes decision in a given context: there should be no difference between decision making performed by students and by engineers, since reasoning is a process that any human performs. Secondly, subjects are management students, trained to make decisions under uncertainty as in real conditions. Thirdly, we test external factors, i.e., factors that are given to subjects and have the same influence for students and for requirements engineers. Consequently, we consider the external validity of this experiment as acceptable.

3.4 Procedure

Questions were distributed into two questionnaires: we refer to them as "For Me" and "For Other". Once divided, all the questions were randomly selected and inserted in their own questionnaire. The answerer can use any kind of material. The assignment clearly mentions that there is no best answer, but that some answers are better than others. It also tells the subject that the objective of the questionnaire is to better understand how managers are reasoning in a situation of general and imperfect information about a decision problem.

3.5 Results

Table 1 provides observed proportions of answers for each point of view. It appears that the perspective adopted by subjects when they answer questions has an impact on the decision they make. Answering for another person seems to make subjects more reluctant to select the benchmark. Rather than selecting an inappropriate answer, subjects prefer the "Can't Say" answer. This could be interpreted as cautiousness: because others will suffer from potential drawback related to subjects' decision, they prefer not to give an answer at all.

Table 1. Proportion of Answers by Perspective

	Benchmark	Exception	Other	Can't Say
For Other	.467	.079	.165	.289
For Me	.579	.151	.083	.187

Significance tests are performed using the same approach as Elio and Pelletier, namely repeated measure ANOVA on the proportion of answers in each category of answer and for each problem [8]. Elio and Pelletier's approach implies that the category of answer is another factor. Perspective's influence is confirmed through a significance test. A significant effect is observed for the answer category. Results also suggest a significant interaction between answer category and Perspective [$F(3, 183)=5.923$, $p=0.000$].

Beside Perspective, other factors are tested in this experiment. Table 2 summarizes proportions of answers for each possible combination of factor, for both perspective. The reference question is considered as the Vague-Low and Real-Without Number versions of problems. In a nutshell, these are problems where factors' modality are neutral.

Table 2. Proportions of Answers

	Benchmark	Exception	Other	Can't Say
Specific-High	.559	.088	.191	.162
Specific-Low	.485	.059	.206	.250
Vague-High	.338	.221	.221	.221
Reference "For Others"	**.500**	**.059**	**.162**	**.279**
Artefact-With	.485	.059	.147	.309
Artefact-Without	.456	.044	.147	.353
Real-With	.412	.044	.103	.441
Specific-High	.613	.194	.032	.161
Specific-Low	.742	.032	.113	.113
Vague-High	.339	.452	.065	.145
Reference "For Me"	**.645**	**.081**	**.081**	**.194**
Artefact-With	.565	.097	.081	.258
Artefact-Without	.581	.081	.065	.274
Real-With	.500	.194	.145	.161

The reference question for the "For Other" perspective has 50% of benchmark answer, while it ranges between 50% and 75% for the Robot perspective of Elio and Pelletier [8]. Subjects in our experiment do better under the "For Me" perspective, with a 65% part of benchmark answer, against an average below 60% for Elio and Pelletier. Table 2 suggests mitigated results. On the one hand, variations in proportions of answers for the Specificity and Similarity are observed, which make the influence of factors visible. On the other hand, variations for the Nature and Size are minimal.

The same conclusion is drawn for the "For Me" perspective. One of the most surprising aspects is probably the majority of "Exception" in the case of Vague-High questions. Under the "For Me" perspective, influence of Similarity and Specificity is evident. Size and Nature slightly decrease the proportion of benchmark in favor of the "Can't Say" answer, but no clear pattern appears.

Results do not give clear conclusions. Impact of factors in the two perspectives are identical, even though influence is stronger under the "For Me" perspective. In the case of Similarity and Specificity, impact is clearer, particularly in the case of the "For Me" perspective. Under this perspective, a difference of 35% is observed between the two factors. Tests displayed in Table 3 confirm that some external factors identified in the literature review do not have, in our experiment, the effect that they have been argued to have in prior research.

Table 3. Significance Test for the Factors Dimension

	For Other		For Me	
Factor(s)	Fisher	P-Value	Fisher	P-Value
Answer	18.486	.000***	44.940	.000***
Answer*Similarity	1.966	.120	16.273	.000***
Answer*Specificity	2.468	.063*	7.725	.000***
Answer*Specificity*Similarity	2.363	.072*	3.349	.020**
Answer	19.856	.000***	30.457	.000***
Answer*Nature	.328	.805	1.672	.174
Answer*Size	.666	.574	1.758	.157
Answer*Nature*Size	2.296	.079*	1.420	.238

For the two Perspectives, there is always a significant influence of answer. Table 3 shows that influence of Similarity and Specificity is confirmed under the "For Me" perspective. Regarding the "For Other" questions, a two-ways interaction between answer category and Specificity [F(3,201)=2.468, p=0.063], together with a three-ways interaction between answer category, Similarity and Specificity [F(3,201)=2.363, p=0.072] is observed. Impact of Similarity in such perspective is not confirmed. Results for the "For Me" perspective are negative. P-values are often larger than 15 percent and influence of Size and Nature cannot be concluded. We emphasize here the conspicuous influence of Perspective on factors being tested. A single factor has always a clearer impact under the "For Me" perspective. The combination of factors leads sometimes to a strengthening effect [F(3,201)=2.298, p=0.079] under the "For Other" perspective.

3.6 Preliminary Conclusion

Our experiment confirms the influence of Similarity and Specificity as observed in prior NMR research, in RE context. It also highlights how factors' influence depends on the conditions in which they are tested. Perspective significantly impacts the strength of factors, and factors like Size or Nature seems to have no influence on reasoning unless when working together. Results obtained in this experiment are different in some regards from what Elio and Pelletier propose in their own studies. This can be explained by the way problems are introduced to subjects. The importance of studying factors influencing decision making during the elicitation of requirements cannot be denied. However, we observe that classical empirical approach used in NMR literature is not well suited to tackle the issue of elements influencing decision making during elicitation. Factors cannot be identified and tested in an isolated way. We argue these factors are part of a broader concept, namely the context.

In the second part of the paper, it is argued that a set of factors defines a context, and that this context influences selection of default requirements. The factor itself is only a mean to define the content of a problem. The standalone factor has no clear role without considering the context in which it occurs. Since factors influence context, and context influences decision making, it is important to keep a control on factors. Not only those that are tested, but also those that form context and that are not identified as being in the scope of the experiment. This position justifies why our experiment and the experiment from Elio and Pelletier are so different: contexts are different. Some factors related to RE have probably interacted with external factors initially identified, thereby leading to a significantly different context compared to what Elio and Pelletier originally studied. Since decision making is strongly related to the context, it is interesting to study what conditions or factors influence the decision process.

4 Context as a Reference for Requirements Elicitation

Experiments performed on external factors are always based on a context. Sometimes, the content of a context is limited, with only a few aspects being explicit. This is the case of a simple Benchmark, proposed by Lifschitz [11] and broadly used in NMR research.

A and B are heavy.
Heavy blocks are normally located on this table.
A is not on this table.
What about B?
B is on this table.

In such case, context is a room with a table and the influence of undefined factors can probably be neglected: no actor interacts, no information can influences the individual. In the case of this paper however, context is more complex due to the richer decision problem. As a consequence, unidentified factors must be considered because they can interact with elements to test, and alter the influence these standalone elements could actually have in a basic context.

As an answer, we propose another approach of external factors. We argue that the "Pick up and test factor" method is not adequate because factors are part of a context, and as a consequence should be considered within this context. Doing so requires a relevant definition of context. Given the RE orientation of this paper, we consider definitions of context that are proposed in literature on context-aware computing. Numerous lexical definition of context exist. However, few of them enable to consider how factors relate to context. As a consequence, we focus on three particular definition of context: Dey's general context definition [12], Lenat's decomposition of context into twelve dimensions [13] and Zimmermann et al's de operational definition of context [14]. Each of these definitions emphasizes different "categories" or "dimensions" of the context. Table 4 summarizes context's categories identified by former authors:

Table 4. Categories of Context

	Location / Time	Individuality	Knowledge	Relation	Activity
Dey	X	X		X	
Zimmermann et al.	X	X		X	X
Lenat	X	X	X	X	

"Location/Time" category consists of every factor that is related to the time when the context occurs or to the place where it can be located. "Individuality" category forms the basis of the context. It gathers all the factors that are related to entities existing inside the context. An entity can be of different nature: natural, human or artificial [14]. "Granularity" consists of all the factors dealing with the quantity or the level of information that is provided about the context (whatever its nature). "Activity" refers to factors that deal with the set of goals and intentions of individuals existing in the context. "Relationship" corresponds to any factor dealing with the relationships between previously defined entities, i.e. in what way two or more entities are related to one another. "Knowledge" refers to the information that is part of the context and not specifically attached to an individual. An example is law: individuals know that laws exist, yet they do not always know their content. Laws are parts of "external" knowledge that depends on the environment and not on the individual.

Based on Table 4, we propose a context framework which consists in six major categories : Spatio-Temporal, Items, Knowledge, Relationship and Activity categories. The framework gathers categories of context that were not proposed together in former definitions. Based on this framework, it is possible to capture the major aspects of context that influence communication and decision during RE elicitation. The framework enables to define the different categories that form context and to classify factors that are proven to influence decision making.

The framework could be used in several ways. Firstly, it could be used as a tool to support future empirical research on human reasoning. For instance, it enables to account for the differences between the experiment that is proposed in this paper and experiments from AI literature. The reason why replication of results fails in our experiment is that we actually replicate a single aspect of context, e.g. granularity category with similarity or specificity factors. We did not pay specific attention to the remaining categories of context. Thereby, we designed an experiment with different experiment settings and consequently different impacts of factors on default reasoning. Therefore, the framework could be used to list items - or other categories - in order to accurately identify the context's definition and decreasing the gap in experimental settings.

Secondly, we consider that the framework offers numerous opportunities for requirements engineers. It could support the requirements engineers in the identification of the complete set of factors that are likely to have a bearing on the communication with stakeholders. The advantages of categorizing elements of the context in well defined dimension are multiple. Firstly, it is a way to structure a requirements elicitation process by considering each category of the framework. Secondly, the categorization offers a taxonomy to support further formalizations, by decomposing a blurry context notion into a well structured one. Thirdly, it enables to avoid subjectivity during elicitation, since external elements of the context are given to each stakeholder. Finally, it is possible to relate categories of context to other important RE notions: does that part of the context make sense to the user or the machine? Who or what can sense it? Is it relevant regarding the system-to-be?

5 Conclusion

The paper presents preliminary results of an empirical study on decision making during requirements elicitation. We focus our attention on non-monotonic theories, and more precisely on default logics, which offer an adequate groundwork to empirically study human decision making. The paper discusses the distinction between internal and external factors, and tests factors initially studied by Elio and Pelletier in the case of RE. We find in our experiment different results than what is proposed in the NMR literature. We argue these differences can be explained by the definition of the context on which individuals base their decisions. As a response, we suggest to transfer our empirical efforts on the study of context. Our claim is that during elicitation, the requirements engineer should try to obtain as much information about context as possible, in order to uncover default requirements. By looking empirically at which dimensions of context are relevant, we provide a definition of context useful for the elicitation of requirements. Requirements engineers can use that definition - which lists dimensions of context - as a tool to determine the questions they should ask stakeholders to identify stakeholder's default requirements, and thereby verify the completeness of requirements that the stakeholders have provided. Further research is required to validate the context framework, but we believe the framework can be a useful tool to support requirements engineers during requirements elicitation.

References

1. Nuseibeh, B., Easterbrook, S.M.: Requirements engineering: a roadmap. In: ICSE - Future of SE Track. (2000) 35–46
2. Brewka, G., Niemelä, I., Truszczynski, M.: Preferences and nonmonotonic reasoning. AI Magazine **29**(4) (2008) 69–78
3. Reiter, R.: A logic for default reasoning. Artif. Intell. **13**(1-2) (1980) 81–132
4. Ford, M., Billington, D.: Strategies in human nonmonotonic reasoning. Computational Intelligence **16**(3) (2000) 446–468
5. Ford, M.: Human nonmonotonic reasoning: the importance of seeing the logical strength of arguments. In: Synthese. (2005) 71–92
6. Vogel, C.: Human reasoning with negative defaults. In: Lecture Notes in Artificial Intelligence: Practical Reasoning, Springer, Berlin (1999) 606–621

7. Hewson, C., Vogel, C.: Psychological evidence for assumptions of path-based inheritance reasoning. In: Proceedings of the Sixteenth Annual Conference of the Cognitive Science Society. (1994) 409–414
8. Elio, R., Pelletier, F.J.: Human benchmarks on ai's benchmark problems. In: Proceedings of the Fifteenth Annual Conference of the Cognitive Science Society. (1993) 406–411
9. Elio, R., Pelletier, F.J.: On relevance in non-monotonic reasoning: Some empirical studies. In: Relevance: American Association for Artificial Intelligence 1994 Fall Symposium Series. (1994) 64–67
10. Wason, P.C., Shapiro, D.: Natural and contrived experience in a reasoning problem. Quarterly Journal of Experimental Psychology **23** (1971) 63–71
11. Lifschitz, V.: Benchmark problems for formal nonmonotonic reasoning, Berlin: Springer (1989) 202–219
12. Dey, A.K.: Understanding and using context. Personal and Ubiquitous Computing **5**(1) (2001) 4–7
13. Lenat, D.: The Dimensions of Context-Space. Technical Report, Cycorp (1998)
14. Zimmermann, A., Lorenz, A., Oppermann, R.: An operational definition of context. In: Proceeding of the Sixth International and Interdisciplinary Conference on Modeling and Using Context. (2007) 558–571

Context Aware Decision System in a Smart Home: Knowledge Representation and Decision Making Using Uncertain Contextual Information

Pedro Chahuara, François Portet, and Michel Vacher *

Laboratoire d'Informatique de Grenoble,
UMR 5217 CNRS/UJF/G-INP, FRANCE
{pedro.chahuara,francois.portet,michel.vacher}imag.fr

Abstract. This research addresses the issue of building home automation systems reactive to voice for improved comfort and autonomy at home. The paper presents a complete framework that acquires data from sensors and interprets them, by means of IA techniques, to provide contextual information for decision making. The system uses a two-level ontology to represent the different concepts handled during the processing which also contains SWRL instances to automatise some of the reasoning. The focus of this paper is on the relationship between the knowledge representation and the decision process which uses a dedicated Markov Logic Network approach to benefit from the formal logical definition of decision rules as well as the ability to handle uncertain facts inferred from real data. The entire approach is situated w.r.t. the Sweet Home project whose aim is to make possible context-aware voice command at home.

Keywords: Ambient intelligence and pervasive computing, Decision making, Frameworks for formalizing context and context-aware knowledge representation, Reasoning under uncertainty

1 Introduction

As the development of Smart Homes (SH) has gained a growing interest among many communities — such as medicine, architecture, computer sciences, etc. — two major challenges have emerged in the area of Ambient Intelligence. Firstly, the need for knowledge representation models featuring high readability, modularity and expressibility. Secondly, the requirement to develop decision making methods that can leverage knowledge models to take context — the particular situation under which a decision is taken — and its uncertainty into account. Indeed, in most cases the information gathered to infer context comes from sources affected by uncertainty and imprecision.

In the literature, logical models, mostly ontologies and logic rules, seem to have reach a consensus due to the high readability and expressibility they offer.

* This work is part of the Sweet-Home project founded by the French National Research Agency (Agence Nationale de la Recherche / ANR-09—VERS-011)

The Open AAL platform [19] uses an ontology that describes in-home entities belonging to low and high abstraction levels. The framework designed around this ontology is appropriate to facilitate the integration of devices from different providers, as they share a common taxonomy, and the implementation of computational methods to make context inference. The independence between knowledge representation and inference methods guarantees modularity, however it does not take advantage of the reasoning capacities supported by logical reasoners, as the only purpose of the ontology is to be an artefact of integration. Chen et al. [3] have proposed a method to perform activity recognition in home, an important element of context awareness, by using subsumption checking in an ontology, but uncertainty is not supported in this work. A more general approach was designed by Liao [9], in which some context elements, such as level of risk, are defined through logic rules using RDF-based events to perform activity recognition. However, uncertainty of the information sources is not considered even if a prior probability of risk is estimated. Answer Set Programming (ASP) is another logic approach for representation and reasoning that has been applied by Mileo et al. [11] to estimate the evolution of the inhabitant's health state. They present a framework that can properly deal with reasoning under incompleteness and uncertainty. Furthermore, the knowledge encoded in the ASP rules could be integrated into an ontology as well. Although their approach is very relevant for context recognition, they have not developed formal decision models containing essential elements such as utilities, risks and actions. On the side of decision methods for SH dealing with uncertainty, several Bayesian approaches have been suggested, as in the SOCAM project [5]. Influence diagrams [7], which are based on Bayesian networks, have been also applied to model the causal relation among decision actions, uncertain variables, risk, and utilities [14, 4]. However in these works, the decision process is not supported by a formal knowledge representation that can be exploited in other tasks besides decision.

It seems that there exists a gap between the development of formal models to represent knowledge in pervasive environments and the methods for decision making that must act under uncertain information. We are tackling this problem in the Sweet-Home project, a new smart home system whose main man-machine interaction modality is based on audio processing technology. Our proposed solution involves the representation of concepts by means of ontologies and a set of logical rules. It takes advantage of description logic reasoners and SWRL for situation recognition and obtains a system adaptable to other SH implementations, as well. In the decision stage, a part of the logical rules is employed to construct an influence diagram based on Markov Logic Networks (MLN), a statistical method that makes probabilistic inference from a model consisting of weighted logic rules. The rest of this paper describes the Sweet-Home framework. Sections 2 presents the project and Section 3 the framework architecture. Section 4 shows the ontologies and how situation recognition is performed. A detailed explanation of our decision making model is given in Section 5. Finally Section 6 concludes with a brief discussion.

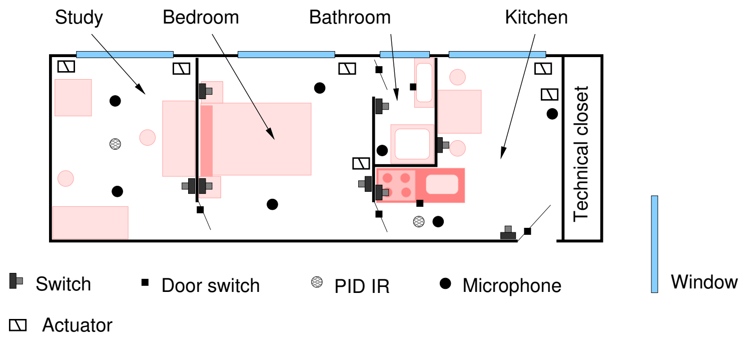

Fig. 1. The DOMUS smart home.

2 The Smart Home context

This research is related to the Sweet-Home project (`http://sweet-home.imag.fr`), a French national supported research project aiming at designing a new smart home system based on audio technology which focuses on three main aspects: to provide assistance via *natural human-machine interaction* (voice and tactile command), to ease *social interaction* and to provide *security reassurance* by detecting situations of distress. In this project, the SH under consideration is DOMUS, a flat filled in with sensor technology which was set up by the Multicom team of the Laboratory of Informatics of Grenoble. This $30m^2$ suite flat, depicted in Figure 1, is equipped with sensors and actuators such as infra-red presence detectors, contact sensors, video cameras (used only for annotation purpose), etc.

This kind of smart home can support daily living by making context-aware decision base on the current situation the user is. To illustrate this support let's consider the following two scenarios:

Scenario 1 *The inhabitant arrives to her apartment at night and goes to the bedroom immediately, forgetting to lock the door. She prepares to sleep and turns all the lights off but the bedside lamp as she usually reads before sleeping. After some minutes, she turns off the lamp and, at this moment, from the sequence of her interactions with the environment, the system recognizes that she is about to sleep, and a relatively dangerous situation is recognized as the main door is not locked. A decision could result in sending a message through a speech synthesizer – considering the risk of interrupting her rest– to remind her of the state of the door.*

Scenario 2 *The inhabitant wakes up in the middle of the night and utters the vocal order "Turn on the light". This simple command requires context information (location and activity) to realize which light to turn on and what the appropriate intensity is. In this case, the system decides to turn on the bedside lamp with a middle intensity since the ceiling light could affect her eyes sensitivity at that moment.*

From these scenarios it can be noticed that contextual information, such as location and activity, plays a major role to deliver appropriate support to the user. In this paper we define Location and Activity as follows:

Definition 1 (Location). *$l(t) \in L$, where L is the set of predefined locations in the SH and $t \in \mathbb{N}$ is the time, specifies where the inhabitant is located.*

In this work a specific area corresponds to a room and we assume a single inhabitant in the environment.

Definition 2 (Activity). *Routine activities performed during daily live; such as, sleeping, cooking, or cleaning. In an instant t the activity might be undetermined; so an activity occurrence, a is defined in an interval of time, $A(t_{begin}, t_{end})$. Thus $A : t_b, t_e \rightarrow a$, $t_b, t_e \in \mathbb{N}$ and $t_b < t_e$*

Moreover, many more information can be inferred from the raw data such as agitation, communication, etc. They are defined as sources of information:

Definition 3 (Source of Information). *The system contains a set of variables V that describes the environment. A source of information is a variable $V_i \in V$ with domain $Dom(V_i)$ representing the information provided by a sensor or a inference process i.*

Definition 4 (System state). *If Υ is the set of possible values of V, a system state is an assignment $v \in \Upsilon$ making $V = \{V_1 = v_1, V_2 = v_2, ..., V_n = v_n\}$*

The Situation is defined by:

Definition 5 (Situation). *A situation $S \subset \Upsilon$ is defined by a set of constraints $C = \{C_1^{k_1}, C_2^{k_2}, ..., C_m^{k_m}\}$, where each constraint $C_i^{k_i}$ establish a set $A_i \subset DOM(V_{k_i})$ to constraint the value of a source of information V_{k_i}. Thus $S = \{v / \forall C_i^{k_i} \in C, v_{k_i} \in A_i\}$*

For example, if we have two sources of information, V_1 and V_2, corresponding to the the state of the main door and the location of the inhabitant, a situation can be defined by constraints, C_1^1, C_2^2, holding the following sets: $A_1 = \{open\}, A_2 = \{study, bedroom\}$.

Definition 6 (Temporal Situation). *Let's consider a temporal sequence of system states $\delta = (v_1^{t_1}, v_2^{t_2}, ..., v_n^{t_n})$ where t_i is the time of occurrence. A temporal situation R, is defined by a set of constraints $T = \{T_1, T_2, ..., T_m\}$, where each T_k defines a pair of situations (S_k^1, S_k^2) and an interval $[a_k, b_k]$ such as $R = \{(v_i^{t_i}, v_j^{t_j}) / \forall T_k \in T, v_i^{t_i}, v_j^{t_j} \in \delta, v_i \in S_k^1, v_j \in S_k^2, a_k \leqslant t_j - t_i \leqslant b_k\}$*

Thus, if a temporal constraints T_1 establish an interval $[t_i, t_j]$, a temporal situation will be recognized when two instances of the situations S^1 and S^2 occur with a difference of time falling into the interval. In the rest of the paper we refer to temporal situations simply as situations.

Based on our study of the context, we define it as follows:

Definition 7 (Context). *Set of informations characterizing the circumstance under which an inference is made.*

The main usage of context is disambiguation. When a situation is recognized, context provides the complementary information to evaluate the circumstance in terms of a certain quality $Q \in \{risk, comfort, safety, ...\}$. Let's a function F_Q assigning a value, in the scope of Q, to a situation S. The final value of F_Q depends on the information contained in the context $\kappa : F_Q(S|\kappa)$. This function in our work is given by the decision model.

3 The Sweet-Home System: an Audio-controlled Smart Home

The input of the Sweet-Home system is composed of the information from the domotic system transmitted via a local network and information from the microphones transmitted through radio frequency channels. The microphone data is processed by an audio processing chain delivering hypotheses about the sound or the sentences being uttered by the user [18]. All these streams of information (audio and domotic) are captured by an intelligent controller which interprets them to recognize situations and makes decisions. The diagram of this intelligent controller is depicted in Figure 2. The knowledge of the controller is defined using two semantic layers: the *low-level* and the *high-level* ontologies which are described in the next section. Besides knowledge representation, another role of the ontologies is to store the events from which inference is carried out.

The estimation of the current situation is carried out through the collaboration of several processors, each one being specialized in a specific source of information. All processors share the knowledge specified in both ontologies and use the same repository of facts. Furthermore, the access to the knowledge base is executed under a service oriented approach that allows any processor being registered to be notified only about particular events and to make inferred information available to other processors. This data and knowledge centred approach ensures that all the processors are using the same data structure and that the meaning of each piece of information is clearly defined among all of them.

We have considered that the main aspects for situation recognition are the location of the inhabitant, the current activity and the period of the day. These informations are useful to eliminate ambiguity in the decision making process. For example, in Scenario 2, when the vocal order *Turn on the light* is uttered by the inhabitant, in order to decide which light must be activated, the controller infers inhabitant's location. Furthermore, there can be many lights in the same room, so if the command is given in the middle of the night after the inhabitant has interrupted her sleep, knowing the previous activity and time period helps to infer that the best choice of light are the bedside lamps. Other works have also reckoned location and activity as fundamental for context inference [11, 16].

In order to perform location and activity inference, two independent modules were developed and integrated in the framework. Due to space limitation the reader is referred to [2, 1] for further details.

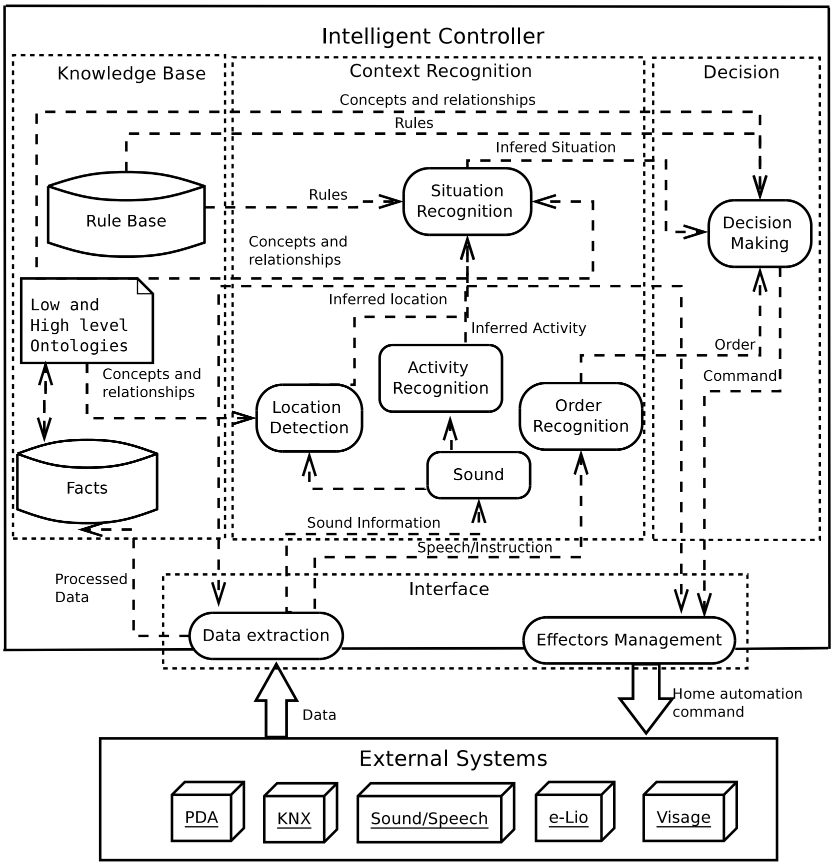

Fig. 2. The Intelligent Controller Diagram.

4 Ontologies and Rules for Situation Recognition

The intelligent controller performs inference in several stages, from raw input data until the evaluation of situations. Each event is produced by the arrival of a sensor information. These events are considered of low level as they do not require inference. Once they are stored in the facts base, processing modules are executed hierarchically (e.g., location then activity then situation). Thus, each inference corresponding to a high level event is stored in the database and used subsequently by the next module. Within the controller architecture, other inference modules can be added without compromising the processing of the other components. The two ontologies were designed, not only for domain knowledge representation, but also for storing the events resulting from the processing modules. Furthermore, situations are defined within the ontologies allowing description logic reasoners to evaluate if a situation is happening. Consequently, the importance of the ontology transcends the mere description of the environment.

4.1 Low and High Level Ontologies

The knowledge of the controller is defined using two semantic layers: the *low-level* and the *high-level* ontologies. The former ontology is devoted to the representation of raw data and network information description. State, location, value and URI of switches and actuators are examples of element to be managed at this level. The high level ontology represents concepts being used at the reasoning level. These concepts are organized in 3 main branches: the Abstract Entity, the Physical Entity,and the Event concept that represents the transient observations of one abstract entity involving zero or several physical entities (e.g., at 12:03 the dweller is sleeping). Instances in the *high-level* ontology are produced by the inference modules (e.g. activity, location, and situations) after treating information coming from sensors. This separation between low and high levels makes possible a higher re-usability of the reasoning layer when the sensor network and the home must be adapted [8].

Figure 3 shows some of the concepts and relations of both ontologies. The ABoxes serve as an example of the state of the fact base at a certain moment. Let's refer to the scenario 1 when the inhabitant turns the bedside lamp off to sleep. The controller updates devices states in the low level ontology and it can be inferred, still at a low level, that every light in the room is off. In the high level ontology, the interaction with the switch lamp is stored as a device event having time and room as properties. At this stage, the module on charge of location is requested and it gives a straightforward answer as the switch is placed in the bedroom. Then, the evidences of the inhabitant being in the bedroom, having all lights turned off, and the evening as the period of the day, can be used to infer that the current activity is sleeping. Finally, these inferences provide the context on which situation recognition is applied. Under the same scenario, if the inhabitant forgot to close the main door and a situation was defined for this case, the situation will be labelled as detected in the ontology. Detected situations are treated by the decision module explained in section 5.

4.2 Application of SWRL to Situation Recognition

A situation can be seen as a temporal pattern of the system state which is given by the facts base. Ontologies provide an appropriate foundation for situation recognition since they store all the facts and a complete semantic description of the environment as well. Furthermore, temporal representation can be achieved by means of role properties among event concepts defining temporal relations such as *previous* and *next* which, through chaining property of OWL2, can generate the *after* and *before* relations. Under some restrictions, Datalogs describing situations as logic rules can be transformed in description logic and written on ontologies [6]. However the scope of this approach is very limited as it does not allow to specify complex definitions. Even when it is limited to safe rules, it overcomes several restrictions of description logics while having the definitions still as part of the ontology. In addition, SWRL builtin functions further extend the semantics of context definitions.

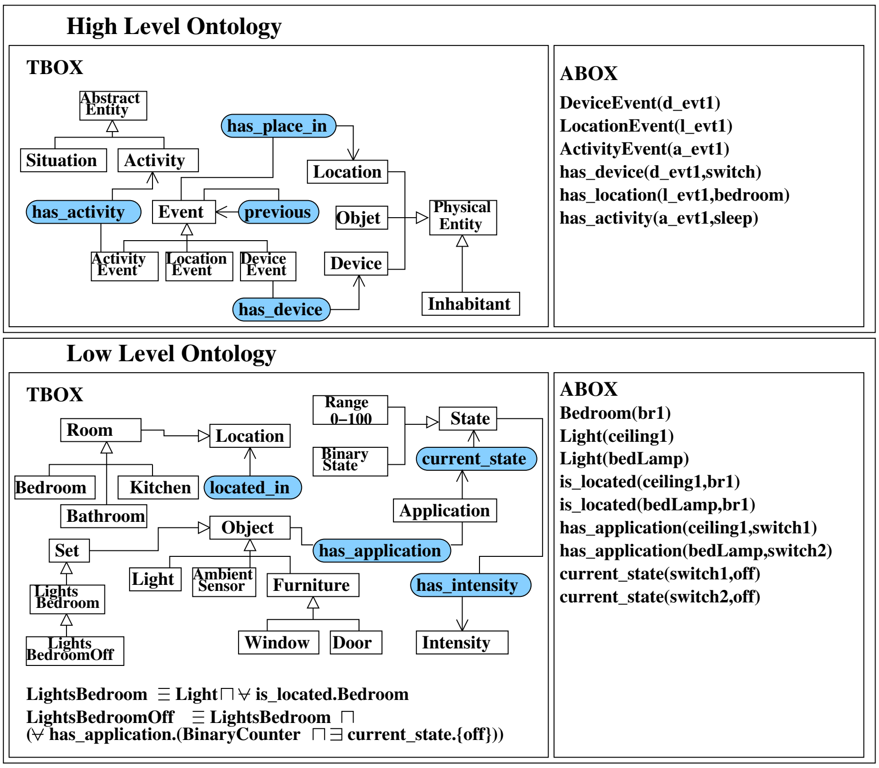

Fig. 3. The low and high level ontologies.

A possible situation definition in SWRL, based on scenario 1, is given below:
DeviceEvent(?d), has_ associated_ object(?d, door),
takes_ place_ in(?d, kitchen),state_ value(?d, open),
DeviceEvent(?l), has_ associated_ object(?l, setLights),
takes_ place_ in(?l, kitchen),state_ value(?l, off), temp:after(?l,?d)
$\rightarrow$ *current_ state(LightsOffOpenMainDoor, detected)*

We assume that these events reflect the current state of the system. Note that a high level, events can be defined by means of sets of devices as well.

5 Decision Making

The decision making module is the main component of the intelligent controller. When a situation is recognized, this module employs the high level knowledge in order to construct dynamically a decision model that takes into account the context and its degree of uncertainty. In this section we briefly describe the base method used for decision making, and give details about our implementation.

5.1 Markov Logic Networks (MLN)

MLN [15] combines first-order logic and Markov Networks, an undirected probabilistic graphical model. A MLN is composed of a set of first-order formulas each one associated to a weight that expresses a degree of truth. This approach soften the assumption that a logic formula can only be true or false. A formula in which each variable is replaced by a constant is *ground* and if it consists of a single predicate is a *ground atom*. A set of ground atoms is a *possible world*. All possible worlds in a MLN are true with a certain probability which depends on the number of formulas they agree with and the weights of these formulas. A MLN, however, can also have hard constraints by giving a infinite weight to some formulas, so that worlds violating these formulas have zero probability. Let's consider F a set of first-order logic formulas, i.e. a knowledge base, $w_i \in \mathbf{R}$ the weight of the formula $f_i \in F$, and C a set of constants. During the inference process [15], every MLN predicated is grounded and Markov network $M_{F,C}$ is constructed where each random variable corresponds to a ground atom. The obtained Markov network allows to estimate the probability of a possible world $P(X = x)$ by the equation 1:

$$P(X = x) = \frac{1}{Z} exp\left(\sum_{f_i \in F} w_i n_i(x)\right) \quad (1)$$

where $Z = \sum_{x' \in \chi} exp\left(\sum_{f_i \in F} w_i n_i(x')\right)$ is a normalisation factor, χ the set of possible worlds, and $n_i(x)$ is the number of true groundings of the i-th clause in the possible world x. Exact inference in MLN is intractable in most cases, so Markov Chain Monte Carlo methods are applied [15].

Learning an MLN consists of two independent tasks: structure learning and weight learning. Structure can be obtained by applying machine learning methods, such as Inductive Logic Programming, or rules written by human experts. Weight learning is an optimisation problem that requires learning data. The most applied algorithm in the literature is *Scaled Conjugate Gradient* [10].

5.2 Influence Diagrams with MLN

Influence diagrams [7] are probabilistic models used to represent decision problems. They result from an extension of Bayesian networks – composed only of state nodes – by the inclusion of two types of node: actions and utilities. An action node is a variable corresponding to a decision choice. The state nodes in the Bayesian network represent how the variables in the problem domain are affected by the actions. Finally, utility nodes are variables that represent the value obtained as consequence of decisions made. Formally, given a set of actions A, an assignment of choices to these actions a, $a \in A$, is taking according to its utility function, $U : X \to [0, 1]$, where X is the state of the random variables in the network after the decision is made. The expected utility for the assignment of choices a is computed as: $EU(a) = \sum_X P(X|a, e)U(X)$ Where e is the evidence given to the network. The process of finding the optimal decision, i.e.

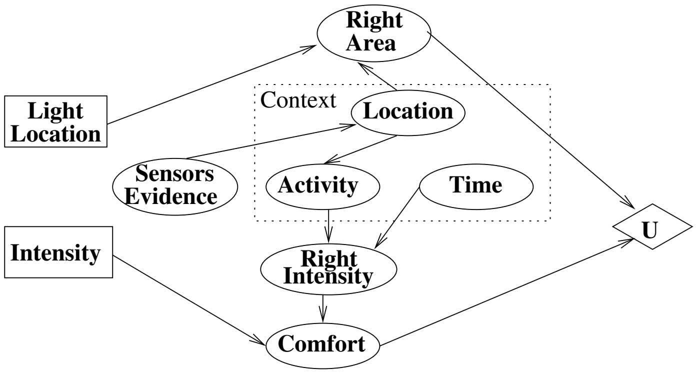

Fig. 4. Influence diagram for a decision after a vocal order is recognised.

the assignment of choices to actions, consists of solving the Maximum Expected Utility (MEU) problem which demands to compute every possible assignment: $argmax_a EU(a)$.

Figure 4 shows an example of Influence Diagram, based on the scenario 2, where a decision is made as a response to a vocal order *Turn on the light*. In this case, the setting of action variables, represented by rectangular nodes, designates which lights devices in the environment are operated and the intensity of the lights. Round nodes are the state nodes affected by the decision. Among the state nodes, information belonging to the context is bound within a dashed area. There are two variables influencing directly the utility: the comfort of the inhabitant and the suitability of the activated lights location that ideally should be the same of the inhabitant. Note that this location is not easy to determine in some cases since the inhabitant could be moving in the environment while uttering the vocal order.

Since a Markov network is a more general probabilistic model than a Bayesian network, Influence diagrams can also be implemented by means of MLN [12]. Nath et al. [13] have proposed an algorithm that evaluates all the choices in a set of actions without executing the whole inference process for each choice resulting in an efficient way to estimate the optimal assignation. We have considered this approach suitable for implementing decision making in our framework for two main reasons: First at all, MLNs are defined through logical rules which can be stored in an ontological representation, using the concepts already established in order to keep a standard vocabulary besides achieving decision model readability. Secondly, it allows to deal with the uncertainty related to context variables and evidence.

A MLN for the influence diagram in figure 4 can be defined as follows:

Predicates	**Domain**	**Type**
Intensity	{low,medium,high }	Action
Comfort	{low,medium,high }	Utility
Location	{bedroom,kitchen,toilet... }	State
LightLocation	{bedroom,kitchen,toilet... }	Action
Activity	{sleep,cook,clean,dress... }	State
RightArea	{good,bad,acceptable }	Utility

Weight	**Rule**
2.0	$LightLocation(l) \wedge Location(l) \rightarrow RightArea(good)$
1.8	$LightLocation(l1) \wedge Location(l2) \wedge NextTo(l1, l2)$ $\rightarrow RightArea(acceptable)$
2.0	$Intensity(d) \wedge Activity(a) \wedge RightIntensity(a, d)$ $\rightarrow Comfort(high)$
1.2	$Intensity(d1) \wedge Activity(a) \wedge RightIntensity(a, d2) \wedge d1! = d2$ $\rightarrow Comfort(bad)$

Utility Values

U(RightArea(bad))=-1	U(RightArea(acceptable))=0	U(RightArea(good))=1
U(Comfort(low))=-1	U(Comfort(medium))=0	U(Comfort(high))=1

Evidences(When they are not factual, then probability is indicated)

Location(bedroom)[0.8]	Location(kitchen)[0.15]	Location(toilet)[0.05]
Activity(sleep)[0.75]	Activity(read)[0.17]	Activity(dress)[0.08]
RightIntensity(sleep,low)[0.95]	RightIntensity(read,low)[0.80]	
NextTo(kitchen,bedroom)	NextTo(bedroom,toilet)	

This model must be constructed dynamically since the probability of context variables, location and activity, can not be known *a priori*. These variables are provided by the specialised modules of location and activity that supply also a probability for their inference results. These results are uncertain evidences. To introduce them into the MLN model, we have followed an approach similar to the one implemented by Trans et al. [17]. Therefore, after the vocal command is received, the context variable values are requested by the decision module, the decision model is constructed using the rules saved in the ontology and decision inference is performed using MLN. Given fixed values for the action nodes, LightLocation and Intensity; the inference will give the probability for each grounding of the utility predicates, RightArea and Comfort. Let's consider the case where action nodes are fixed as $a = (LightLocation(kitchen), Intensity(low))$, then for this configuration we obtain the following expected utility:

$$EU(a) = \sum_{x \in \{bad, acceptable, good\}} P(RightArea(x) \mid a).U(RightArea(x)) + \sum_{x \in \{low, medium, high\}} P(LightLocation(x) \mid a).U(LightLocation(x))$$

The optimal assignment of actions will be the one having the maximal EU.

6 Discussion and Future Work

Dealing with context in pervasive environments involves treating uncertainty, imprecision, and incompleteness; and so far, not a single method can overcome all these problems. Therefore Ambient Intelligence projects must rely on the application of several methods sharing a common base and serving each one a specific purpose. Our proposed framework is an attempt towards this direction.

Decision making by means of Markov logic networks seems very promising as it can take the best of logic and probabilistic models: a simple and clear representation in the framework while being able to treat uncertainty through probabilistic inference. However, as most of probabilistic models, MLN learning requires a considerable amount of data to estimate the optimal parameters. Unfortunately, corpora on pervasive environments with annotated data useful for decision making is rarely available. Furthermore, to the best of our knowledge there is no available corpora for decision making from vocal orders. Therefore, we plan, in the short term, to carry out experiments on a real SH platform that will provide us with data to optimize our decision models and to test the complete framework in realistic circumstances.

To further improve our framework, we intend to work on two improvements: the first one relates a tighter integration of the decision model with the ontology. We consider very interesting the possibility to check for coherence of the decision model rules by means of an ontology reasoner. In general, this integration is not trivial as MLN rules are defined in first-order logic, while description logic and safe rules are only a subset of first-order logic. Our second idea consists in extending the semantics of the situation recognition module in order to be able to define situations in terms of complex events.

References

1. Pedro Chahuara, Anthony Fleury, Michel Vacher, and François Portet. Méthodes SVM et MLN pour la reconnaissance automatique d'activités humaines dans les habitats perceptifs: tests et perspectives. In *Actes de la conférence RFIA 2012*, pages 340–347, Lyon, France, jan 2012.
2. Pedro Chahuara, François Portet, and Michel Vacher. Fusion of Audio and Temporal Multimodal Data by Spreading Activation for Dweller Localisation in a Smart Home. In *STAMI Series, Space, Time and Ambient Intelligence*, 2011.
3. Liming Chen and Chris Nugent. Ontology-based activity recognition in intelligent pervasive environments. *International Journal of Web Information Systems*, 5(4):410–430, November 2009.
4. Berardina De Carolis and Giovanni Cozzolongo. C@sa: Intelligent home control and simulation. In Ali Okatan, editor, *International Conference on Computational Intelligence*, pages 462–465. International Computational Intelligence Society, 2004. International Conference on Computational Intelligence, ICCI 2004, December 17-19, 2004, Istanbul, Turkey, Proceedings.
5. Tao Gu, Hung Keng Pung, and Da Qing Zhang. A service-oriented middleware for building context-aware services. *Journal of Network and Computer Applications*, 28(1):1–18, January 2005.

6. P. Hitzler and B. Parsia. Ontologies and rules. In S. Staab and R. Studer, editors, *Handbook on Ontologies*, pages 111–132. Springer, Heidelberg, 2nd edition, 2009.
7. R. Howard and J. Matheson. Influence diagrams. *Readings on The Principles and Applications of Decision Analysis*, 1 and 2:720, 1981.
8. Michael Klein, Andreas Schmidt, and Rolf Lauer. Ontology-centred design of an ambient middleware for assisted living: The case of soprano. In *30th Annual German Conference on Artificial Intelligence (KI 2007)*, 2007.
9. Hsien-Chou Liao and Chien-Chih Tu. A rdf and owl-based temporal context reasoning model for smart home. *Information Technology Journal*, 6(8):1130–1138, 2007.
10. Daniel Lowd and Pedro Domingos. Efficient weight learning for markov logic networks. In *In Proceedings of the Eleventh European Conference on Principles and Practice of Knowledge Discovery in Databases*, pages 200–211, 2007.
11. A. Mileo, D. Merico, and R. Bisiani. Reasoning support for risk prediction and prevention in independent living. *Theory Pract. Log. Program.*, 11(2-3):361–395.
12. Aniruddh Nath and Pedro Domingos. A language for relational decision theory. In *Proceedings of the Sixth International Workshop on Statistical Relational Learning*, Leuven, Belgium, 2009.
13. Aniruddh Nath and Pedro Domingos. Efficient belief propagation for utility maximization and repeated inference. In *AAAI*, 2010.
14. Takashi Nishiyama, Shinpei Hibiya, and Tetsuo Sawaragi. Development of agent system based on decision model for creating an ambient space. *AI & Society*, 26(3):247–259, 2011.
15. Matthew Richardson and Pedro Domingos. Markov logic networks. *Machine Learning*, 62(1-2):107–136, 2006.
16. Bill Schilit, Norman Adams, and Roy Want. Context-aware computing applications. In *In Proceedings of the Workshop on Mobile Computing Systems and Applications*, pages 85–90. IEEE Computer Society, 1994.
17. Son D. Tran and Larry S. Davis. Event modeling and recognition using markov logic networks. In *Proceedings of the 10th European Conference on Computer Vision: Part II*, pages 610–623, Marseille, France, 2008. Springer-Verlag.
18. Michel Vacher, Dan Istrate, François Portet, Thierry Joubert, Thierry Chevalier, Serge Smidtas, Brigitte Meillon, Benjamin Lecouteux, Mohamed Sehili, Pedro Chahuara, and Sylvain Méniard. The sweet-home project: Audio technology in smart homes to improve well-being and reliance. In *33rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC 2011)*, Boston, USA, 2011.
19. Peter Wolf, Andreas Schmidt, and Michael Klein. Soprano - an extensible, open aal platform for elderly people based on semantical contracts. In *3rd Workshop on Artificial Intelligence Techniques for Ambient Intelligence (AITAmI'08), 18th European Conference on Artificial Intelligence (ECAI 08), Patras, Greece*, 2008.

ARCOE-12 Workshop Notes