Expert Systems with Applications 40 (2013) 4446–4462

Contents lists available at SciVerse ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Adapting groupware systems to changes in the collaborator's context of use

Dominique Decouchant [a,b,*], Sonia Mendoza [c], Gabriela Sánchez [c], José Rodríguez [c]

[a] C.N.R.S. – Laboratoire LIG, 681 Rue de la Passerelle, 38400 St Martin d'Hères, France
[b] Department of Information Technologies, UAM-Cuajimalpa, Av. Constituyentes 1000, Lomas Altas, Del. Miguel Hidalgo, Mexico, D.F. 11950, Mexico
[c] Department of Computer Science, CINVESTAV-IPN, Instituto Politécnico Nacional 2508, San Pedro Zacatenco, Del. Gustavo A. Madero, Mexico, D.F. 07360, Mexico

ARTICLE INFO

Keywords:
Adaptive groupware systems
Context of use
Dynamic collaborative environments
Context-aware computing
Contextual variables

ABSTRACT

Owing to the dynamic nature of collaborative environments, the software intended to support collaborative work should adapt itself to the different situations that may occur. This requirement is related to the concept of "context of use", which has been considered as an important aspect in the design of interactive systems. Nevertheless, two main problems about this concept have been identified by current research in context-aware computing: (1) most of the studies have mainly focused on the context of a single user, so the context of multiple users involved in a common endeavor remains little explored, and (2) adaptability in context-aware systems generally takes into account a reduced number of contextual variables (mainly the user's location and platform). In this paper, we firstly re-conceptualize the notion of "context of use", in order to consider the main characteristics of collaborative environments. Based on this new notion, we then design and implement a framework that allows application developers to specify the adaptability of groupware systems in terms of the state of activities, roles, collaborators' location, available resources, and other typical variables of working groups. This framework has been generalized from scenarios that highlight dynamic situations presented in real collaborative settings. Finally, we validate our proposal by a set of applications that are able to adapt their user interface and functionality, when significant changes are produced in the environment, the working group, and/or the used devices.

1. Introduction

Human behavior adapts itself to the different situations of daily life, e.g., people hurry when being late for a meeting, people raise their voice to be heard in noisy environments, or people avoid distractions when having much work to do. On the contrary, most of the computer systems do not adapt themselves to the user's context, and the rest does it in an elementary way, e.g., some characteristics of the smartphones can be modified, depending on the user's active profile.

Many research works are oriented to define, model, and detect changes in the context of use of interactive systems. However most of them focus on single user systems that lay emphasis on the same contextual variables, such as the platform (more specifically the size of the display screen) and the user's current physical location. Examples of this type of systems are FlexClock (Grolaux, Van Roy, & Vanderdonckt, 2002) and UbiCicero (Ghiani, Paterno, Santoro, & Spano, 2009).

Brézillon et al. point out that current approaches to context-aware computing seldom considers practical aspects of the context of use in real-life applications, such as collaborative work (Brézillon, Borges, Pino, & Pomerol, 2008). They stress that almost all efforts are centered on some physical elements of the context of use, such as time, weather, and other information easily gathered through sensors and directly used by applications. Moreover, they conclude that most of the studies generally consider a single user's context, whereas a collaborative group's context remains a little explored subject.

Collaborative environments involve not only people with different background and skills, but also several communication, coordination, and production mechanisms, which are used depending on the users' location, activities, and available resources. These factors make collaborative environments highly dynamic.

This paper focuses on studying the context of use, from the perspective of collaborative environments, by placing emphasis on typical variables of working groups, e.g., state of activities, roles, collaborators' location, and available resources. In order to support context-aware groupware systems, we define a new concept of "context of use" from situations that are present in real collaborative settings. Based on this concept, a new framework has been designed and implemented, in order to facilitate the development of adaptive groupware systems, which are able to support changes in both physical and logical variables of collaborative work environments.

* Corresponding author at: Department of Information Technologies, UAM-Cuajimalpa, Av. Constituyentes 1000, Lomas Altas, Del. Miguel Hidalgo, Mexico, D.F. 11950, Mexico. Tel.: +52 5591776650x6922; fax: +52 5591776650x6935.
E-mail address: decouchant@correo.cua.uam.mx (D. Decouchant).

0957-4174/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.eswa.2013.01.043

In particular, we intend to answer the following questions: Do users take advantages of groupware systems when considering their context of use? How to model changes in the context of use of such systems? We assert the answer to the first question is affirmative, as shown by the following scenario:

"A group of collaborators works on a project consisting in designing the logo for a brand new product that a client is about to release to the market. They are in a meeting room. At a given moment, a person enters it. The groupware system used to achieve their goal detects that the person who just entered is external to the group. Thus, it takes care of hiding or altering sensitive information, e.g., product name, image, and logo. When the external person leaves the meeting room, the hidden information is displayed again".

The previous scenario shows that a contextual support increases the system capabilities, since users are no longer responsible for concealing sensitive information from external persons, but this action is made automatically.

This paper is organized as follows. After presenting related work (Section 2), we propose a new notion of "context of use" and a development framework for adaptive groupware systems. We firstly express the proposed notion in terms of UML class diagrams, since it relies on some design patterns (Section 3). We then describe the layers of the proposed framework, as well as an example of information flow through these layers, and the main mechanisms for processing contextual variables (Section 4). These proposals are illustrated and validated by means of some scenarios that involve the design and implementation of an adaptive shared workspace. This last one is composed of several groupware applications that are able to work together and to adapt themselves to contextual changes (Section 5). Finally, we concludes this work and propose some future extensions (Section 6).

2. Related work

Context aware systems can adapt their operation to changes in the context of use, without needing the explicit intervention of the user, in order to increase their usability and effectiveness (Baldauf, Dustdar, & Rosenberg, 2007). In this section, we firstly describe relevant context-aware systems and then provide a comparative analysis of them.

Call Forwarding (Want, Hopper, Falcao, & Gibbons, 1992) provides a receptionist with information about the staff's location, in order to facilitate the task of forwarding calls to the closest place, where their receivers are located. Information includes a person's name and location, the closest extension, and the probability of finding him/her in such a location.

UbiDraw (Vanderdonckt & Gonzalez-Calleros, 2008) is a vectorial drawing application that adapts its graphical user interface by displaying, hiding, resizing, and rearranging the toolbars and icons, according to the user's current task, the usage frequency of the toolbars, the user's preferences (Melchior, Vanderdonckt, & Van Roy, 2012), and the available display space. The state of these contextual variables is monitored, in order to display the maximum number of tools. The adaptability in UbiDraw always results from the user's initiative.

UbiCicero (Ghiani et al., 2009; Paterno & Santoro, 2012) is a location-aware museum guide that provides visitors with services and information about works of art. When a visitor enters a new room, UbiCicero provides him with a summary of the works of art exhibit at the room, and when he approaches a work of art, UbiCicero asks him whether he wants additional information or not. UbiCicero also supports multi-user interaction by means of collaborative games. In their mobile devices, visitors can see the map of each room, where the works of art visited by each team are shown. In public displays located in strategic parts, visitors can see the map of the whole museum with each player's location and each team's visited works of art.

Table 1
Contextual variables considered by the analyzed systems.

System	Contextual variables
Call Forwarding	Staff's location
UbiDraw	User's current task, usage frequency of tools, and window size
UbiCicero	Location of works of art and visitors' location
Heating control	Platform
Conference Assistant	Users' preferences and location, and details of presentations
Nokia Situations	Time, day, GPS location, and availability of WiFi networks

The home heating control system from the French Electricity Company (Coutaz, 2010) allows the user to control the heating of several rooms of his house. This application adapts its user interface to the screen size: (1) in a PC, the temperature of all rooms is simultaneously shown; (2) in a personal digital assistant (PDA), the temperature of each room is displayed on a different tab; and (3) in a mobile phone, the graphical user interface is completely remodeled towards a textual form.

Conference Assistant (Dey, Salber, Abowd, & Futakawa, 1999) is an application that provides people attending a conference with useful information. Particularly, it displays the conference activities and timetable, and it highlights those activities that can be of interest to each user, depending on his preferences. When a user enters a room, Conference Assistant shows him on his laptop, the speaker's name and biography, as well as the lecture title. Each user is also notified of his nearby colleagues' current activities.

Nokia Situations[1] is a mobile application that adapts the behavior of the user's phone to the context, which is detected depending on the situations specified by the user as rules based on the phone internal sensors. A situation can be defined according to the day, the time, the GPS location, and the availability of WiFi networks or Bluetooth devices. Once the situations have been defined, the user configures the desired behavior of his mobile phone.

The analyzed systems are compared, according to the following dimensions: (1) single-entity systems or multi-entity systems, (2) contextual variables, (3) underlying frameworks or architectures, and (4) the adaptation type.

Single-entity systems only consider the context of one entity, e.g., a user or a device, whereas multi-entity systems simultaneously take into account the context of several entities (Olivares, Mendoza, & Luca, 2011), e.g., the staff of an organization. Hence, Call Forwarding, UbiCicero, and Conference Assistant belong to the multi-entity category, since they consider multiple users' location and the state of other entities, e.g., the works of art visited in a museum or the activities available at a conference. UbiDraw is a single-entity system, since it only employs the user's state expressed by his current task and the usage frequency of tools. The heating control system is also single-entity, because it just considers the platform state, in particular the screen size. Although Nokia Situations uses multiple contextual variables, all refer to one mobile phone, thus it is also a single-entity system.

Table 1 shows the contextual variables considered by the studied systems to adapt themselves. Most of them use physical variables, e.g., location and time. Only a few systems, such as Conference Assistant and UbiDraw, take into account logical variables, e.g., the users' preferences and the usage frequency of tools, respectively.

[1] <http://betalabs.nokia.com/apps/nokia-situations>