IADIS International Journal on Computer Science and Information Systems Vol. 7, No. 2, pp. 117-128 ISSN: 1646-3692

CONTEXT DETECTION IN ONTOLOGY

Achraf Mtibaa, Higher Institute of Electronics and Communication MIRACL Laboratory, University of Sfax, Tunisia

Maha Maalej, Higher institute of computing and multimedia MIRACL Laboratory, University of Sfax, Tunisia

Faïez Gargouri, Higher institute of computing and multimedia MIRACL Laboratory, University of Sfax, Tunisia

ABSTRACT

In the requirement specification phase, several users express the same requirement. In fact, they are situated in different contexts. The context detection of such expressed requirement faces many problems. Among these problems, we cite the implicit aspect of a context. In this work, we propose an approach to detect the requirement context by expliciting this context notion with contextual parameters. Our approach is able to infer the context after knowing the contextual parameters of the user requirement. To realize our proposed approach, we updated the ContextOntoMR prototype.

KEYWORDS

Context Parameters, Context Detection, Ontology, Inference, Requirement Specification

1. INTRODUCTION

The knowledge engineering designates the interface between applications and users. These applications must take into account the users changing context (location or activity for example) to adapt their behaviour. Besides, there exist certain needs for context aware applications. For example, these applications should be able to integrate and interpret context information. To support these tasks, defining the context is needed. Brézillon [BRE 11] defines context as the set of appropriate conditions and other influences that make a situation unique and understandable. In fact, the context is integrated in ontologies that enable automated context reasoning and efficient knowledge sharing. The term "ontology" has been defined as "an explicit specification of a given field conceptualization" [GRU 93]. In this context, ontology is an explicit encoding of a domain model that can be shared and reused,

while a context may be seen as an explicit encoding of a domain model that is expected to be local and which may contain a part of the domain. Taking into account that the strengths of ontologies are the weaknesses of contexts and vice versa, a number of approaches have been developed to join the benefits of the two concepts.

The requirement specification (RS) has several processes that consist of identifying, analyzing and documenting customer's requirements. Among the problems faced in this step, we find the problem of expressing requirements through the same system's actors. This is due to the several actors' contexts. Many problems related to the multitude of contexts arise in the RS. In the context of RS, a requirement is specified by a user in given context. To recognize this context we have to explicit it. In addition, it may depend on different factors that make it detectable. Besides, a requirement can be expressed in different contexts. Different requirements can as well be expressed in the same context. For this reason, we propose in this paper an approach to consider the context after knowing his parameters. It is necessary to infer the requirement context to know if the new requirement exists in the ontology.

The remaining of this paper is organized into four points. After introducing our paper, the second point provides the related work of previous researchers that used two concepts ontology and context as well as some systems using this coupling (ontology and context). The third point details the context parameters that we defined. The fourth point presents the approach proposed in order to establish the contextual dimension in an ontology related to the RS. Finally, we focus on the implementation of our contributions in *ContextOntoMR* prototype [MTI 11], developed by our team.

2. MOTIVATION AND RELATED WORK

It is imperative to recognize that developing context aware applications should be supported by adequate context information modeling. These techniques improve the maintainability of context-aware applications and reduce their complexity. Knowledge engineering designates the interface between applications and users. These applications must take into account the users changing context to adapt their behavior. However, the context information models do not address this aspect of computing. Besides, there are some requirements for context aware applications. For instance, these applications should be able to integrate and interpret context information.

To support these tasks, it is necessary to base the work on ontologies that enable formal knowledge representation. A study of the literature revealed to us the approaches suggested for context representation in ontology COBRA [CHE 03], CMF [FLO 05], C-OWL [BOU 03]. Some works have been interested in modeling context by using parameters. [HAM 11] have defined a meta-model for context parameters. The context meta-model represents the context definition. The context meta-model in [HAM 11] which is based on concepts of ontology is represented by UML formalism. RDF language (Resource Description Framework) of W3C is used to represent contextual information at level 1 of the model. This information is related to the user's profile, its location, the devices he uses and the time. RDFS language of W3C (RDF schema) is used to represent contextual information at level 2 of the model. The latter is related to the context elements namely context properties, context data types and the container.

In the context of Software Engineering, especially, in the analysis and design phase, requirements engineering can benefit from ontologies in terms of knowledge representation and process support. This step deals with gathering the desired system functionality from the customers. An ontology can be used to describe requirements specification documents [MAY 04], [DEC 05] and formally represent requirements knowledge [LIN 96], [WOU 00]. Consequently, it is possible to use ontologies which are generally more precise and helps to get hold of formal system specification. As the degree of expressiveness can be adapted to the actual needs, ontologies can cover structured as well as formal representations [WOU 00]. Moreover, ontologies can be used to support requirements management [MAY 04], [LIN 96].

We put into practice our approach by updating the *ContextOntoMR* prototype [MTI 11]. It is based on the ArgoUML CASE (Computer Aided Software Engineering). It is designed to assist users to specify their requirements, their requirements operations and their context without using contextual information. In fact, it provides interfaces for requirements specification using uses case diagram or other. The requirement specifications expressed by elearning domain users, are not always clear and require an analysis and reformulation phase. The problems of multiple contexts have been noticed through several examples shown in elearning. Context-aware systems need to detect the context of use and interpret it properly. The need for a contextual dimension in a very rigorous ontology dedicated to the RS is essential to detect the requirement context. Hence, we specify a set of parameters that helps to find a context. In the following, we define, first, the context parameters. Then, we present, our proposed approach to take into account the contextual dimension in the ontology dedicated to the RS. After that, we present our case study by modifying some features in the *ContextOntoMR* prototype to make it able to detect the context by knowing his parameters.

3. CONTEXT PARAMETERS

As we need to specify all the parameters necessary for our application, we opted for the definition of the context parameter as "information that helps to understand the user context and to exploit it later". This definition helped us to choose the context parameters specific to our application. Therefore, the parameters that we identified in our work [MTI 12] are the following. The first parameter is the User. It represents the user's physical and mental profile (name, function...). The second parameter is the Activity. It represents user's activities, goals or tasks that it develops. The Location is the third parameter and it indicates the geographical location of the user. The Time parameter, which is the fourth parameter, keeps a history of actions. It can be represented by the date and time of the system. The fifth and the last parameter is Physical environment which represents the devices, the network and the various types of equipment used by the user.

The reason behind this choice of parameter resides in the examination of certain sources of information which help to detect the context parameters. These information sources are the user context, the communication context and the platform context. The user information source includes two parameters: User and Location. First, the User determines user profile (administrator, tutor or student in the domain e-learning). Second, Location determines the university to which he belongs. Information source of the platform includes a single parameter called the Physical environment that contains the IP address parameter. For the

communication information source, two parameters are Time and Activity. Indeed, Activity exposes the user activity on the platform at a given time.

We notice that the parameters that we proposed are not fixed for all context-aware applications. They vary by application field. In the case of e-learning field, these sources of information well validate our choice of parameters. In another field, other parameters are chosen according to the relevant area of the context-aware application. Medical and GIS field are practical application for the chosen parameters. In general, in order to specify the context parameters, we must determine, first, the user context. The latter includes User and Location parameters. For instance, we show samples from Geographic information system (GIS) and medical fields. The car driver and the country (where the user exists) represent the example for the GIS field and the doctor and the hospital where the user worked represent the example for the medical field. After that, we determine the communication context. This context is represented by Activity and Time parameters. The User activity is the task that he performs by accessing the platform, such as login or authentication. The time parameter allows keeping the activity history. For the platform context, it is a physical environment parameter. The latter is specified according to the application, namely the IP address in the case of e-learning and the device type for displaying data in the case of the medical field and GIS field as well. For the two first information sources (user context and communication context), it seems that the parameters are appropriate for any application (HMI). For the third, regarding the platform context, it depends on whether the IP address, the terminal type or other parameters. Therefore, the parameters are not fixed and vary depending on the chosen field of application and the application needs as well.

4. CONTEXT DETECTION APPROACH

In order to detect the requirement context in ontology, we proposed an approach [MAA12], [MTI 12]. The importance of the latter appears in its ability to detect the context in a definite number of steps. This helps to achieve the next work, which is making a decision about adding a new requirement to ontology. Consequently, the creation of an approach helps to reduce the probability of errors and to gain a significant time profit. The ontology, that we use, is composed of requirements, their properties, context and context parameters. Two requirements are identical but they can be expressed in two different contexts. The approach that we proposed for the context detection, presented in figure 1, consists of four steps. The first step is a requirement and context parameters acquisition. It consists of acquiring the requirements expressed by the users for a given field in various forms of representations which can be textual representation or with use case diagrams.

CONTEXT DETECTION IN ONTOLOGY



Figure 1. Our context detection approach

In the second step, the requirement and the contextual parameters are processed. At the end of this step, we obtain a conversion of the expression of the requirements and acquired parameters in a common form and through the markup language XML. The comparison between the acquired parameters with those existing in ontology represents the third step. It allows comparing acquired requirements with the already existing requirements in domain ontology which forms the base for the ontology implementation. This comparison aims to decide if this concept is a new requirement or if it already exists in another form in order to insert it in ontology. The fourth and the last step has the objective to ensure the context detection and the decision of the ontology extension.

In the following sub-sections, we expose the various steps of this approach, in order to detect the context of a new requirement in ontology.

4.1 Parameters Acquisition

The first step consists of the requirements and context parameters acquisition for a given field. These requirements and parameters are expressed with a given representation. The representation is chosen by the user. He can indicate to textually represent its requirement and the context parameters. Otherwise, he can employ semi-formal techniques for the same objective.

We apply our approach in the e-learning field. A user, who can express his requirements, can be a "*Student*", a "*Tutor*" or an "*Administrator*". For instance, we can cite a requirement specification example for an actor "Tutor". The requirement "To subscribe" is expressed by two representations. The first describes a use case of the actor "*Tutor*". The latter wants "*To subscribe*" and needs to authenticate through the system. This requirement is expressed in a context having the following parameters: *User* is a *tutor*, *Activity* is *subscription*, *Localisation* is a *university of Sfax*, Academic year (*Time*) is "2009/2010" and *Physical Environment* is connected with IP Address "172.168.0.1". The second representation is textual, and underlines the various parameters related to the requirement "*To subscribe*", context. Let us quote the example of the requirement "*to subscribe*" which is expressed by a "*tutor*", registered at the "*University of Sfax*", its *activity* is "*subscription*", during the academic year (*Time*) "2008/2009" and with IP address "192.68.2.2".

4.2 Acquired Parameters Processing

To clarify certain valuable information, we need a processing step which is useful to be a base for concept extraction. This step allows converting the information expressed in various representations in only one common representation comprehensible by the machine to support the passage to the following phase of comparison between the acquired parameters and those which exist in the ontology. Let us take the example of the actor "tutor" in the e-learning field. After the "authentication", the "tutor" can "subscribe". The data processing allows clarifying certain useful information by determining the representation used to represent the requirement and the context parameters.

The XML is known as a powerful language that we need as an intermediate language to structure the information in a standard way. This is justified by the fact that XML is a structured meta-language. Therefore, it contributes to the exchange of information between various users. One of the strengths of this language resides in the documentary investigation. Thanks to its specific markup, it is possible to extract quickly and effectively relevant information.

For this reason, the specification will be transformed in a structured language like XML containing all information related to this requirement through the technique of markup. As an example, the markup <representation>..</representation > includes the representation with which the requirement and the parameters are expressed. With another textual representation, the same actor can express the requirement but according to another context and by using another requirement term. Another example that we can quote is that of the markup <parameter>..</parameter> which defines the specific context parameters for each requirement expressed in the textual specification. As a result of these two steps, we obtain an XML file containing the actor requirements according to different representations and contexts. Consequently, we have requirements defined in a pivot format which represents the input of the following step which is the comparison.

4.3 Comparison between Contextual Parameters of Ri Requirement and the M Contexts Parameters of the Existing Requirements in Ontology

This step aims to determine the relation between the new requirement and the requirements which already exist in ontology through a comparison between the parameters of the acquired context and the existing context parameters in ontology. This relation determines if there is an identity relation as a result of the comparison or if there are other types of relationships. We started by comparing between the acquired parameters and those already exist in the base. The major objective of this comparison is the resolution of the conflicts and similarity problems between the requirements expressed by the users at the requirements specification stage. By admitting that these RS are multi-contexts, we next propose a comparative study between the requirements for an ontology. We then propose a comparison algorithm between the requirements through the contexts and their parameters. Figure 2 exposes the proposed algorithm.



Figure 2. Comparison algorithm between the acquired context parameters and those already existing in the ontology

The possible cases for the comparison between contexts Ci are enumerated as follows. If the acquired *Parameters with New Requirement* (PNR) are similar to the *Parameters of an existing Context in Ontology* (PCO), the context of these parameters is known. It is the identity *Contextual Relationship* (CR) between this *Context* Ci and an existing *Context in Ontology* (CO). In the contrary case, if acquired parameters PNR do not resemble to any parameter i.e. any context of existing contexts in ontology (CO) is composed of these parameters PNR. In this case, a disjunction CR is found i.e. new context will be added to ontology with this CR. The last case arises when acquired parameters PNR partly resemble or cover with parameters constituting a context in the ontology i.e. with part of parameters PCO. In this case, we carry out a measurement of similarity where we calculate the *Similarity Coefficient* (SC).

Determining the degree of similarity between two concepts represents a problem in many applications: disambiguation, information extraction... A complete state of the art is presented by [PAT 03] where different similarity measures are compared regarding to an evaluation made by human subjects. The coefficient of similarity in our work is calculated by dividing the intersection of the number of parameters acquired PRi with the number of parameters PCO, by the PCO number of parameters (5 in our case).

 $SC = (PRi \cap PCO)/PCO$

If the similarity coefficient is higher than a threshold fixed by the expert, then we will insert this new context Ci with its parameters PNR in ontology. Consequently, an equivalence CR exists between these contexts. In the contrary case where CS is lower than the threshold, this context C_i is inserted with a part_of CR with other contexts.

4.4 Context detection

This step concerns the requirement context detection. After elaborating the comparison between the acquired parameters and those existing in the ontology, we can infer the context related to the parameters. Therefore, the objective of this step is revealed in the decision of inserting the new requirement to ontology or not. A new requirement which arrives at ontology must be identified by its context. This decision on the extension of ontology is made only after knowing if the context of the new requirement exists in the ontology. In this case, the requirement exists with its context in the ontology. In the context is not known as it does not exist in ontology. The requirement and its context will be added to the ontology.

When we add the requirement and its context, it is necessary to know which relations will be added to maintain the consistency and the coherence of ontology after the achievement of this step. These relations are described by relations between contexts of already existing requirements and context C_i of the requirement R_i lately arrived. If the context is not present in the base, we will add a new context C_i with acquired parameters P_{NR} under a new name. This name is composed by labeling of three parameters. This will facilitate the context addition operation.

The new context which will be added to the ontology must have a name. The best method, to name it, is by doing it automatically through the use of three context parameters which are only made up of letters: *user*, *location* and *activity*. We take the first three letters of each parameter to label the context. For instance, if the context is formed by these parameters: *student*, *rst_sfax*, *subscription*, then, the context name will be "stu_rst_sub", as formed by this labeling method. This method has the following advantages: automatic naming, facility, speed and autonomy of the application. The evolution of the ontology in a domain is an indicator of the importance of ontology in the modeling of user requirements in this area. Several issues arise from the evolution of ontologies: identifying needs for change, specification changes, application changes... [Dje 10], [Jaz 10]. The change that we can mention in our ontology dedicated to the RS is represented by the addition of new instances of concepts. These instances are related to the concepts "Context", "User", "Location", "Activity", "Time" and "physical environment" as well as relations between them.

5. OUR PROTOTYPE

The studied case of context integration in ontology is implemented through the *ContextOntoMR* prototype. The figure 3 illustrates the first widget of requirement specification. First, we choose to create a textual RS via choosing the "*Create new textual RS*" button. Second, we specify the context parameters. Through this step, the system actor enters his profile, his location, the activity that he will perform and the academic year. These parameters are acquired automatically via the interface. The fifth parameter which is related to physical environment and which is called IP Address is detected automatically. These parameters are then saved into an XMI file. This file contains not only the context parameters but others information that we don't need, this is why we use XQUERY. This latter enables us to get a formatted XML file with only the information that we need. The figure 4 shows this second widget of requirement specification.

CONTEXT DETECTION IN ONTOLOGY

Once the parameters are inputted, the application of JENA techniques comes then to operate on the acquired parameters from the interface. The figure 5 shows the context parameters tree on the left side. This window shows the user the result of the comparison from the parameters identified. An example of context "stu_rst_sub" that appears in the figure 6. It is detected by the inference mechanisms used and applied to parameters "Student", "rst", "subscription", "172.0.0.1", "2008/2009". The details of comparison are shown in the same figure.



Figure 3. ContextOntoMR interface



Figure 4. Context acquisition widget

	36 Textual KS		
Order By Type, Name	Pres Presentation Presentation - textual Presentation - textual Presentation - textual Presentation	Comparison Result The name of the new costoxt and its parameters completely coincide with a costext name and its parameters in ontology Result: Stop processing Detail Return	Checklist

Figure 5. Context detection widget



Figure 6. Context detection details widget

6. CONCLUSION

During this paper, we have demonstrated that the necessity to specify context parameters is entirely indispensable and observable, to better integrate the context in ontologies. Recognizing the context is a complicated activity due to the use of one requirement term into several utilizations. In fact, the requirement term can have different meaning according to its utilization context. Therefore, the parameters are important as they facilitate the task of detecting the requirement context which will give it its right sense. Besides, this paper convinces that the integration of the context in ontology is necessary to keep the context sharing and comprehensiveness as they present ontology advantages. Moreover, the use of the context allows giving the particularity to the instances according the utilization context. Toward this reason, we proposed an approach to support contextual dimension integration in ontology. The result of our approach is encouraging. The more we rely on the parameters, the more we have an idea of the correct and concrete context detected. However, the multitude of parameters affects the complexity of the ontology. We aim, in our further works, to optimize the comparison algorithm. We envisage also extending the prototype, by adding a feature, which can add a new requirement in the ontology, knowing its context. In addition, this feature will assist to clarify the relationship between the new requirement and other requirements that exist in the ontology. We intend, as well, in our future works to test and evaluate other parameters, in other fields.

REFERENCES

- Bouquet, P., Giunchiglia, F., Van, Harmelen F., Serafini, L. & Stuckenschmidt, H., 2003. C-OWL: Contextualizing Ontologies. International Semantic Web Conference, pp.164-179
- Brézillon P.: "Context and explanation in e-collaborative work", chapter in Handbook of Research on Methods and Techniques for Studying Virtual Communities: Paradigms and Phenomena., pp. 83-102, (IGI Global), (ISBN: 1-11111) (2011).
- Chen, H., Finin, T., and Joshi, A., 2003. An ontology for context aware pervasive computing environments. Special Issue on Ontologies for Distributed Systems, Knowledge Engineering Review, Vol. 18, Issue 03, pp. 197-207
- Decker, B., Rech, J., Ras, E., Klein, B., Hoecht, C., 2005. Selforganized Reuse of Software Engineering Knowledge supported by Semantic Wikis. In: Proc. of Workshop on Semantic Web Enabled Software Engineering.
- Djedidi R., Aufaure M.-A.: ONTO-EVOAL an Ontology Evolution Approach Guided by Pattern Modeling and Quality Evaluation. FoIKS 2010: 286-305
- Floréen, P., Przybilski, M., Nurmi, P., Koolwaaij, J., Tarlano, A., Wagner, M., Luther, M., Bataille, F., Boussard, M., Mrohs, B., Lau, S., 2005. Towards a Context Management Framework for MobiLife. In proceedings of the IST Mobile & Communications Summit, Dresden, pp. 19-23.
- Gruber, T., 1993. A translation approach to portable ontology specifications. Knowledge Acquisition, Vol. 5, pp. 199–220.
- Hammoudi Slimane: Model Driven Development of Context-aware Services using Parameterized Transformation. ICSOFT (2) 2011: 244-248
- Jaziri W., Sassi N., Gargouri F., 2010. Approach and tool to evolve ontology and maintain its coherence. In the *International Journal of Metadata, Semantics and Ontologies, Volume 5*, pp. 151-166.
- Lin, J., Fox, M. S.; Bilgic, T., 1996. A Requirement Ontology for Engineering Design. Enterprise Integration Laboratory, University of Toronto, Manuscript.
- Mayank, V., Kositsyna, N., Austin, M., 2004. Requirements Engineering and the Semantic Web, Part II. Representation, Management, and Validation of Requirements and System-Level Architectures. Technical Report. TR 2004-14, University of Maryland.
- Maalej M., Mtibaa A., Gargouri F. "Context parameters in ontology", The 2nd International Conference on Information Systems and Technologies (ICIST'12), ISBN 978-9938-9511-2, 7, March 24-26 2012, Sousse, Tunisia, pp 447-452.
- Mtibaa. A., 2011. Une ontologie de multi-représentation pour la spécification des besoins multi-contextes. Ph.D. thesis, MIRACL Laboratory, University of Sfax, Tunisia.
- Mtibaa A, Maalej M. and F. Gargouri, "Contextual dimension in ontology dedicated to requirement specification", IADIS International Conference Information Systems, Berlin, Germany, 10-12, March 2012.
- Patwardham S., 2003. Incorporating Dictionary and Corpus Information in a Measure of Semantic Relatedness. M.S. Ph.D Thesis, University of Minnesota, Duluth, MN.
- Wouters, B., Deridder, D., Van Paesschen, E., 2000. The Use of Ontologies as a Backbone for Use Case Management. In: *European Conference on Object-Oriented Programming*, *Workshop : Objects and Classifications, a natural convergence.*