Collaboration and the Knowledge Economy: Issues, Applications, Case Studies
Paul Cunningham and Miriam Cunningham (Eds)
IOS Press, 2008 Amsterdam
ISBN 978-1-58603-924-0

Towards Context Modeling for Cooperative Rural Living Labs

Ganna FRANKOVA[1,2], Abdelghani CHIBANI[3],
Yacine AMIRAT[4] and Fabrizio SANNICOLÒ[1]
[1]*ArsLogica, Viale Trento, 117, Mezzolombardo (TN), 38017, Italy*
Tel: +39 0461 600253, Fax: + 39 0461 603250, Email: ganna.frankova@arslogica.it
[2]*DISI, University of Trento, Via Sommarive 14, Povo, Trento, 38050, Italy*
Tel: +39 0461 882092, Fax: +39 0461 883964, Email: frankova@disi.unitn.it
[3]*CityPassenger, 14 avenue du Québec, VilleBon, 91945, France*
Tel: +33 1 60138594, Fax: + 33 1 60138594, Email: achibani@citypassenger.com
[4]*Paris XII University, 120 rue Paul Armangot, Vitry sur Seine, 94400, France*
Tel: +33 1 41807383, Fax: + 33 1 41807383, Email: amirat@univ-paris12.fr

Abstract: There are many obstacles in the path of development in rural European areas. The Collaboration@Rural (C@R) European project aims to remove such obstacles through collaborative technologies adopted among Rural Living Labs across Europe. In this paper we define a context model that represents context information used in the Rural Living Labs involved in the C@R project. The proposed model is based on ontologies, which offers designers capabilities to describe and extract semantic from context information and build reasoning process on top of them. The design of Rural Living Labs context ontologies is based on an incremental approach using a common template, which is an interesting proof of concept as it lets us validating the context model.

Keywords: Context Awareness, Ontologies, Rural Living Lab, Collaborative Services, Collaborative Working Environment.

1. Introduction

In large scale pervasive environments as Rural Living Labs are, designing context-aware collaborative services need an open and extensible framework. Building such services requires resolving three main issues: user context awareness, seamless services management mechanisms, and semantic interoperability.

This paper presents a part of work carried out in the C@R: “A Collaboration Platform for working and living in rural areas” EU project with reference FP6-2005-IST-5-03492. The paper defines a context model that represents the context information of the Rural Living Labs involved in the project.

Context awareness was first introduced by Schilit and Theimer [1]. In their work on the ParcTab project they defined context of an entity as a set of information concerning the identity of the entity, its location, identities of nearby objects and changes to those objects. Ryan et al. [2] present context of an entity as its environmental information, such as location, time, temperature and its identity. Dey [3][4] considers context of an entity as its physical, social, emotional, and mental (focus-of-attention) environments, location and orientation, date and time of day, other objects in the environment. Abows et al. [5] argue that such definitions only define context by example. One needs a definition to judge whether a piece of information can be considered as context of an entity or not. The authors consider that context is about the whole situation relevant to the application and its set of users. While reasoning on which aspects of the situation are important is a domain specific task. The authors give a generic definition of context we stick to in this work:

Definition 1 [Context] Context of an entity is any information that can be used to characterize the situation of the entity.

An entity can be a person, place or object that is considered relevant to the interaction between a user and an application including the user and the application themselves.

The first research investigation on context-aware computing was by Olivetti Research Ltd. in 1992 [6]. The definition of context-aware application given by Schilit and Theimer [1] restricts context-aware application that is informed about context to the one that adapts itself to context. The definitions given in the state of the art about context-aware computing falls into two categories: using context and adapting to context. There are several works that employ the more general case of using context.

Hull et al. [7] see context-aware computing as the ability of computing devices to detect and sense, interpret and respond to aspects of an entity's local environment and the computing devices themselves. Salber et al. [8] define context-aware computing aiming to provide maximal flexibility of a computational service based on real-time sensing of context. Dey at al. [3] introduce the notion of adaptation as they consider context awareness leading to the automation of software system based on knowledge of the entity's context. More specifically, Brown [9] considers context-aware application to automatically provide information and/or take actions according to the user's context detected by sensors. Furthermore, the author states that the actions can take the form of presenting information to the user, executing a program according to context, or configuring a graphical layout according to context. Fickas et al. [10] define context-aware application as application that monitors changes in the environment and adapt their operation according to the predefined or user-defined guidelines. Abows et al. [5] propose a definition of context-aware computing we stick to in this work:

Definition 2 [Context-aware Computing] A system is context-aware if it uses context to provide users with relevant information and/or services where relevance depends on the user's task. The authors do not require a context-aware application to detect, interpret and respond to context as Hull et al. [7] do. They only require the response to context allowing the detection and interpretation to be performed by other computing entities. Furthermore, the proposed definition is focused on the user that makes it attractive for such user-centric approach as the C@R project follows.

This reminder of paper is organized as follows. Section 2 provides a description of the proposed context information modeling framework. The context information modelling methodology is outlined in Section 3. Section 4 deals with the application of the proposed methodology in several living labs. Finally, in Section 5 we give a short conclusion related to the experience of using a common model template from different ontologies designers and the work in progress.

2. Context Information Modeling Framework

In general context consists of pieces of information that can be used to characterize the situation of a participant entities to an interaction and the interaction it self. We call this piece of information context attribute, e.g., identity of an entity, description or profile of an entity, spatial information (e.g., location, orientation, speed, and acceleration), temporal information (e.g., time of the day, date, and season of the year), environmental information (e.g., temperature, air quality, and light or noise level), activity and schedules and agendas (e.g., talking, reading, walking, and running) [11]. Some context attributes are closely dependent to each other and can be the result of an aggregation or fusion of different context attribute types.

From our point of view, an efficient modeling approach must have characteristics like flexibility, extensibility and expressiveness, which are very necessary to build context-aware system. These characteristics make the system able to identify and describe any contextual complex attribute according to homogenous information representation system. Extensibility is important in the way that it enriches the description and the representation of any new context attribute during run time without review of the global system knowledge model. For example, a person located in a region, could be represented intuitively by GPS coordinates. This representation can be extended seamlessly to include person address and location name e.g., home or enterprise. Ontologies are used to provide more expressiveness and semantics when describing contextual attributes.

The proposed Context Information Modeling Framework is based on a semantic model of context management. This model provides applications and services with transparent context knowledge sharing mechanisms. It is characterized by two core OWL (Web Ontology Language) ontologies: contextual attributes ontology and contextual services ontology. OWL makes it possible to increase context model expressivity by adding new contextual attributes through ontology extension mechanisms. In this work we focus on the design of the context ontology. This latter is considered as Meta layer on top contextual knowledge, it allows context agents to publish a semantic description of the contextual knowledge they provide. This description is understandable by any component service of the living lab environment. Living lab's services or users which are looking for contextual knowledge will use this published descriptions to identify and select the best context provider and how to interact with it.

Context information is modeled by each living lab application designer separately, using a common context template. This involvement and incremental design approach of the living lab designers is an interesting experience and proof of concept as it validates our model and gets the context information directly from the stakeholders of each living labs.

3. Context Information Modeling Methodology

The use of OWL language makes it possible to increase context model expressivity by adding new contextual attributes through ontology extension mechanisms. Inspired from Description Logics DL [13] and DAML+OIL [14], OWL ontology language does not only enable knowledge representation as web based Meta data, but also reasoning on this knowledge. Built on top DLSH constructors, OWL is articulated around three main modeling entities: the class (also called concept), the property (also called role) and the instances of a class. Using these atomic entities, it is possible to extend the core ontologies to describe complex classes. By these extension mechanisms we target to describe the context of living lab users and services. It is also possible to use query languages like RDF-Query [15].

An important feature of OWL is its ability to perform reasoning. For example, class based reasoning consists of subsume tests that allow building a full taxonomy of ontology classes. Properties based reasoning consists of classes' relationship inference among instances of various classes through according to OWL properties like transitive or functional property. To determine in the ontology through transitivity the existence of a relation between instances of two different classes which do not have an explicit relation in ontology. Finally, instances (individuals) based reasoning consists of two tests:

1. Consistency check test to check if a class has an instance in the knowledge;
2. Verification test to verify in the knowledge base the classes which correspond to a full or partial description of a given instance.

3.1 Context Common Template

The proposed context model is represented by a collection of OWL ontologies [12]. The core ontology that we call it Context Ontology consists of a set of basic concepts necessary to describe contextual attribute semantics. Associated to inference rules language, this ontology makes it possible to a context aware service to derive user situation in a specific time.

$$ContextInformation \equiv \forall trueAt.Time \sqcap (=1\ createdAt.TimeInstant) \sqcap \exists contextOf.T \quad (a)$$

$$User \equiv \forall context.ContextInformation \sqcap \forall preference.ContextInformation \sqcap \exists memberOf.Group \quad (b)$$

Formula 1: User Context formulation in Description Logics.

Formula 1 (a) gives a DL description of what means Context Information concept; any knowledge which can be described with its creation and validity time stamp, using createdAt and trueAt DL roles, while Formula 1 (b) gives a DL description of User Context concept using preference and context roles and also describes user as a member of group.

Table 1: Context Core Model Description

Class Name			
Property	**Range**	**Cardinality**	**Role Description**
ContextInformation			
createdAt	InstantTime	=1	Timestamp of context knowledge capture or inference
trueAt	TemporalThing	>=1	Contextual attribute value is a proposition which can be true at a specific time, for example instant or time interval
contextOf	User	=1	Defines who is the user that refers to ContextInformation instances
User			
Context	ContextInformation	>=1	Defines all ContextInformation instances used to describe User context
Preference	ContextInformation	>=1	Defines all ContextInformation instances that represent a particular interest for a user. A list of key values characterizing user preferences.

In Table 1, we give a description of our core ontology properties for context modeling. For each class we denote in the table what are properties value ranges, cardinality and a literal description of its role. Context modeling technique must also take into account user preferences (or desires). Consequently, on an ontological level, it should give the possibility of defining new properties and/or inference rules making it possible to characterize logical relationship existing between captured contextual knowledge, user's desire and/or the triggering of certain actions related to user preferences: for example, when user receives a mail from a preferred contact, system will notify user directly on its mobile.

4. Application of the Methodology

There are seven Rural Living Labs (RLLs) involved in the C@R project, namely, Soria, Turku, Frascati, Sekhukhune, Homokhàti, Czech, and Cudillero Living Labs. The RLLs are divided into four following business categories:

1. Enterprise Incubator. In this category we denote three living labs. Soria, Turku, Frascati, Sekhukhune RLLs that concentrate on the development of services dedicated for agricultural collaborative working environment.

1. Open Communities. In this category we denote Homokhàti RLL that concentrates on automated services and monitoring of agriculture cooperatives.
2. Collaborative Governance. In this category we denote Czech RLL that concentrates on agriculture online services and geospatial portals.
3. Collaborative Fishery. In this category we denote Cudillero RLL that concentrates on traditional fishery enterprises.

We have applied the proposed context modelling approach to the Living Labs. The extension of the common template to design context ontologies for the different Living Labs is realized using Protégé tool that is free and open source ontology editor. Protégé is composed of a set of plugins (sub editor modules) as Class Editor, Properties Editor, Instance Editor, each of which displays a different aspect of the ontology in a specialized view called tab. OWL Plugin is one of the most used plugin. It allows building semantic web ontologies according to the OWL language. It can be used to load and save OWL files in various formats, to edit OWL ontologies with custom-tailored graphical widgets, and to perform intelligent reasoning based on DLs.

Context information used in the different RLLs applications is modeled using the following main inter-related Context sub-classes: Location, Activity, Device, Observation and Sensor. Location class is used to describe any indoor or outdoor location even to describe locations and place or entities locations. We denote two subclasses, namely, OutdoorLocation and IndoorLocation, as they refer to an outdoor place (e.g., the one in the forest) or indoor place (e.g., in some office or building in the town). Activity class allows the description of the activities undertaken by the RLL users, while Device class refers to the devices which are used in the different activities according to the class of the user. In this sense, there are two sub classes of devices: Pocket-PC and PC. Also, these devices can have different functionalities as GPS, Networking interfaces like Bluetooth, or other wireless channels. Observation and Sensor both are used to model each sensed environmental context attribute. Sensor Class details the characteristics of the used sensor while Observation describes the sensed information. In the next subsection we give details about the use of these main classes in the description of a living lab context.

4.1 Modeling of Living Labs's Users

In the first stage the core context ontology of our model was extended by the different RLLs applications designers to describe on one hand the users involved in the RLLs and on the other hand the context information being used. All RLLs users are described through the extension of Person a subclass of User Template Class.

In the Sekhukhune RLL we consider the following Person subclasses as Supplier PublicSector, FinancialServiceProvider, Customer, LogisticPartner, Infopreneur, denoted in Table 2. Supplier class is used to describe all suppliers involved in the RLL environment, for example, farmers is modeled using SameAs OWL Constructor. In the Soria RLL we have denoted three user's types: Emergency Staff, Tourist and Guard, while the user identity class defines the login, password, and language (English or Spanish) which are associated and linked to the user profile. In this sense, the user interfaces can be shown and loaded in different languages designed according to the user profile. In the Cudillero RLL the users have been classified into groups according to the described activities Emergencies-staff, Expert, Emergencies-staff, Buyers, Harbour-staff, Sanitary-inspectors and Fishermen. For example, ontology designer of this living lab used the class Public Sector to describe Sanitary-inspectors class using OWL SameAs constructor. The same example can be applied for the Buyers class and Customer class of the Sekhukhune RLL. The Frascati and Homokhati RLLs are quite similar to the Sekhukhune and Soria RLLs because they targets similar community composed mainly of farmers which are modeled using Farmer sub class.

Table 2: User Classes in the Different Living Labs

Person Class	Sub Classes	SameAs	Rural Living Labs			
			Col-tive Fishery	Col-tive Governance	Open Com-ties	Enterprise incubator
Supplier	--			X	X	X
Customer	Tourist, SpazaShop Owner	Buyers		X	X	X
Logistic-Partner	--					X
Infopreneur	--	Public Sector = Sanitary-inspectors	X			X
Farmer	--	Supplier		X	X	X
Fishermen	--		X			
Harbour-staff	--		X			

4.2 Modeling of Living Labs Context Information

Location information is the central concept of the context model in RLLs. For example, in Homokhati and Sekhukhune, farmer's site information, is modeled as an outdoor location with GPS coordinates having three Data Type properties corresponding to altitude, longitude and latitude coordinates. We consider here different outdoor location sub classes corresponding to farmers/suppliers workplaces (e.g., field location and the green house location) and customers locations. The same classes are used in the Frascati LL, where location of a farmer or of a plant is modeled using an outdoor location with GPS coordinates. To take an example of indoor location, Accommodation class is used to describe Guest house and Estate. It is characterized mainly by hasOwner object property to describe Owner of the Accommodation and hasRoom object property describing the different rooms as indoor locations. Data type properties like numberofAllrooms, numberOfFreeRooms and offersBreakfast provides quantitative description of the accommodation facilities.

Activity and working device are also contextual attributes characterizing RLL user context. In the Cudillero RLL the activity context involves three subclasses: Emergency (activity to resolve some alerts or emergency incidences), Fishing (daily activity of the fishermen), Inspection (activity related to the consult of data or information in the central system). In the Sekhukhune RLL, Activity has two subclasses used to describe RentAccomodation activity and BuyStock to describe sales representative activity.

The Czech RLL involves mainly contextual attributes related to the location of different sensors disseminated in the rural environment, while in the Turku RLL, the location information is important in the way it is used to localize the customer and markets by mean of GPS sensors. We add to the existent model the concept of near location to describe the nearest place to the user, see Figure 1 (a).

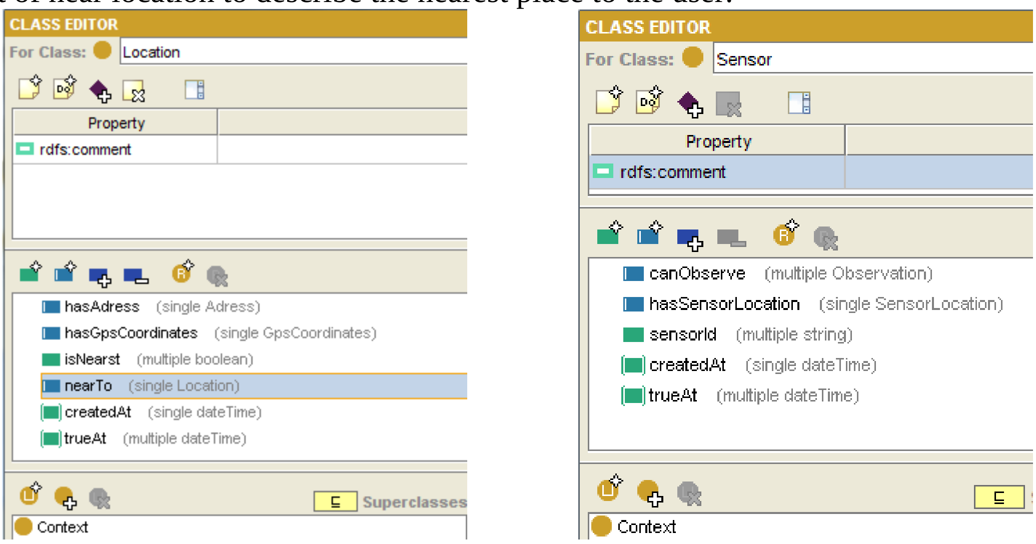
The sensed context information is modeled through subclass of Context named ObserveValues. An Observation or Sensor are both modeled as a subclass of OWL:Context Class. The children classes of Observation detail the characteristics of each observable context attribute type while the children of Sensor Class detail the characteristics of the used sensor. Each Observation sub class is characterized by an object property denoting what is the sensor being used for this observation. In this Living Lab we have identified the following list of Weather related Observation classes (Absolute Humidity, Air Temperature, Azimuth of Wind Direction, Dew Point, Light, Relative Humidity, Specific

Humidity, Wind Speed). The Sensor class is also characterized by three additional properties (Figure 1 (b)):

- canObserve: is an OWL object property which denotes the captured or the observed contextual type (any subclass of Observation).
- hasSensorLocation: is an OWL object property which denotes the location of the sensor used for the observation. This property range is a subclass of OutdoorLocation named “SensorLocation”. Adding to the inherited properties from outdoorLocation SensorLocation is characterized by forSensor object property which denotes the sensor.
- sensorID: is a string data type property which denotes the identification of the sensor in the system.

The Czech Living Lab involves mainly contextual attributes related to the observation from different sensors disseminated in the rural environment. It also involves location to denote location of sensor providing the observed of values of contextual attributes and time range of observation.

In the Turku RLL, the location information is important in the way it is used to localize the customer and markets by mean of GPS sensors. We add to the existent model the concept of near location to describe the nearest place to the user.



(a) Location Description Using Protégé *(b) Sensor Modeling Using Protégé*

Figure 1.

5. Concluding Remarks

In this paper we present the work carried out in the Collaboration@Rural project. In the framework of the project we have highlighted the importance of context-awareness in identification and development of new application and also as core strategy that can affect the existing applications. It has been put in evidence that not all the applications can become context-aware as in some cases it has no sense or it is not allowed by the specific structure of the application. From our experience we have learnt that if we want to develop really useful applications for final users, the developments must reflect well all user needs.

As a result, we have experimented and validated our skills and know-how in the field of context-awareness using a user centric strategy the project following. We have defined the context awareness information model with the aim to represent the context information of the Rural Living Labs involved in the project. The model has been applied to the Living Labs even if they are not homogeneous in their maturity. The open source Protégé tool has been recognized as the best tool to capture context information for each Living Lab: this task has been performed directly by the RLLs representatives. This full involvement of the representatives has been a positive and fruitful experience as it let us validate and proof our model and get the context information directly by RLLs.

Taking the developed model into account, a number of the context-aware components planned to be developed are identified. The identified components are currently in the development phase. For example, User profile provides users information, authorized services and preferences list, geo-location web catalogue provides information related to specific geographic location. The combination of these different component services can offer high level context awareness information related to living lab issues. For example in Cudillero living Labs in Spain, we are implementing a prototype of context aware emergency service. The fisherman can use this emergency service to send high priority messages (eg, SOS) to the other boats or to backend system according to network context. Context information includes mainly user profile, language, networking and location. The implementation of the emergency service is realized using a BPEL based orchestration of the component services available in the living services bus platform.

Acknowledgement

This work has been partly funded by the European Commission through IST Project *Collaboration@Rural: A Collaborative Platform for Working and Living in Rural Areas* (No. FP6-2005-IST-5-03492). The authors acknowledge the Commission for their support. They also acknowledge our gratitude and appreciation to all the C@R project partners for their contribution during the development of various ideas and concepts presented in this paper.

References

[1] Schilit, B., Theimer M. Disseminating Active Map Information to Mobile Hosts. IEEE Network, 8(5), pages 22-32, 1994.
[2] Ryan, N. S., Pascoe, J., Morse, D. R. Enhanced Reality Fieldwork: the Context-aware Archaeological Assistant. Computer Applications in Archaeology, 1997.
[3] Dey, A.K. Context-Aware Computing: The CyberDesk Project. In Proceedings of AAAI 1998 Spring Symposium on Intelligent Environments (AAAI '98 IE), pp. 51-54, March 23-25 1998.
[4] Dey, A.K., Abowd, G.D., Wood, A. CyberDesk: A Framework for Providing Self–Integrating Context–Aware Services. Knowledge Based Systems, 11(1):3-13, September 1998.
[5] Abowd, G.D., Dey, A.K., Brown P.J., Davies, N., Smith, M., Steggles, P. Towards a Better Understanding of Context and Context-Awareness. In Proceedings of the First International Symposium on Handheld and Ubiquitous Computing (HUC 99), pp. 304-307, September 27-29 1999.
[6] Want, R., Hopper, A., Falcao, V., Gibbons, J. The Active Badge Location System. ACM Transactions on Information Systems, 10(1), pp. 91-102, 1992.
[7] Hull, R., Neaves, P., Bedford-Roberts, J. Towards Situated Computing. In Proceedings of the First International Symposium on Wearable Computers (ISWC 97), pp. 146-153, October 13-14 1997.
[8] Salber, D., Dey, A.K., Abowd, G.D. Ubiquitous Computing: Defining an HCI Research Agenda for an Emerging Interaction Paradigm: Technical Report GIT-GVU-98-01, Georgia Institute of Technology, February 1998.
[9] Brown, P. J. Triggering Information by Context. Personal Technologies, 2(1):1-9, September 1998.
[10] Fickas, S., Kortuem, G., Segall, Z. Software Organization for Dynamic and Adaptable Wearable Systems. In Proceedings of the First International Symposium on Wearable Computers (ISWC 97), pp. 56-63, October 12-14 1997.
[11] Chen, G., Kotz, D. A Survey of Context-Aware Mobile Computing. Technical Report TR2000-381, Dartmouth College, November 2000.
[12] Dean, M., Schreiber, G., OWL Web Ontology Language Reference, W3C Recommendation 10 February 2004, http://www.w3.org/TR/owl-ref/
[13] Nardi, D., Brachman, R.J., An Introduction to Description Logics. Description Logic Handbook, Cambridge University Press, pages 5-44, 2002.
[14] McGuinness, D.L., Fikes.,R, Hendler,J., Stein,L.A., DAML+OIL: An Ontology Language for the Semantic Web, IEEE Intelligent Systems, 17 (5): 72-80, September/October 2002.
[15] Miller,L., RDF Query by Example, 2002-04-04, http://ilrt.org/discovery/2002/04/query/