MeCoCo: A Context-Aware System for Mediated Communications

S. George and A.S Lekira

Université de Lyon, Villeurbanne, France

Abstract—In this paper, the potential of contextual communications with mobile technologies is explored. Context-aware applications allow relevant information and services to be delivered to users at a given time. One of the two objectives of our work is to propose a generic user's context model for mediated communications. This model is based on six dimensions: User's profile, Activity, Device, Environment, Location and Time. In addition, to show the utility and to validate the proposed approach, we designed a prototype named MeCoCo (*Mediated Contextual Communications*) with a specific interface promoting the search of interesting contacts and messages. The MeCoCo prototype illustrates the power of contextual information.

Index Terms—Computer-Mediated Communication, Context-Aware Computing, Contextual Communication, User's Context.

I. INTRODUCTION AND RESEARCH ISSUES

With the development of mobile and ubiquitous computing, contextual information is becoming more and more frequent. Our work deals with context-aware computing i.e. systems that adapt to the users' context. In particular, we are interested in systems which link mediated discussions to documents or situations. Thus, depending on their activities, their locations or on other parameters which we develop later in the paper, users will have the opportunity to see discussions (synchronous or asynchronous) related to their contexts at a given time. Users will also be able to interact directly with people who have a close context.

Certain research works, concerned with the adaptation of applications to the user's context, define context as the following triplet : <user, platform, environment> [1].. In the field of HCI (Human Computer Interaction) generic adaptations are made at the interaction levels between users and applications. In our work, we focus on a particular adaptation: mediated interactions between people.

The potential of contextual communication applications has been proven in previous research [2]. In this work, contextual forums (based on educational scenario or on knowledge) are designed to link communication to learning activities. In the current work, we aim at taking advantage of mobility to contextualize communication. Thus, depending on their location and their profile, users will have access to different discussions. For instance, a student who visits an archaeological site could have access to reviews written by scientists about the site and could post messages (by asking questions or leaving comments) to the community or to a particular person. Communications can be synchronous (chat or phone) or asynchronous (fo-

rum or email). On the other hand, a tourist in the exact same location would have access to wide public material such as simple reviews, videos and discussions.

Another example could be given in a professional context: a worker may be in a new situation (e.g. using a new type of material) and have a synchronous video communication with an expert who knows how to use the material. In this case, the goal is to have a just-in-time and contextual learning.

In these three examples, the goal is to have a dynamic information filtering according to the user's context. The information can be either discussions or people profiles.

Our work is broken down into two fundamental issues:

- How to define a generic model of user's context for computer-mediated communication in mobility situations?
- How to use different dimensions of a context to offer a simple and adaptable interface to suggest potentially interesting messages and people for a particular user?

The paper is organized as follows: section 2 presents related research works, section 3 describes in detail the proposed generic users' context model for mediated contextual communication. The architecture and the prototype of our system called MeCoCo is developed in section 4. Finally, we discuss our proposal in section 5 by providing a comparison of our approach with existing research before concluding in section 6.

II. RELATED WORK

Various projects have been conducted in the field of context-aware computing. We have selected those which are most relevant to our research.

Calvary's team [1] established a reference frame in the HCI field for context-aware computing. The user's context is broken down into three parts: the end-user of an interactive system, the device the user interacts with and the physical environment around the user. The context is thus defined as the triplet <user, device, environment> where:

- The user is the person who uses the interactive system and is described by a set of values that characterize the user's capabilities, cognition and actions in order to choose the best methods for manipulating the interactive system.
- The device includes hardware and software resources. The device model determines how the information is calculated, transmitted, rendered and

manipulated by the user. It includes memory size, bandwidth and device interactions.

- The environment is a collection of objects, people or events that are peripheral to the activity but may have an impact on the system or on the users' behavior. The environment includes information on the users practice as well as considerations for technical constraints. For example, the ambient light could be important when it affects a video communication system.

This definition of context is quite general. In the next part, we will therefore describe some research works on practical applications. These works are based on various implementations of context models that we will discuss.

Ranganathan and Campbell designed a context model based on the first order predicate [3] and built a sensitive context system based on this model, ConChat [4], which is used in a pervasive work environment called the "intelligent room". With this system, users are aware of what their colleagues are doing and what is happening in their immediate environment during a conversation. Different contextual information are used such as team location, number of non-team members in the room, identities of these people, users' mood (happy, sad, irritated...), user's status (at lunch, phoning, sleeping…) as well as the level of light and noise in the room and the main activity in the room (meeting, reading, coffee break…).

For example, if user A knows that user B is talking with someone else or is involved in an activity requiring full attention, A can expect B to not respond quickly or not at all. However, if A is aware that B is currently attending a meeting directly related to their work, A knows that B will be able to respond almost immediately.

Kirsch-Pinheiro *et al.* designed an object-oriented context model [5] which takes into account two contextual aspects [6]: physical aspect (location, device, application) and organizational aspect (group, role, calendar, activity, objects and processes shared). The context filtering process is based on general profiles that describe the users' current context and preferences. Filtering rules reflect the user's preferences considering the context associated to the general profile (i.e. the information users would like to have when they are in a given context).

Wang *et al.* suggest a context model based on ontologies [7]. They designed CONON (Context Ontology) based on OWL (web ontology language) to model the context in a pervasive environment. CONON provides an upper ontology with general concepts as a basis to define context and also to provide opportunities for expansion by adding specific domain ontology in a hierarchical way. Logical reasoning mechanisms are used to verify the context consistency but also to deduce a high level implicit context from low level explicit context. For example, a mobile phone can adapt its behavior depending on its context of use. If the user is sleeping, calls will be transferred to the voice mailbox. If the user is cooking or watching TV in the living room, the ringing volume will increase, on the contrary if the user is having dinner with people in the dining room, the phone will be set on vibrating mode.

In addition, in the few past years, other ontology-based context models have been established: CC / PP (Composite Capabilities / Preferences Profile) [8] created by W3C and UAProf (User Agent Profile) [9] created by Open Mobile Alliance. These two approaches are based on RDF (Resource Description Framework) to describe contextual information related to mobile devices. The CoBrA (Context Broker Architecture) [10] system uses a model based on OWL to acquire, share and reason on contextual information. CoOL (Context Ontology Language) [11] is another example of a system using ontologies to represent contextual information.

The system architecture proposed by Basaeed *et al.* [12] is dedicated to mobile-learning (m-learning) and is focused on the contextualization of resources (learning objects, presentation and navigation). This architecture is suited for m-learning applications but does not aim at supporting the creation of communication between people.

Only a few m-learning architectures have been developed to facilitate communication among users. This is one of the observations made by Laine and Joy [13] in their survey on context-aware pervasive learning environments: "*most of the systems supported multiple simultaneous users, but few facilitated virtual communication*".

If we focus on systems dedicated to putting people in contact with each other, we can cite:

- CybreMinder [14] which allows users to associate contextual information with "to do" items,
- AwareNex [15] which displays information on the location and activity of people in the user contacts list,
- Smart Instant Messenger [16] which proposes a unified interface for all communications among humans, software services and devices.

Despite the efforts made, we noticed that hardly any research focuses on modeling user's context to favor communications between people in mobility. In our work, the idea is to develop a generic users' context model for mediated communication. We present our work as a generic model that future designers can use by adding their own representation needs. The proposed model is not generic for all types of applications, it is rather specific to a particular field: mediated communications between people.

III. GENERIC USERS' CONTEXT MODEL

The notion of context is crucial in adaptive and sensitive systems. According to one of the most widely accepted definitions, 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 the applications themselves*" [17].

This definition does not focus only on the user, but it also takes into account other entities such as location, devices, etc. We can draw an immediate consequence: the multidimensionality aspect of the context is essential. This analysis leads us to define the users' context through six key dimensions: user's profile, activity, device, environment, location and time. Thus, we choose to represent the user's context as a sextuplet:

< User's Profile, Activity, Device, Environment, Location, Time >

and a context *i* is defined as the following:

$$C_i = < U_i, A_i, D_i, E_i, L_i, T_i >$$

Each of these dimensions is important in our model in order to define the users' context in a comprehensive manner and to provide them with the most relevant dis-

cussions or comments according to their current situation. The model is illustrated in Fig. 1 where each dimension is represented by a rectangle (activity model, location model, device model, etc.)..

In the next part, we will briefly present each of these dimensions:

Users' Profile. The information about the user is fundamental and central in our model. The quality of the definition of users' profile will influence the information which will be provided. For instance, during the visit of an architectural site, the information and comments presented to an expert in this field (an architect in our case) or to a neophyte (a tourist for example) will be different. Thus, in the proposed model, the user's dimension takes into account personal and professional information, like general profile (name, age, job, interests, etc.), preferences, disabilities, culture, etc. This list is not exhaustive and may be completed in accordance with the needs of a particular application. We point out that taking into account possible user disabilities is very important in order to adapt the information display. Adaptivity may also be based on the user culture, preferences or particular likings.

Activity. Information on users' task or activity describes what the users are doing and what their goals and availability are. This last information is especially crucial during the contextualization of communication between users. For instance, users who are "in a meeting" activity might not respond if other users want to communicate with them synchronously. This dimension is organized and represented through an activity model which is specific to a given area. For example, Bouzeghoub *et al.* [18] proposed an activity model based on educational activities (learn, review, read, write, solve, study, discuss, collaborate, etc.).

Device. The devices used affect the user's contexts. For instance, the means of communications will be different for users using devices with a small screen (PDA, smartphone, etc..,) and users working on a desktop computer. This dimension is organized and represented through a device model which takes into account the following information:

- Type of device (example: PDA, Smartphone, desktop computer, etc.)
- Possible connections (example: ADSL, WIFI, Bluetooth, etc.)
- Connection speed (example: 100 Mbps, etc.)
- Memory space
- Screen size
- Type of input/output (example: keyboard, touch screen, microphone, headset, etc.)
- …

Time. Time dimension gives a temporality to a context. For instance, it could mark the beginning and the end of a particular context (a time interval). Time information is linked to activity dimension so as to specify temporal and dynamic aspects.

Location. This dimension indicates the location of users and their relative position to one another . The model of localization can provide the following information:

- User logical location (an URL for example).
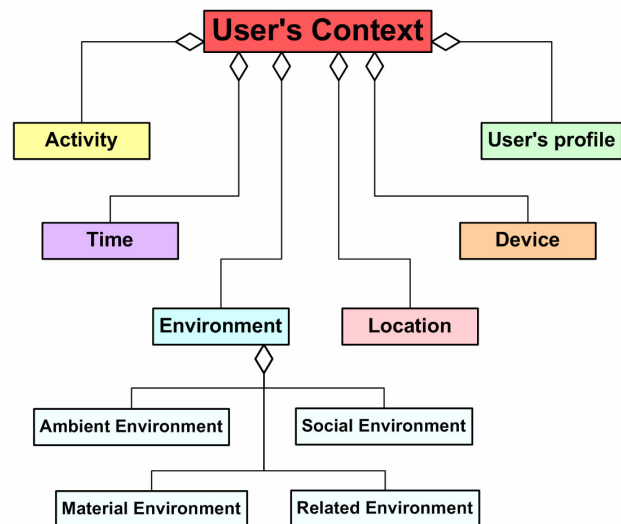- User physical location (example : Paris, Eiffel Tower with GPS coordinates, …)

Figure 1. The proposed users' context model

Environment. The surrounding environment is part of the context because when users perform a task or activity, they may act on and interact with the environment around. A definition of environment is given in [19]: « *An environment covers the set of objects, persons and events that are peripheral to the current task(s) but that may have an impact on the system and/or the user's behavior, either now or in the future* ».

In the proposed model, environment includes four distinct parts:

- *Social environment* which lists people who are physically or logically close to the current user.
- *Technical environment* which lists all electronic and computer devices physically close to the user.
- *Related environment* which gives an overview of places near the user's current location. These can be logical or physical places.
- *Ambient environment* gives information on temperature, noise, luminance, etc. This part is useful because it can limit certain interactions. For example, if the ambient noise is too loud, calling functionality will not be recommended.

Furthermore, the six dimensions of the context are linked together through semantic relationships. For instance, a device is located at a given location, several users' profiles are involved in an activity, and so on. It is possible to make a complete graph and to link the six context dimensions together according to the needs of the considered application.

The notion of event takes its importance in our model to define a change of context. Indeed, to move from a context C1 to a context C2, an event must occur. We define an event as a significant change in the current context, more precisely, a change or a modification of one or more dimensions that build the context. For example, during the visit of a museum, moving from one room to another leads to a change of the user's context. In another situation, it could be insignificant. Thus, the "significant" change that will alter the current context intrinsically depends on the final application. In addition, the different contexts should be saved in a history that can be used to contextualize communications. Using this history, users can for example

look for people who have a succession of contexts similar to their own.

In the proposed model, the 6 context components do not necessarily have the same importance but each one can affect the mediated communications. For example, the device context could favour written or oral communication. Thus, a user could prefer contacting someone who uses the same type of device as s/he does in order to choose a particular communication modality.

The proposed architecture is a generic basis for contextual communication. We try to be exhaustive to show how the architecture could be used in different situations. Thus, this architecture includes several aspects but some of them will not be functional or useful in every situation. For instance, due to cost reasons, mobile devices can only be equipped with a small selection of sensors (temperature, noise, light, …). Nevertheless, specific remote sensors could be used for particular needs. Therefore, we include these possibilities in the context model.

IV. PROPOSITION

In this part, we present the MeCoCo (Mediated Contextual Communications) system. First we explain the overall architecture and in a second part we describe the prototype.

A. The MeCoCo System Architecture

The general system architecture is illustrated in Fig. 2. We discern two parts in the Context Manager system: the client side (Client Context Manager) and the server side (Server Context Manager).

1) Client Context Manager

The Client Context Manager can be broken down into three parts:

- The *Interface Manager* which manages the interface and displays information from the server.
- The *Information Collector* which collects information that builds the context via external sensors (ambient noise level, room temperature, RFID reader to know the existence of materials, a GPS to know the location, …) or internal sensors (to determine the connection speed of the device, the screen size, …).
- The *Context Generator* module which receives information from the *Information Collector*, sorts and combines this information to generate an XML file which is sent to the server. The structure of the XML file is not described here but the Document Type Definition (DTD) covers all the dimensions described in Fig. 1 (User's Profile, Activity, Device, Environment, Location and Time).

2) Server Context Manager

The Context Manager server is composed of two parts:

- The *Contextualizer*, the most important part, manages the contextual discussions engine. This module receives information from the client side through XML files. First, the *Contextualizer* stores the context relevant to the new user (C_x) into the database (*Contexts DB*). Then, it searches all the "close" contexts of C_x with the *Inquirer*.

To do this, the notion of proximity is crucial to find "close" contexts. Thus, we define the global proximity between two contexts C_x (with $< U_x, A_x, D_x, E_x, L_x, T_x >$)

and C_i (with $< U_i, A_i, D_i, E_i, L_i, T_i >$) with the following formula:

$$\text{Global\_Proximity } (C_x, C_i) = \sum_{j=1}^{6} (\sigma_j * \text{proximity } (D_x[j], D_i[j]))$$

$$D[j] \in \{U, A, D, E, L, T\}$$
$$\sigma_j = \text{weighting coefficient of a dimension } D$$

The calculation of the proximity between two dimensions is totally dependent of the application domain. For instance, the proximity between two dimensions *Time (Tx and Tj)* can be considered high if two contexts happen in the same hour for an application but could be low for another application. To simplify the system, we choose to define the proximity as a value between 0 and 1.

The weighting coefficient of a dimension (σ_j) is important because it allows organizing the six dimensions into a hierarchy. The dimensions that should be put forth compared to others depend on the application and on the user's needs.

Finally, after the identification of all the contexts considered as being close to the users' current context C_x, the *Contextualizer* searches for the people and for the messages relevant to these contexts.

- The *Resources Deliverer* returns the contextual information as XML data to the *Interface Manager* on the client side that will display the information to the user. The MeCoCo system was implemented based on this architecture.

B. The MeCoCo prototype

The main goal in the conception of the prototype is to show the feasibility of the proposed approach and architecture. Furthermore, we also aim at developing a demonstrator for the computer-human interface.

We developed the various modules of the Context Manager including the Context Client Manager and Context Server Manager. The Information Collector is simulated in the prototype, i.e. the environment (location...) is not caught by sensors but is simulated.

The MeCoCo prototype was developed using various programming languages such as PHP, JavaScript and Flash. We combine these languages with HTML and XML files and also with an SQL relational database. AJAX technologies were used to favour dynamic interface.

The prototype was built to be used while visiting a museum. The museum is divided as following:

- Each floor is dedicated to a specific painting movement (renaissance, baroque…).
- On each floor, the rooms are specialized in one artist (Monet room, Picasso room, etc).

The MeCoCo interface offers a large amount of contextual information to the user. Two types of contextual information are presented to the user:

- Contextual contacts (= users) who have been or who are in a context close to the user.
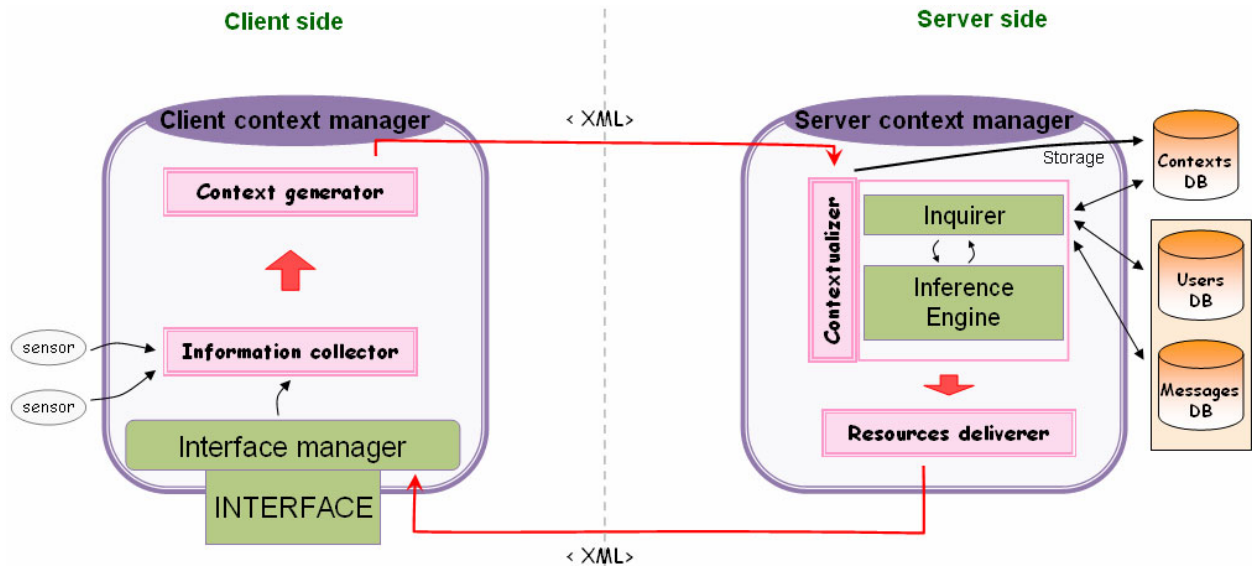- Contextual messages which have been written by these contacts.

Figure 2.   MeCoCo architecture

The contextual contacts proposed to the current user have at least one similar dimension of context. Each contact is represented by an avatar with specific accessories that represent the six dimensions (user's profile, activity, device, environment, location and time). Fig. 3 shows this avatar. A contact who has written a message is shown in Fig. 4.
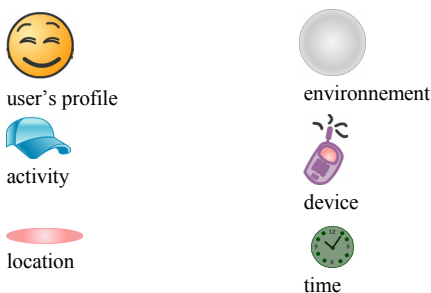

Figure 3.   Avatar

Figure 4.   Avatar with a message

Legend:

user's profile

environnement

activity

device

location

time

If a contact dimension is not similar to the current user one, the corresponding item is not coloured on the avatar (Example: if the profile is close and otherwise).

In the prototype, we voluntarily simplified the calculation of proximity between dimensions. The proximity value is equal either to 1 if two dimensions are similar and 0 otherwise. In the prototype application, for each dimension, the proximity is equal to 1 in the following cases:

- Location: the same room in the museum (i.e. "Picasso Room", "Monet Room", …)

- Activity: the same activity (i.e. visiting the museum, guiding visitors, …)
- User's Profile: the same general profile (student, tourist, guide, …)
- Time : connected at the same time (connected or not)
- Device: the same device (PDA, smartphone, …)
- Environment: the same ambient noise (quiet, noisy, …)

Furthermore, we decided to use the same weight coefficient for the 6 dimensions. The global proximity can therefore take any integer value between 0 and 6. We are aware that these choices are not necessarily very realistic but they are suitable to test the prototype.

The prototype offers three different views:

1)   A Global View

The global view (Fig. 5) shows contextual contacts on a radar on a scale of one to six. A contact in a context C_i will be placed on the radar according to the global proximity value. Thus, if the context C_i has a global proximity of 6 with the current user's context, it will appear on the left of the radar (in the zone 6/6). On the contrary, a user having only one similar dimension of context will appear on the right of the radar (in the zone 1/6). The interesting aspect of this representation is that users are able to choose which contacts or messages seem to be the most appropriated according to the dimensions they want to favour at that particular moment.

Fig. 5 shows the global view interface of a user named Matthieu Lemarchand. This user can compare his context to the ones of the proposed contacts by placing the mouse over the corresponding avatar. By clicking on an avatar, more complete information appears in the bottom part of the interface. To read a comment left by a contact, the user has to click on the message above the avatar.

In Fig. 5, the avatar (1) represents a user with 6 identical dimensions to those of the current user. The avatar (2) represents a user with a single dimension (Device) in common with the current user.
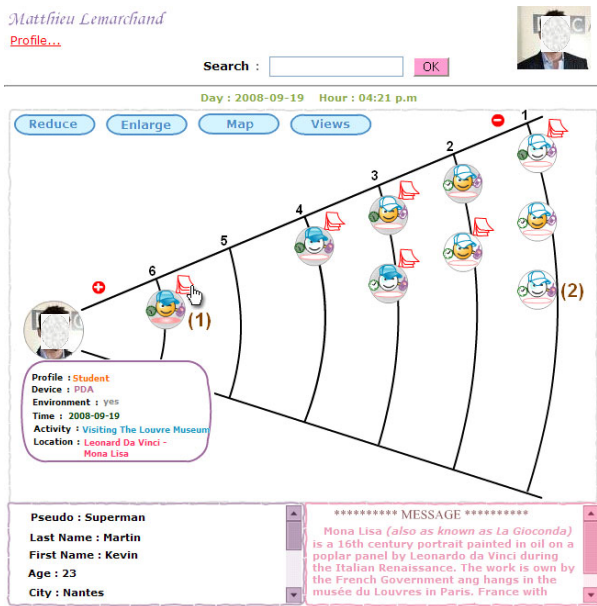
Figure 5.   The global view of contextual contacts

It is important to notice that the system aims at giving a certain degree of transparency by letting the users inspect information about their own context. They can also modify the information (such as their profile) and choose which data they want to provide to the system. To sum up, the system is totally under the control of the users because they can control the information they provide to the system and, even though the system proposes contacts, the final choice is left to the user. With this feature, we believe that the system will be widely accepted and used.

2) A Time View

The prototype interface offers a second view which highlights the time aspect. This time view places contextual information according to two axes: the vertical axis corresponds to the time and the horizontal axis represents the context score on the five other dimensions (except time). The main aim of this view is to give more precise information on time. Thus, in the prototype, we propose a simple time scale with six values (from "Now" to "A year ago", Fig. 6).
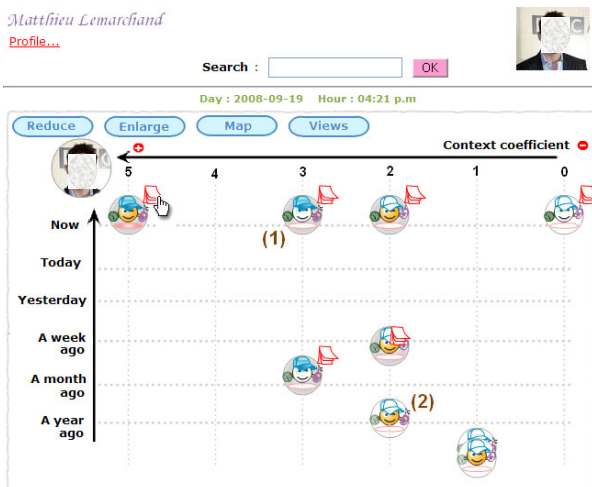
Unlike the global view, it is possible for a contact to have a context score of 0/5. This means that the time is the only dimension in common between the current user and this contact.

In Fig. 6, the contact (1) represents a user who is logged at the same time as the current user, with 3 proximal dimensions of context (Activity, Device and Environment). The contact (2) represents a user with a similar profile and device but who wrote a message one year ago.

3) A Location View

In the location view, the principle is similar to the time view but it is the location aspect which is highlighted on a particular axis.

For the prototype, we created a model of location based on painting movements and painters. This model is not detailed here but it is a taxonomy tree with different levels: painting, painters, painting movements and other pictorial tendencies. In fact, this classification is done according to the layout of the rooms in the museum. So, with this location view (Fig. 7), users can easily see the proximity of contacts regarding the painting they are currently looking at.

In Fig. 7, the current user is located in front of the Mona Lisa painting and can look for contacts or messages relevant to the painting, to the painter Leonard Da vinci (the same room) or to the renaissance painters (the same floor). This particular view sorts contacts according to the location proximity but still expresses information on the proximity score on the five others dimensions.

The prototype location view shows that it is possible to define the proximity for a context dimension with a fine grain. The solution should be chosen according to the specific application aims.

V.   DISCUSSION

We have defined a user model context for specific applications: mediated communications between human users. Compared with existing context models, the proposed model suggests a broader vision of context by integrating users themselves in the model thanks to their profile as well as their location, their current activity, the device they are using and their environment at a given time.

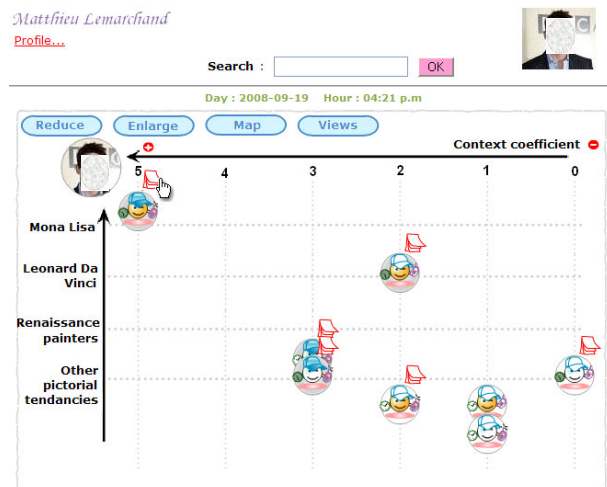Figure 6.   The time view of contextual contacts

Figure 7.   The location view of contextual contacts

Ranganathan *et al.*'s model [3] take into account the location, the environment, the user status and activity but does not consider the temporal aspect. In our opinion, time is important to manage the change of context but also to define the validity of a context. Kirsch-Pinheiro *et al.* [6] limit the context model to two aspects: physical (location, device and application) and organizational (group, role, calendar, activity, objects and processes shared). This modeling does not take into account the context temporality nor the context environment as context dimensions that affect human communication. The environment dimension is present in certain models but they do not integrate the four aspects defined in the MeCoCo model (ambient, material, social and related environment). All of these four last aspects are important to understand the user's context at a particular moment. The importance of this dimension is shown by Guralnick [20], when he uses environmental context to design m-learning products.

Moreover, the human-computer interface of MeCoCo is another significant contribution. The MeCoCo interface offers a great amount of freedom to the users by letting them choose the people and the discussions they might be interested in. Furthermore, users have the possibility to choose the dimension they want to emphasize regarding their context. Unlike other systems, this type of visualization does not put the different context dimensions in competition. The main advantage is that the system enables the users themselves to choose the dimension which seems the most important to them.

VI. CONCLUSION

In this paper, we propose a model of user-centred context to promote mediated communications between humans. This model takes into account six key dimensions: user's profile, activity, device, environment, location and time. Based on this model, we then build a context-sensitive architecture system dealing with communications between users. In addition, the implementation of a prototype, named MeCoCo, enabled us to test the model and to propose a suitable user interface to choose relevant contacts and discussions.

In order to improve the prototype, we have decided to focus on three different research issues:

- Developing advanced search features that will allow users to make proactive search by using the history of contexts or via the definition of a "virtual" context.

- Setting up a mechanism to control the context stability. With the current model, the change in a dimension leads automatically to a change of context. In reality, we should ensure that the context does not change too fast to make the system more usable. We will build a mechanism that identifies important events and which could be parameter according to the different applications.

- Defining classes or patterns of contextual models. The limits of the current model lie in the necessary work that has to be done for each specific application domain. For instance, we would have to define an activity model and a location model before using our solution in a new application field. We do not believe that context aware applications can be totally generic. Nevertheless, we could propose some classes of models (for example based on domain ontologies) in order to avoid the design of models from scratch for each application area. Models for a given application could then be built starting with these classes of models.

Finally, this type of work cannot be done without asking questions about the confidentiality of contextual data that contain personal information. This raises ethic issues. The disclosure of such data may affect the individuals' privacy protection. In addition, it also raises the question of user's consent about the disclosure of personal information. So to deal with the confidentiality of data and privacy of users, we plan on creating metadata to define the level of confidentiality that users want to use for each context dimension. The idea is to give full transparency to users and give them the choice to see and show what they want at any time.

REFERENCES

[1] Calvary G., Coutaz J, Thevenin D., Limbourg Q., Bouillon L., Vanderdonckt J., A unifying reference framework for multi-target user interfaces, *Journal of Interacting With Computer*, Elsevier Science BV, June, Vol 15, Issue 3, pp. 289-308, 2003.

[2] George S., Labas H., E-learning standards as a basis for contextual forums design, *International Journal of Computers in Human Behavior*, Vol.24, Issue 2, Elsevier, pp. 138-152, march 2008.

[3] Ranganathan A., Campbell R.H, An infrastructure for context-awareness based on first order logic, *Personal and Ubiquitous Computing*, Vol 7, Issue 6, Springer London, pp. 353-364, 2003.

[4] Ranganathan A., Campbell R.H, Ravi A., Mahajan A., ConChat: a context-aware chat program, *Pervasive Computing IEEE*, Vol 1, Issue 3, pp. 51-57, 2002. (doi:10.1109/MPRV.2002.1037722)

[5] Kirsch-Pinheiro M., Villanova_Oliver M., Gensel J., Martin H., Context-aware filtering for collaborative web systems: adapting the awareness Information to the users Context, *20th ACM Symposium on Applied Computing (SAC 2005)*, Santa Fe, New Mexico, Mars 2005, ACM Press, pp. 1668--1673, 2005.

[6] Kirsch-Pinheiro, M., Gensel, J., Martin, H., Representing Context for an Adaptative Awareness Mechanism, *X International Workshop on Groupware*, San Carlos, Costa Rica, pp. 339-348, Sept. 2004.

[7] Wang X., Zhang D., Gu T., and Pung H., Ontology- Based Context Modeling and Reasoning using OWL, *Proc. 2nd IEEE Annual Conference on Pervasive Computing and Communications Workshops*, pp. 18-22, 2004.

[8] Kiss, C. Composite Capability/Preference Profiles (CC/PP): Structure and Vocabularies 2.0, W3C Working Draft, http://www.w3.org/TR/CCPP-struct-vocab2/, 30 April 2007.

[9] Open Mobile Alliance, User Agent Profile, http://www.openmobilealliance.org/Technical/release_program/uap_v2_0.aspx, 6 February 2006.

[10] Chen H, Finin T., Joshi A., "An Ontology for Context-Aware Pervasive Computing Environments*", Special Issue on Ontologies for Distributed Systems, Knowledge Engineering Review*, Cambridge University Press, Vol 18, Issue 3, pp. 197-207, 2004.

[11] Strang T., Linnhoff-Popien C., Frank K., CoOL: A Context Ontology Language to enable Contextual Interoperability, *Proc. 4th IFIP International Conference on Distributed Applications and Interoperable Systems*, pp. 236-247, 2003.

[12] Basaeed E.I., Berri1 J., Zemerly M. J., Benlamri R., Web-based Context-Aware m-Learning Architecture, *International Journal of Interactive Mobile Technologies (iJIM)*, Vol 1, No 1, pp 1-6, 2007.

[13] Laine T. H., Joy M., Survey on Context-Aware Pervasive Learning Environments, *International Journal of Interactive Mobile Technologies (iJIM)*, Vol 3, No 1, pp. 70-76, 2009.

[14] Dey A.K., Abowd G.D, CybreMinder: a context-aware system for supporting reminders. *In 2nd International Symposium on handheld and Ubiquitous Computing*, Springer, pp. 172-186, 2000.

[15] Tang J., Yankelovitch N. et al, CoNexus to AwareNex: extending awareness to mobile users. *Proc. SIGCHI Conf. Human factors in Computer Systems*, pp. 566-573, 1998.

[16] Law C.F., Zhang X., Chan M.S.M, Wang C.L., Smart Instant Messenger in Pervasive Computing Environments, *Advances in Grid and Pervasive Computing*, Lecture Notes in Computer Science, Volume 3947/2006, Springer-Verlag, pp. 32-41, 2006.

[17] Dey A.K., Understanding and using context, *Journal of Personal and Ubiquitous Computing*, Vol 5, Issue 1, pp. 4-7, 2001. (doi:10.1007/s007790170019)

[18] Bouzeghoub A., Do K.N., Lecocq C., A Situation-Based Delivery of Learning Resources in Pervasive Learning, *EC-TEL 2007*, LNCS 4753, Springer-Verlag Berlin Heidelberg, pp. 450–456, 2007.

[19] Calvary G., Coutaz J, Thevenin D., A Unifying Reference Framework for the Development of Plastic User Interfaces, *In IFIP WG2.7 (13.2) Working Conference, EHCI01,Toronto, Springer Verlag Publ., LNCS 2254, M. Reed Little, L. Nigay Eds.*, pp. 173-192, 2001.

[20] Guralnick D., The Importance of the Learner's Environmental Context in the Design of M-Learning Products, *International Journal of Interactive Mobile Technologies* (iJIM), Vol 2, No 1, pp. 36-39, 2008.

AUTHORS

Sébastien George (Sebastien.George@insa-lyon.fr) is with the National Institute of Applied Sciences, Université de Lyon, INSA-Lyon, LIESP Laboratory, F-69621, Villeurbanne, France.

Aina Sylviane Lekira (rasoa_aina@yahoo.fr) is with the National Institute of Applied Sciences, Université de Lyon, INSA-Lyon, LIESP Laboratory, F-69621, Villeurbanne, France.