Towards Ontological Model Accuracy's Scalability: Application to the Pervasive Computer Supported Collaborative Work

Kahina Hamadache
Orange Labs
Caen, France
kahina.hamadache@orange-ftgroup.com

Luigi Lancieri
University of Lille
Lille, France
luigi.lancieri@univ-lille1.fr

Abstract—**In this paper we define an ontological model to accurately represent the context in Pervasive Computer Supported Collaborative Work. A major issue in this domain is the mass of information required to correctly depict a situation. As we need to represent users and devices according to multiple aspects (physical, computational, social …) the amount of information can quickly become unmanageable. Besides, as a PCSCW context model has to be usable on limited resources devices such as cell phones, GPS, ADSL Modems we needed a more efficient way to represent information. In this perspective the model we propose offers the possibility to represent a situation with more or less precision; that is to say with more or less abstraction. The final goal of this work is then to provide a model able to reason with a precise or fuzzy description of a situation.**

Keywords-component; ontology; ontology reducing; pervasive computer supported collaborative work; swrl rewriting

I. INTRODUCTION

Socialization of computer systems and technological progresses of the recent years combined with the increasing need for efficient collaborative softwares have led to the development of a branch of study known as Computer Supported Collaborative Work (CSCW). This research field covers an extremely wide range of disciplines from computer science, until education, via psychology, anthropology, sociology, ergonomics, etc. Applications that fall under the CSCW umbrella are diverse: electronic mail, newsgroups, chat, multi-user editors, meeting support, videoconferencing, shared simulations, and workflow are some examples.

Modern Computer Supported Collaborative Work systems integrate almost all classical features required for what we can call “traditional collaboration”. But some research are going beyond these classical consideration, they want to raise collaboration to unprecedented and unexpected levels to provide new tools, features and behaviors. Among these researchers, many of them are working hard on a better management of users’ context. This can be done at different steps of the human-machine interaction. For example some researches focus on the context acquisition, the different manners to collect context data by the use of hardware and software detectors. Others concentrate on the context representation by the development of context models. In an upper step, people are trying to infer information basing their inference rules and models on previously developed context representation. With inferred data, some of these scientists have designed context awareness mechanisms, allowing the adaptation of applications and systems to users’ context.

The context concept has many meanings depending on the area it comes from [10], [11], there is no complete consensual definition about what is context. Several domains have already elaborated their own working definition of context.

In the area of Computer Supported Collaborative Work, Rittenbruch [12] considers that context is a complex description of shared knowledge about physical, social, historical, or other circumstances within which an action or an event occurs. Context may be seen as a dynamic construction composed of five dimensions: time, usage episodes, social interactions, internal goals, and local influences [2]. Although contextual elements in some situations are stable, understandable and predictable, this is not always true. Mäntyjärvi et al [3] describe *collaborative context* as the "summary of the situation of the other devices in the local range corrected by the local context" providing an update strategy for these devices and associated trigger conditions. Wang et al [8] define a context as any information that can be used to characterize the situation of entities in the collaborative space.

What we want to do in this paper is to provide a base to develop ontologies able to adapt their representation and reasoning according to the limitations of the current device.

The content of this paper is organized as follows: in section two we will introduce the basic concepts of the PCSCW Model. In the third section we present our approach to reduce and limit the size of ontologies and reasoning rules. The fourth section provides a simple use case with OWL and SWRL. The fifth section proposes a review of the literature in the collaborative context modeling domain. Finally we outline some perspectives and discuss of our proposition.

II. PCSCW MODEL

The As our research interest have focused on the integration of the pervasive computing aspect in the computer supported collaborative work, we have proposed in a previous work [14] an original model which aims at making smart devices cooperate seamlessly to improve and facilitate the collaboration of users. This model, named PCSCW for Pervasive Computing Supported Collaborative Work, relies on some simple but essential "sub-models":

- A *Task Model* composed of mainly two concepts:
    - *Task*: represents a meaningful process to be perform by one or more users to achieve a specific goal, for instance "creating a webpage", can be composed of a set of subtasks or actions;
    - *Action*: describes an atomic step of a task, it has no discriminatory meaning as it can't be understood outside of a task. To illustrate it we can consider the action "opening a web browser" that doesn't convey any specific meaning but can be integrated in tasks such as "searching the web" or "checking mails".
- A *Role Model* built above the task model, it extends it by providing one more concept and some refinements about tasks:
    - *Role*: it defines a role to be played by one or more users by wrapping tasks into subsets: mandatory, allowed and forbidden tasks;
    - *Tasks* can require one or more roles to be performed. Thus a single task may be shared among several roles and then becomes a "Collaborative Task".
- A *Resource Model* providing a common ground to represent:
    - *Required resources*: the set of resources an action requires to be performed. By describing these requirements in term of software, hardware, human and social resources at the action level we can efficiently describe resources required for a given task;
    - *Available resources*: the set of resources available in user's environment, it provides a structured representation of the context;
    - *Device*: the representation of a contextual device is merely a part of available resources but with a particular extent as it is considered as an active agent of the collaboration.

In addition to these sub-models the PCSCW Model includes *Device Collaboration Rules* [15]. The main principle of these rules is the following: by comparison of resources required to perform a task or an action with available device resources we can trigger specific interactions between devices to make them cooperate to finally provide all required resources to the user. Going a little further these rules can even perform whole actions or tasks and prevent users from doing repetitive and thoughtless ones.

All these features create a model allowing smart devices of users' context to automatically and seamlessly cooperate to facilitate, channel and enhance the collaboration of users.

III. LIMITING THE SIZE OF ONTOLOGIES

To describe and represent the context in the Pervasive Computer Supported Collaborative work, we decided to use ontologies.

As we have already evoked our main purpose is to find a way to reduce the weight of the context representation to be able to reason efficiently and quickly even on small devices. Even though we need to reduce the size of our ontology it is not possible to simply "forget" a part of the context. Indeed, you can't know when you will have to use this or this information. Thus, keeping in mind the work that has been done by [13], we tried to organize pervasive computing context in multiple levels. Doing so we rapidly came to the conclusion that a great amount of information is redundant in an ontology.

Our main idea to reduce the size of such ontology is to represent existence dependencies between classes, that is to say: which class requires an instance for the existence of an instance of another class. Again, let's illustrate this principle by a simple example on Figure 1.

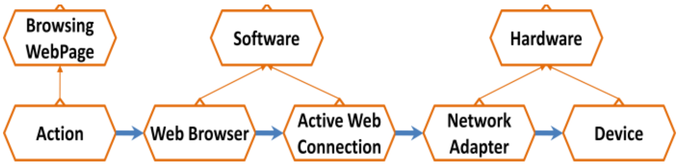

Figure 1. Information Dependency

On the precedent figure we have represented some of the implication induced by the existence of the action "Browsing WebPage". Indeed, the simple fact of knowing that a user is browsing a webpage has a set of direct and indirect involvements: firstly if the user is browsing a webpage it implies that he is using a web browser; this web browser has to be supported by an active web connection which itself requires a network adapter embedded in a device. These cascading requirements are a simple but eloquent example of information redundancy. Indeed, if you only need the "web browsing" action you will have created the representation of, at least, three useless resources.

To help defining what instances can be "obscured" we propose to organize resource representation in layers. As depicted by Figure 2 on the top layer we've got containers that are absolutely required for the existence of sub layers elements. On the second layer we've got the main components embedded in a container. These components provide functionalities represented on the third layer. For each functionality we've got a set of associated data that forms the fourth layer. In a convenient way we can have several "levels" of components or functionalities according to the accuracy of the representation.

On the Figure 3 we have depicted how some of the context information can be represented and how they interact inside the PCSCW Model. On the left side of the figure we've got the representation of the hardware and software aspect of a device. On the right side we've got the representation of the task model and how it can drive the interaction between users. Both of the sub-models are organized according the previously evoked layers that represent successive levels of context abstraction.

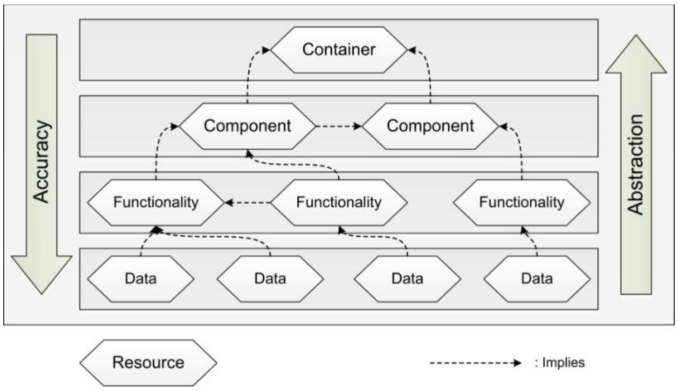

Figure 2. Layers and Dependencies

For the resource description part we can identify four main levels:

1) The root level, corresponding to the device itself (or the device body) for the hardware part and to the operating system (we could even have gone down to the bios) for the software part;

2) The second level corresponds to main components: applications and services for the software aspect and peripherals (integrated or not) for the hardware part;

3) The third level depicts functionalities offered by each components of the precedent level. For instance a web browser can offer various functionalities to browse the web, history saving, favorite pages management, forum monitoring, etc; while a network adapter can support different standards of Wifi;

4) The last level of the resource model is the representation of data. It obviously relies on the third level as it is the expression of the current value of functionalities. For instance a network adapter can be connected or not to the network.

The task part of the PCSCW model is also organized in layers:

1. The top-level layer represents users taking part in the collaboration;
2. The second layer is a representation of their associated roles;
3. The third layer is composed of the tasks contained by each role;
4. The fourth layer represents actions required by tasks;
5. The fifth layer is probably the most interesting as it depicts the resources required by the actions to be realized.

Thus if we refer to the main principle behind the PCSCW model, devices have to decide how they have to cooperate to properly channel the collaboration between users. This is done by a comparison between resources required to perform an action and resources available in the current state of the environment. Thus optimally we have to map each required resource to an available one. In the perspective of reducing the size of the context ontology the layers we have defined can help us automatically infer information according to the deepest level of abstraction.

As we evoked previously, knowing that a user is performing the action "browsing the web" implies the existence of several resources (web browser, web connection, network adapter...). Then in an optimistic view we could say that the action surpasses the description of resources and that we can avoid to represent all implicit resources. We think this assumption is only partially right, indeed if we keep on considering our web browsing example it can be problematic if we completely remove the network adapter representation from the current ontology. Indeed even if the "web browsing action" can be sufficient for a part of the reasoning made on context knowledge, it can be insufficient if you have to deal with more specific mechanisms requiring the precise value of connection setting and functionalities. For instance if current user's task involves an action requiring an high speed bandwidth (such as video conferencing), knowing that he's already surfing the Internet doesn't bring sufficient information to determine if the network connection suits the requirements.

To solve this problem we propose a simple mechanism based on the analysis of reasoning rules. It can be summarized as following: "If a resource A implies the existence of a resource B and no reasoning rule contain a condition involving a data of B uncharacterized by A; then all references to B in reasoning rules can be replaced by A". This mechanism ensures that the resource B can be replaced by the resource without losing reasoning capabilities.

In the next section we'll present some details of the potential use of our work with OWL and SWRL.

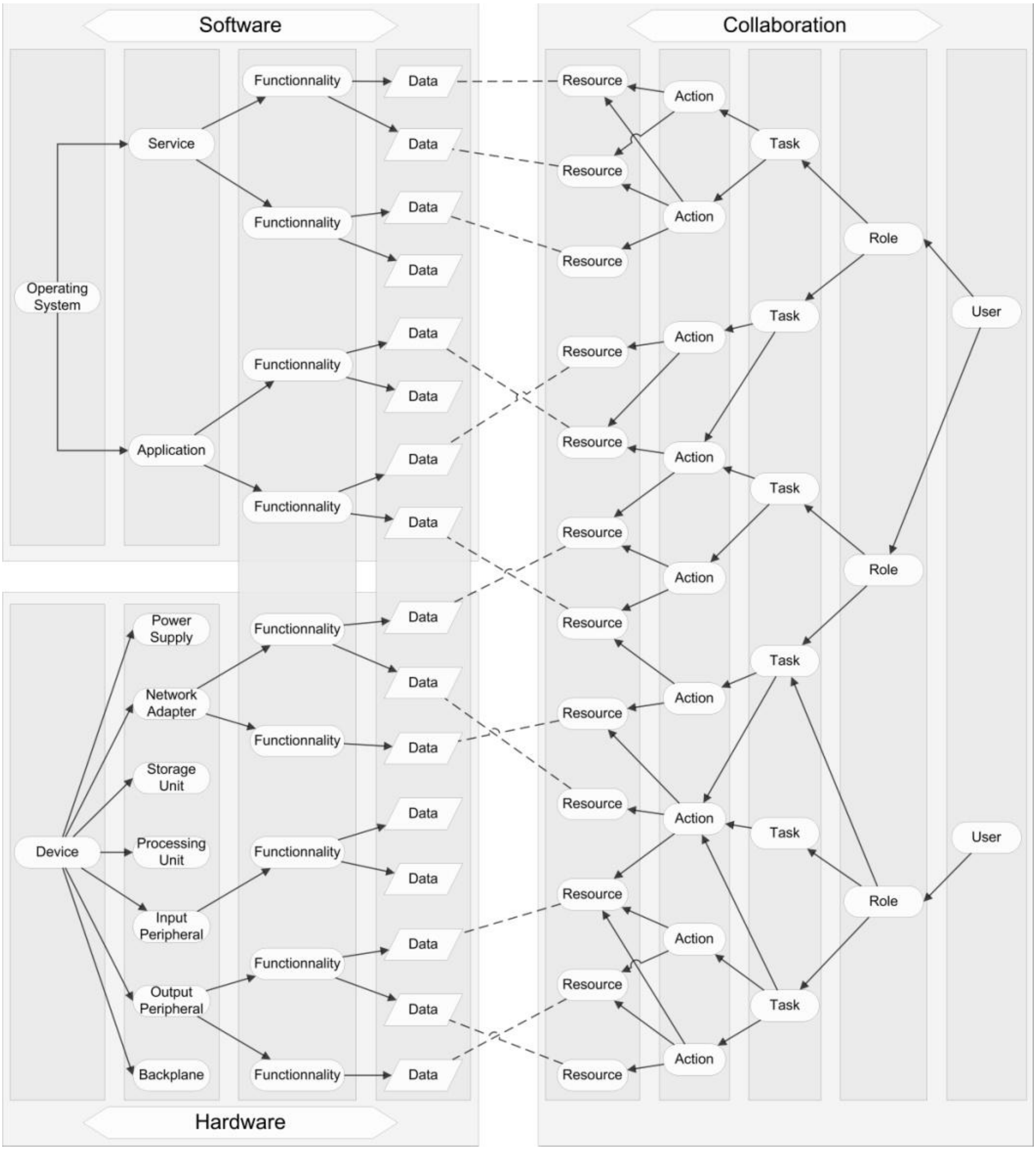

Figure 3. Resources Layers and the PCSCW Model

IV. USE CASE WITH OWL AND SWRL

Our work in the Pervasive Computing and collaboration domains has led us to the use of ontologies, to develop them we rely on the use of the OWL and SWRL languages. OWL is used to represent the ontology itself while SWRL allows us to represent reasoning rules. As they are both commonly used languages we will not give further details about them. Considering figure 1 the concepts “Browsing WebPage”, “Web Browser”, “Active Web Connection”, “Network Adapter” and “Device” are represented with OWL classes, while implications links (“Implies”) are represented with OWL Object Properties. Obviously this example doesn’t even try to show the real complexity of the representation of an ontology with OWL, but it is sufficient for our use case.

On the following SWRL Rule we have defined a quite simple behaviour for a device, it simply depicts the fact that if the device has a network adapter, an associated active network connection, a software running on an operating system and requiring an update, the device can start the update of the software.

$$\begin{gathered} Device(?d) \land NetworkAdapter(?na) \land NetworkConnection(?nc) \\ \land supportedBy(?nc, ?na) \land active(?nc, true) \\ \land OperatingSystem(?os) \land Software(?s) \land runs(?os, ?s) \\ \land needUpdate(?s, true) \\ \rightarrow UpdateSoftware(?s) \end{gathered}$$

If we refer to what we said earlier the action “WebBrowsing” can replace a large part of the components of this rule. Then given the fact that the user is currently browsing the web, we can dramatically reduce the size of this rule by rewriting it as following:

$$\begin{gathered} WebBrowsing(?wb) \land Software(?s) \land needUpdate(?s, true) \\ \rightarrow UpdateSoftware(?s) \end{gathered}$$

One of the main benefits of this reducing, apart from its better readability, is the reduction of the required reasoning time to run this rule. Indeed, instead of having nine (antecedent) atoms to evaluate, we only have three. On a single rule of a small ontology it may not be such an improvement, but if you consider a large ontology with hundreds of concepts and numerous rules, the possibility to avoid the representation of at least a small part of the concepts and to reduce the size of rules can doubtlessly make you gain a precious amount of time.

V. RELATED WORK

There are, in the Computer Supported Collaborative Work literature, several propositions and studies on how context can be modeled and managed in collaborative applications, Leiva-Lobos and Covarrubias [4] propose that the context where cooperating users are situated is tripartite: spatial, temporal and cultural. The spatial context addresses artefacts populating physical or electronic space, temporal context refers to the history of performed cooperative processes and to the expected future one. The cultural context gathers users' shared view and practices.

Kirsch-Pinheiro et al [5] propose a context based awareness mechanism which filters the information delivered to the user according to a context description. This concept description takes into account the concepts related to the notion of awareness (user, group role, location, etc.). They represent this notion of context through an object-oriented representation, using the UML notation. The previous concepts become classes and the relationships among them, associations. Their representation starts by the definition of a context description class, which is composed of a set of basic elements and is defined for a user that is currently accessing the groupware system

Vieira et al [7] propose an ontology for context representation in groupware systems. The context information is classified in three main categories: *Physical:* contains information about physical elements that characterise the situation of a user in a specific time; *LocationContext*: proximity, distance, presence, absence. *TimeContext*: time when the interaction occurs, time zone. *DeviceContext*: physical and electronic devices available, such as printers, computers, microphones, webcams. *ConditionContext*: physical conditions such as temperature. *Organizational* (similar to [6], [5]): represents the contextual information related to the whole structure that identifies the user. *Interaction context*: identifies the information related to the context of an interaction that is happening (Synchronous) or that has happened (asynchronous) during group work. It is divided into subclasses: *SharedArtifacts-Context*: contains elements related to the context of the shared artefacts used in the interaction and *ApplicationContext*: includes to the context of the application being used in the interaction.

Wang et al [8] propose a semantic context model for collaborative environment (Ontology for Contextual Collaborative Application OCCA), they classify the contextual information into eight categories: *person context*, *task context*, *artefact context*, *tool context*, *collaboration control context*, *environment context* and *historical context*.

Hu et al [9] present a semantic context model supporting location based contextual and cooperative mobile learning. The following context related to a mobile contextual content service: Personal context, Task context, Role context, Spatio-temporal context.

Truong et al [1] propose a flexible context model; they utilize existing ontologies (FOAF, vCARD, Basic Geo, vCal, ResumeRDF, Time ontolgy) and they also developed five new core ontologies: Location, Activity, Team, Resource and Action.

In [13] the authors propose a method that particularly drew our attention. In that paper they introduce the notion of “multilevel ontology”. It refers to the necessity in certain domains to represent information in multiple layers due to their inherent complexity and the fact that an action on a “lower” layer can impact top level concepts. In their specific case of study they had to model interactions in the biological domain, for instance the fact that the heart is composed of mainly three kinds of tissues : epithelial, connective and nervous, which are themselves composed of specialized cells, composed of molecules. Thus specific molecules can alter the behavior of cells, conducting to the degeneration of tissues and the failure of organs.

This work is in some perspective similar to what we wanted to achieve. Indeed even if our work focuses on a different domain we also needed to represent multiple levels of details and abstraction and the relationships existing between these levels.

VI. DISCUSSION AND PERSPECTIVES

In this paper we have presented a method to reduce and limit the size of ontologies by describing existence requirements and controlling the usefulness of resources according to reasoning rules.

We think this work is particularly relevant in the case of devices with low resources that require to use small ontologies, not fundamentally for their size, but to save reasoning time (and then computing resources). One of the main benefits is the fact that even if a device has a limited description of its current context it may be able to reason as if it had a complete description of it.

In addition this work not only helps limiting the size of ontology, but it can also help the device deciding what information is relevant for its reasoning. In the very same idea the “ontological architecture” we propose could be used to drive the search for context information. Indeed, instead of using unsupervised context information collector, we could use smarter collectors directly driven by the requirement of reasoning rules. Then by having identified the existence requirements we can fasten the context information collection by precisely driving collectors. In the example we proposed in this paper, a context collector would simply had to look for the web browsing action instead of having to search for information about the device, the network adapter and the network connection. Once again this point is particularly interesting in the case of low resource devices, but it’s also relevant in the case of an environment “reluctant” to provide information. Indeed, a device is not always allowed to access all information, even concerning itself.

The main problem with the reduction of ontology size is obviously the risk to lose a useful part of context information. This issue has to be considered seriously and any further progress in the perspective of reducing the size of knowledge representation has to keep in focus this essential need.

As for now we know that our work is only a first step toward the real scalability of knowledge representation inside ontologies; indeed we already know that we’ll have to pursue our efforts to validate our proposition and to explore other ways to reduce the size of ontologies and the complexity of reasoning rules. Our thought remains the same as it was when we started this work, by representing more and more accurate resources and actions we should be able to limit the growth of working ontologies. Indeed, even if this accuracy implies a bit more description of represented resources, it can save the description of numerous less accurate related but unnecessary resources.

Last point we’d like evoke here is the opportunity given by the PCSCW model to help us reducing the size of the ontology. Indeed this model requires to precisely defining actions of users’ roles, thus it makes it possible to summarize a large part of the current context with current user’ actions.

REFERENCES

[1] Truong, H., Dustdar, S., Baggio, D., Corlosquet, S., Dorn, C., Giuliani, G., Gombotz, R., Hong, Y., Kendal, P., Melchiore, C., Moetzky, S., Peray, S., Polleres, A., Reiff-Marganiec, S., Schall, D., Stringa, S., Tilly, M., Yu, H.: inContext: a Pervasive and Collaborative Working Environment for Emerging Team Forms. In: IEEE International Symposium on Applications and the Internet, pp. 118--125. IEEE Press, (2008)

[2] Greenberg, S., "Context as a dynamic construct", Human-Computer Interaction, Vol. 16, No. 2-4, 2001, pp. 257-268.

[3] Mäntyjärvi, J., Himberg, J.,Huuskonen, P. "collaborative context recognition for handheld devices". In PERCOM'03: Proceeding of the first IEEE International Conference on Pervasive Computing and communications, USA, IEEE Computer Society (2003) 161.

[4] Leiva-Lobos.E.P, Covarrubias.E, "the 3-ontology: a framework to place cooperative awareness". In Haake, J.M, Pino, J.A (eds), CRIWG 2002, LNCS 2440. Springer-Verlag, (2002) 189-199.

[5] Kirsch-Pinheiro.M, Gensel.J, Martin. H, "Representing context for an adaptative awareness mechanism". In proc of the International Workshop on Groupware (CRIWG'2004), LNCS 3198, San Carlos, Costa Rica, Springer-Verlag (2004).

[6] R.A. Alarcon, L.A. Guerrero, S.F. Ochoa and J.A. Pino, "Context in collaborative mobile scenarios", the fifth International and Interdisciplinary conference on modelling and using context, Workshop on context and groupware, CEUR Proceedings, France, July, 2005.

[7] V. Vieira, P. Tedesco and A. C. Salgado, "Towards an ontology for context representation in groupware", Proceedings of 11th International Workshop on Groupware, Porto de Galinhas, Brazil, Sept. 25-29, 2005. Lecture Notes in Computer Science, Vol. 3706, pp.367-375.

[8] Guiling Wang, Jinlei Jiang and Meilin Shi, "a context model for collaborative environment". In Proceedings of the 10th CSCWD, 2006.

[9] Chang Hu, Qiong Liu, Xuemin Liu, Chunyuan liao, and Paul McEvoy " POEMS: a paper based meeting service management tool", 2007.

[10] P. Brézillon and J-ch. Pomerol, "Contextual knowledge sharing and cooperation in intelligent assistant systems", le travail Humain 62 (3), PUF, Paris, 1999, 223-246.

[11] T.P. Moran and P. Dourish, "Context-aware computing", Human-Computer Interaction, 16(2-4), 2001, 87-94.

[12] M. Rittenbruch, "ATMOSPHERE: a framework for contextual awareness", International Journal of Human-Computer Interaction, Vol.14, No.2, 2002, pp. 159-180.

[13] R. Elmasri, J. Fu and F. Li, "Multi-level Conceptual Modeling for Biomedical Data and Ontologies Integration" CBMS ‘07. Twentieth IEEE International Symposium on Computer-Based Medical Systems.Volume , Issue , 20-22 June 2007 Pages:589 - 594.

[14] Hamadache, K and Lancieri, L “Role-Based collaboration extended to Pervasive Computing”, Proceedings of the International Conference on Intelligent Networking and Collaborative Systems, Barcelona, Spain, Nov. 04-06, (2009)

[15] Hamadache, K., Lancieri, L.: Dealing with device collaboration rules for the PCSCW model. In: 16th Collaboration Researchers' International Working Groupware Conference on Collaboration and Technology (CRIWG 2010), September, Maastricht, the Netherlands (2010) (in press)