# Computer-assisted learning based on a ubiquitous environment
## - Application to Japanese Kanji learning. -

Yann Jacquinot, Shin Takahashi, Jiro Tanaka
Department of Computer Science
University of Tsukuba

The application of "Ubiquitous Computing" to learning brought new issues in research. Bringing the right learning interaction at the right time in the right situation characterise "Ubiquitous Learning"(U-learning). Therefore, understanding the learner context in any situation is the key point to bring U-learning interactions. We propose and design a framework to bring context aware interactions in learning applications. We also aim to fill the gap between raw data provided by sensors or simple context data, and high level of context information needed by the learning. By using several learning scenarios applied to Kanji learning, we show how ubiquitous interactions increase learning efficiency. Our framework is based on three formalized ubiquitous interactions, and on a context definition representation for U-learning. The three context interactions are: recording context, triggering application action on detected context, and augmenting digital data with context information. The definition representation allows the framework functionalities to manipulate, detect, and create context information, and then enhance the low level sensor data to the higher level needed to provide application U-learning interaction. We describe the design of this framework, and discuss its possible evaluation and evolution.

### 1. Introduction.

The Application of Ubiquitous Computing characteristics, (*24/7, anywhere, anytime, natural interface, context awareness, invisibility*)[1][2], to learning systems could be considered as defining Ubiquitous learning. However, these characteristics can be also found in "Mobile Learning" called M-learning. Therefore, U-learning must not be reduced to Mobile learning. According to Fisher[3], in our world information is available at any time, at any place, in any form, but the challenge is to say the "right" things at the "right" time in the "right" way.

Understanding the learner context in any type of situation, not only during the learning ones, at one moment in time, is the key point for bringing a Ubiquitous learning interaction. There is a gap between the raw context information provided by the sensors or application data which bring simple context information as user schedule, and the higher level of information the learning application needs in order to bring ubiquitous learning interaction.

We want to propose a generic framework to fill this gap and provide this specific U-learning interaction to the Learner, then enhance the learning. The framework must be generic enough to be applied to any U-learning type of application.

On the conceptual aspect, the framework must provide formalized specific ubiquitous learning interactions that we believe any U-learning application should provide. They are: record and remind learner context, augment digital data with context information, detect and infer learner context to trigger actions to the learning application.

In order to fill the gap between low level context data and the higher semantic level needed by the applications, the framework should include three models which represent our particular context definition for ubiquitous learning. These concepts are used in the framework by a

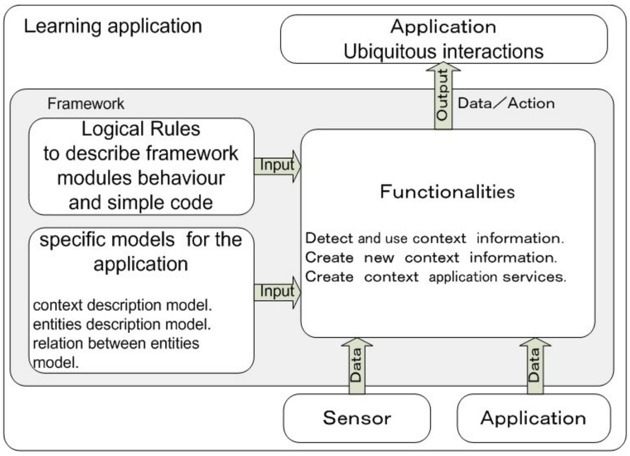


*Fig 1: Approach description.*

set of functionalities to create, detect and use context information from sensors data and application data, then provide context services which represent the three formal U-learning interactions to the learning application *(Fig 1)*.

### 2. Scenarios.

These scenarios describe examples of the ubiquitous interactions that the framework would bring on the Japanese language learning and kanji learning. They are showing how a specific interaction, related to the user situation (U-learning interaction) can enhance learning.

We consider as background a learning scenario. A learner named "Paul", possesses a learning system implanted in a mobile device. This learning system allows him to practice reading and writing Kanji exercises. Paul's tool includes all necessary secondary tools like dictionary, exercises sheet etc. With this tool, Paul has mobile-learning interaction at any-time and in many environments of his daily life .

### 2.1 U-learning Scenario 1:

*Situation A.*

Paul is in a real life situation at the post office to buy some stamps. Even if he is not in a learning situation, he uses his integrated dictionary to search for Kanji and words. For example, he discovers on a signal at the post-office "禁煙 forbidden to smoke". Naturaly, Paul easily understands the meaning of this signal thanks to the symbol on it. However, he does not know the reading of this word. Therefore, he uses his learning tool dictionary to search this reading and he finds it as being "kin en".

In a traditional learning situation, the interaction would stop here, and Paul would have to organise himself to memorize this new knowledge. But, it is not common for a learner in a daily life situation to have time or the necessary materials to record information on a new word. Therefore, the learner often forgets this daily life learned information.

In a Ubiquitous learning situation, Paul's system suggests him to integrate this new word into his learning program. The system records by itself some context information related to Paul's current situation, like: Paul's location ("post office"), Paul's social environment (Paul "is_with" Pierre and Taro). Paul can also manually record additional information which represents his current situation: taking pictures of the post-office, inputting a note or context information through the application menu (e.g. "buying a stamp").

At the same time, the system has to bind the new knowledge with some context information and store this new context augmented data in an application database. In our example, the system binds the word "禁煙" to Paul's location: "post-office". Compared to the previous additional information recorded by Paul, these data do not represent a part of Paul's situation, but they represent a semantic link between the knowledge and information on Paul's situation.

*Situation B.*

One of the following days, Paul is using his mobile device for practicing his Japanese language. In this learning interaction he is practicing reading and writing sentences. The system knows that Paul discovered and recorded in a precedent situation the new word "forbidden to smoke: 禁煙".

In order to reactivate Paul's memory, the system intends to build reading exercise with this new word. (ex:" これは禁煙席です。", meaning "this is a non-smoking seat"). If Paul fails on the word "forbidden to smoke", the application will intend to reactivate Paul's memory, by reminding him the context information recorded during *situation A* (at the post office). To realise that, the system shows Paul the picture taken at the post-office and reminds him the other context information inputted and recorded previously.

The learners has the following advantages from these U-learning interactions described by *Situation A* and *Situation B:*

- The learning exercises can include also knowledge discovered in daily life situations, without any particular effort from the learner
- The system will reactivate the learner memory quickly by creating exercises which include this new knowledge.
- In the case of the learner's failure, recalling the information related to the real situation when the knowledge was learnt may certainly reactivate other part of learner memory ("souvenir") and help him to remember the knowledge.

### 2.2 U-learning Scenario 2:

Beside this capability of the framework to remind context information in order to reinforce learner memory, we saw previously that the system binds knowledge information with context information and stores this bound information in an application database. In our practical case, the record will be Japanese words with learner's location. Every learner who is using the application concurs to update this database. During his training, Paul can request the application to provide him with a specific exercise related to a specific context.

If Paul requests exercises related to the specific place "post-office", he will obtain in the exercise contents words like *stamps, letter, post card and "forbidden to smoke"*. In this situation Paul does not use only the information he personally updated in the system, but also some other user recorded information.

Compared to a traditional learning application that would use a static contextual lexicon to propose this kind of specific exercise, this application allows the automatic update of this lexicon from all users. Moreover this dynamic lexicon will be based on learner's real daily experiences.

### 2.3 U-learning Scenario 3:

*Situation C.*

Later, Paul is at a post-office again. The system detects context information related to Paul situation, compares this current situation with a previously recorded situation in order to detect similarities between both situations. In our specific example, there is already a record of Paul at the post-office. The location is similar, but other information related to the situation is different (friends, time and activity). However, the application looks at the knowledge bound with post-office, and finds the word "forbidden to smoke 禁煙".

The application will discretely try to catch Paul's attention and will suggest him a non mandatory exercise related to this word on his mobile device. If Paul is busy, he can ignore completely this interruption. However, the proposed exercise may catch briefly Paul's attention, and even if Paul does not satisfy the interaction, he will have seen the context information and the knowledge related to this exercise. Therefore, it will reactivate by default part of Paul's memory on this word.

In this scenario, like in scenario 1, the learning objective is to reactivate the learner's memory by reminding him a link between knowledge and information related to a real living situation. But they are acting on the opposite ways. In the first scenario, from the exercise failure, we are reminded the context information. In this scenario, the application helps in reminding the knowledge from context living situation. We believe that such complementary interactions can improve learning efficiency.

These scenarios show several ubiquitous learning interactions examples applied to Kanji learning domain. These interactions enhance the learning by their capacity to adapt the learning interaction to the learner situation.

## 4. Framework formalized interactions.

In order to provide a framework which realizes these interactions in U-learning, they must be formalized to be include in the framework as concepts. These formalized interactions define the requirements for the module that will provide concretely these interactions as services for the learning application.

### 4.1 Context recording.

The framework must be able to store and remind context related information linked to one situation.

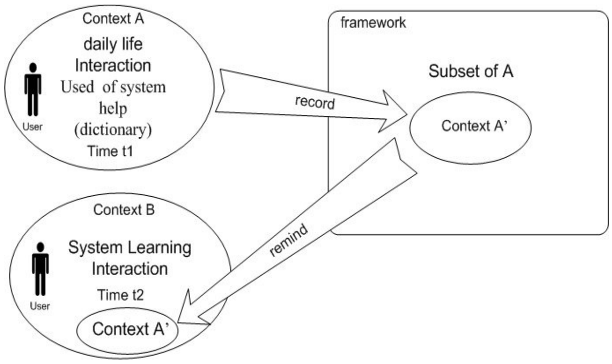


*Fig 2: Context recording.*

In context recording (Fig 2), the framework records a subset A' of the real context A. The situation A concerns a daily life situation at time t1. At time t2>t1 during learning activities (context B), it will propose an interaction with the capability to remind the context reduction A' to the user. We can find this interaction in scenario 1.

### 4.2 Digital data augmentation.

The second interaction included is the capability to bind knowledge data to context information, to store, extract and present these data according to the learning application's request.

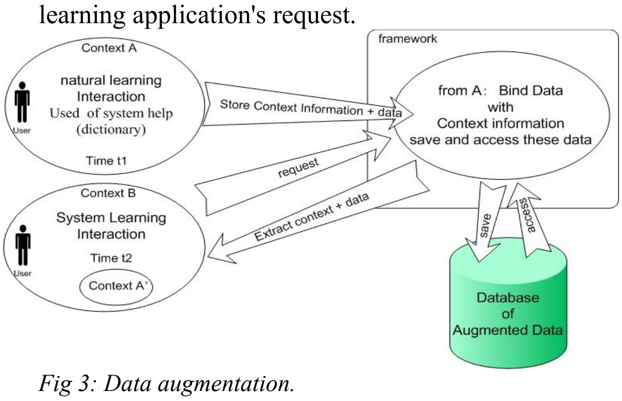


*Fig 3: Data augmentation.*

In data augmentation *(Fig 3)* from a daily situation (Time t1, context A), we bind knowledge data to some context information extracted from A. Then, we insert a record representing this binded information in a database. When needed, in Situation B (context B, time t2>t1), the system accesses this database and extracts digital data or context information related to some digital data, in order to propose a specific interaction. This interaction is used in scenarios 1, 2 and 3.

### 4.3 Context inferring

If there is a similarity between the two situations, an application service is automatically triggered, which brings an interaction in the right context at the right time.

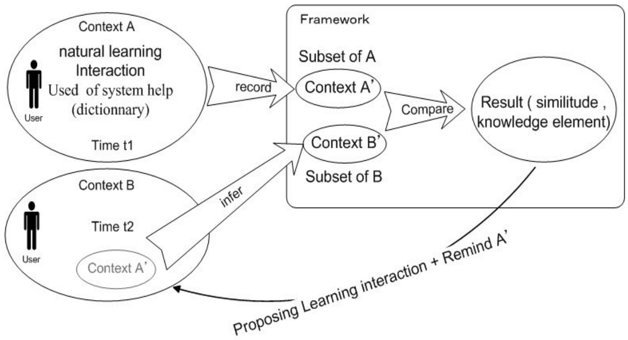


*Fig 4: Context inferring*

In "context inferring"*(Fig 4)*, the framework infers in real time a subset of the current user context (context B time t2), Context B'. It compares B' with the previously recorded subset of context contextA'. When it infers a similitude between both contexts subset, it generates an action to the learning application. This interaction is illustrated on a concrete example in scenario 3.

## 5. Framework context definition and representation.

To realise the transformation from the low level data to higher semantic data usable to provide the application service to the learning application, the framework functionalities need a representation of context definition. This representation offers information on how to manipulate, create, detect context information. Several definition of context were given by different researchers depending on their domain and their point of view [5][7]. We must find or define one and its representation appropriate for ubiquitous learning.
Our definition of context in Ubiquitous Learning is:

*Context in U-learning is any information that describes partially the subjective, physical and social situation of the learner depending on the application needs.*

We propose a general definition, centered to the learner, where the context information must be categorized in three groups: physical, subjective and social.

### 5.1 Framework Entities.

The context representation of the definition must be complemented by a model representing the entity in the framework, because some information that are not all time relevant to the learner, but which concern other entities must be considered. For that we need a representation of the entities in the system.

There are 4 types of entities that could be concerned with context.

- **Living Things** : (user , animal , person).
- **Un-living Things** or **Objects**: (car, computer , book).
- **Place**: (room , building, geographical position)
- **Application**: (entity in virtual world ).

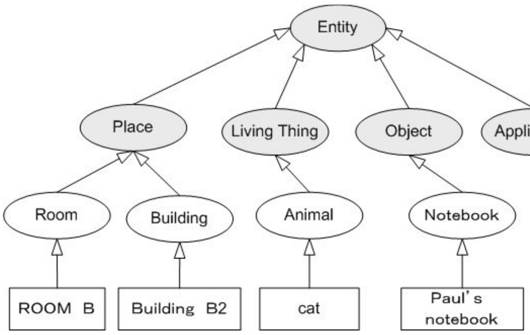


*Fig 5 : Example of entities model.*

### 5.2 Relations between the entities types.

To be complete, the previous representation has to be completed by a representation of the possible relation between different entities in order to be able to link a context information to other entities .These relations can be static or dynamic.

**Living-things ↔ Living-things :**
Proximity (Location) , instant relation (Activity).
*Static:* (Social relation).
**Living-things ↔ object :**
Proximity (Location) , relation usage (Activity).
*Static:* possession (Social relation).
**Living-things ↔ place :**
Position, proximity (Location), displacement (Location).
*Static:* home, work... (Social relation).
**Living-things ↔ application :**
Usability proximity (Location), use, link to (Activity).

**Object ↔ Object :**
Include, proximity, relative positioning... (Location).
*Static:* constitution (Organisation).
**Object ↔ place :**
Inclusion, precise positioning (Location).
*Static:* inclusion (Organisation).
**Object ↔ application :**
Include (Location), link-to (Activity).
*Static:* relation of application to object (Architecture).

**Place ↔ place :**
Organisation, inclusion, proximity... (Location) (Organisation).
*Static:* (Architecture).
**Place ↔ application :**
Link to (activity).
**Application ↔ application :**
Connectivity, communicate (Activity) (Organisation)
*static:* (Architecture).

Finally, we need three representations to describe the context definition in U-learning and to be able to consider any type of information that can be relevant to a learner situation. These representations must be extended when building a specific application, like *Fig 5* for the entities representation. However, the framework includes models which provide the base to the functionalities manipulating and transforming context information.

## 6. Framework functionalities description.

We can focus now on the specific functionalities and the structure used to enhance raw level context data to the learning application level and provide the Ubiquitous Learning interactions. The framework is built on a multi agent platform to benefit of its capability of communication. *Jade* (Java agent Developments environment) with the use of Java and its normalization in the communication aspect between the agents is a good choice for the design of such a framework.

### 6.1 Structural description.

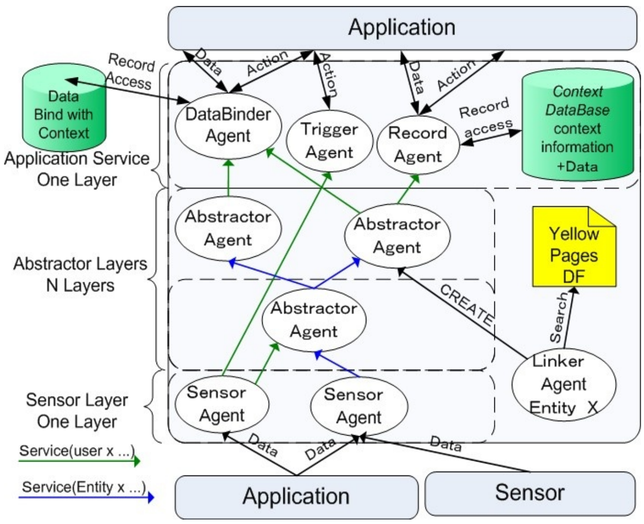


*Fig 6 : Framework: Structural view.*

The framework is structured in layers *(Fig 6)*. The agents exchange one unique type of formalized information context service, that describe the context data.

The lowest layer, the sensor layer composed by sensor agents, represents the input interface between application and the framework. The abstractor layer transforms context information services provided by sensor or other lower abstractors, to higher level of information.

The higher level brings concretely the context aware interactions formalized in the framework and uses only services related to the learner according to the definitions.

The binding between interaction and agent role is:

- Context recording: Record agent.
- Context inferring: Trigger agent.
- Digital data augmentation: Data Binder agent.

The linker represents a special function for the framework. It infers new services concerning the User entity, from service concerning other type of entities. It allows automatic context information enhancement.

### 6.2 Context service.

The different agents will exchange messages and will use the concept of service to communicate context information together. One service describes a context information related to one entity. It is formalized by the n-tuple:

*Service_name(Entity name,*
*Context variable name,*
*Context variable value,*
*Time,*
*Reliability)*

There is a major advantage to formalize on one type of information the information exchanged by all the modules in the framework. This induces a high level of decoupling between the different type of functions realised by the different modules.

## 7. The linker role.

The "Linker" role represent a special function in the design framework. It aims to detect if context information related to another entity can be context information of the learner. Then, the "Linker" considers any kind of information presented in the sensor and abstractor layer, combines them and creates new services related to the learner when these information become relevant to him. There is one linker per entity in the system. It play a main role in enhancing the level of context information by automatically bringing information closer to the application level.

**Linker Example :** illustrated by *(Fig 8 - Fig 9).*

In this example, two sensor agents exist: one provides activity in a room and the other provides the learner activity.

*Room_activity(room B ,activity , "meeting_A", time ,100).*
*User_activity(Paul,activity,"meeting_A",time,80).*

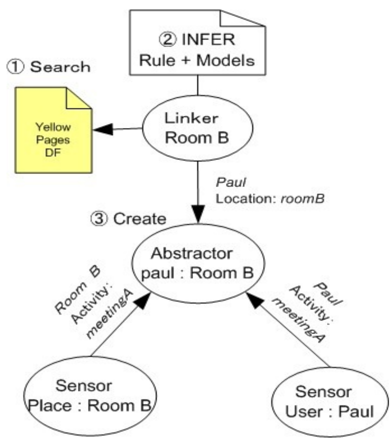


*Fig 8: Linker functioning example.*

The Value of activity describes one instance of meeting, not the general concept of "having a meeting".

① The linker searches the yellow pages and finds other service related to meetingA : Paul (activity:meetingA).

Is there any relation possible between Paul and room B?

② -The linker sees in the entity model .
Paul is Living thing, room B is Place.
-The linker looks at the relation between Living-Thing and place, and he finds Location. After checking the context model, linker knows that we are interested in the user location, then linker can infer
*location_user(paul,location,"room B",time,80).*

③ Linker creates an abstractor who will create this new service by registering to the two concerned sensor.
The abstractor has the following rules:
register user activity
register room B activity
if user activity = room b activity
create user( location ,roomB, time , reliability)
When information changes, for example "Paul is not on meeting" , this abstractor dies.

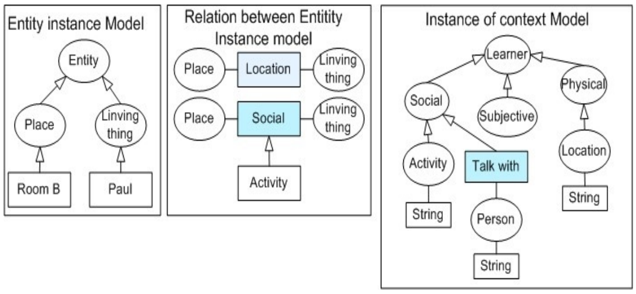


*Fig 9: Models instance.*

## 8. Related work

There are several related works that talk about context in application. We will overview two of these works related to Ubiquitous learning and ubiquitous computing in general, respectively.

The first one is concerned with work on ubiquitous learning application for Japanese language learning. We share the view in this work on situational learning in language for the definition of U-learning interaction. The authors propose a context aware application that brings situational learning applied to Japanese language. However they take an instance of context needed for their specific application.

We differ in our intention to make a generic framework to bring U-learning interactions.

In Ubiquitous computing in general, the work of A.Dey and G.Abowd [8] proposes a toolkit to help people to design and bring context aware features in Ubiquitous application. This toolkit describes the manner to create context aware applications. There is a guide line of the different modules necessary in order to make a context aware application. A.dey based his toolkit on his general definition of context[7]. Our work differs from A.Dey's work by the specific definition of context for U-learning, and by providing in our framework concrete functionalities that will detect and enhance automatically context information and bring them closer to the application service level.

## 9. Discussion on evaluation.

There are several different aspects on which our framework needs to be evaluated. The main goal is to create a framework to bring Ubiquitous learning interactions in application. Therefore, the capability to understand and use the framework, in order to integrate context aware interaction in learning application, will be a global subjective criteria of evaluation.

However, under this global evaluation goal, there are several technical aspects that must be evaluated first. One of them is the capability of the linker to properly infer relations between different context information. This means evaluate Linker's efficiency to provide context services that we expect when we provide information only in the form of models.

Another criteria will be the reliability of the new discovered service. The Linker should not only discover what we expect, but it should also not discover false information.

## 10. Conclusion and future work.

In this paper we describe a framework for providing ubiquitous learning interactions in learning application. We show Ubiquitous learning interactions on practical scenario and show, from the user's point of view, that considering learner situation to bring the right interaction at the right time enhances learning. We formalize this interaction and show the three interactions a U-learning framework must provide (recording/reminding the context, triggering action, and binding data with context informations). This framework is also based on a specific context definition centered to the learner and on additional models which complete this definition. They represent the different entities we can find in context and the possible relations between these entities. These representation allow the framework to fill the gap between raw sensor and application data to the semantic level of data needed by the learning application. These conceptual elements give the key to the framework functionalities to manipulate, detect and automatically create context information, then bring it to the learner's level.

In a future work, we need to implement this framework to integrate these functionalities to the simple kanji learning system example. We also intend to evaluate the framework on the previously exposed criteria.

**References.**


[1]Weiser, M., "The Computer for the Twenty-First Century," *Scientific American,* pp. 94-10, September 1991

[2] Abowd, G.D., and Mynatt, E.D.: Charting Past, Present,and Future Research in Ubiquitous Computing, *ACM Transaction on Computer-Human Interaction,*

[3] Fischer, G.: User Modeling in Human-Computer Interaction, *Journal of User Modeling and User-Adapted Interaction (UMUAI)*, Vol. 11, No.1/2, pp.65-86, 2001.

[4] Ogata, H., and Yano, Y: "How Ubiquitous Computing can Support Language Learning", Proc. of KEST 2003, pp.1-
6,2003.

[5] Yuan-KaiWang: Context Awareness and Adaptation in Mobile Learning. WMTE 2004 154-158

[6] Chen, G. and Kotz, D.: A Survey of Context-Aware Mobile Computing Research. Technical Report TR2000-381. Department of Computer Science, Dartmouth College. 2000.

[7] Dey, A. and Abowd, G.: "Towards a Better Understanding of Context and Context-Awareness", Workshop on the what, who, where, when and how of context-awareness at CHI 2000, April 2000.

[8] Dey, A.K., Salber, D. Abowd, G.D.: "A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Applications", *Human-Computer Interaction (HCI) Journal*, Vol. 16(2-4), pp. 97-166, 2001.