Future Generation Computer Systems 27 (2011) 109–118

Contents lists available at ScienceDirect

Future Generation Computer Systems

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

Towards a reference architecture for the design of mobile shared workspaces

Juan Rodríguez-Covili [a], Sergio F. Ochoa [a,*], José A. Pino [a], Valeria Herskovic [a], Jesus Favela [b], David Mejía [b], Alberto L. Morán [b,c]

[a] Department of Computer Science, Universidad de Chile, Chile
[b] Department of Computer Science, CICESE, Baja California, Mexico
[c] Faculty of Science, UABC, Baja California, Mexico

ARTICLE INFO

Article history:
Received 3 November 2009
Received in revised form
11 May 2010
Accepted 28 May 2010
Available online 8 June 2010

Keywords:
Mobile shared workspace
Design reuse
Mobile collaboration
Nomadic work

ABSTRACT

Software design is a complex and expensive socio-technical process, for which reuse is an increasingly important concern. This article presents an abstract structure capturing the essence of a mobile shared workspace (MSW), and how a particular MSW can be obtained by instantiating such a structure. It is intended to assist designers of mobile groupware tools to conceive new products reusing design knowledge proved successful in previous solutions. The article also illustrates how the abstract structure can be instantiated to obtain MSWs that support construction inspection activities, hospital work and urban emergency responses. This abstract structure can be evolved in order to make it a reference architecture for mobile shared workspaces.

1. Introduction

Design is a complex and expensive socio-technical activity for the development of a variety of artifacts, such as software systems or mechanical parts [1–3]. Provided the complexity of such a process, the reuse of design knowledge is recognized as a vehicle to reduce the costs and difficulties of creating new design solutions. A well-known example of design reuse is the utilization of patterns [4–6]. These abstract structures capture the essence of a design solution and allow inexperienced developers to design as expert ones.

Various abstract designs must be evaluated to determine whether they apply to the problem at hand and its specific context [7]. This process involves an important effort to compare the three main variables involved in design reuse: (1) the problem to be solved, (2) the context in which the problem arises and (3) the proposed abstract design solution. The result of this analysis will determine if it is worth using the proposed design abstraction.

The definition of reusable designs represents a challenge in novel application scenarios, such as computer supported mobile collaborative work, because the knowledge for such application domains is just being built. Herskovic et al. have recognized and discussed the difficulties involved in the design of mobile shared workspaces (MSW), which support mobile collaborative work [8]. However, there is no doubt that reusable designs of MSW will help in increasing the success of these software projects while reducing development and maintenance costs.

Mobile shared workspaces provide the knowledge worker with a partial view of the office (i.e. information and services) through his/her mobile computing device. These systems allow users to collaborate with colleagues or software systems in an ad hoc way (like a plug & play mechanism), adapting themselves according to the interaction context [9]. The design of such collaborative tools has been recognized as a relevant construct for the development of mobile multi-synchronous software applications [10–14]. After developing and testing MSWs for several application scenarios [15,12,16–19,14], the authors have identified commonalities that were incorporated into an abstract design solution. This abstract solution can assist the design of MSWs to support specific tasks, in several application domains.

The paper presents an abstract structure capturing the essence of an MSW design, and it shows how such a structure can be instantiated to obtain particular workspaces. The article describes instantiations of the proposed design to support mobile work in three application scenarios: construction inspections, hospital work and urban emergency responses.

The following section describes and compares mobile work in these scenarios. Section 3 presents related work. Section 4 introduces the proposed MSW abstract structure. Section 5 describes three particular workspaces that were obtained by instantiating the proposed abstract structure. Finally, Section 6 presents the conclusions and future work.

* Corresponding author.
E-mail addresses: jrodrigu@dcc.uchile.cl (J. Rodríguez-Covili), sochoa@dcc.uchile.cl (S.F. Ochoa), jpino@dcc.uchile.cl (J.A. Pino), vherskov@dcc.uchile.cl (V. Herskovic), favela@cicese.mx (J. Favela), mejiam@cicese.mx (D. Mejía), alberto_moran@uabc.mx (A.L. Morán).

0167-739X/$ – see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.future.2010.05.014

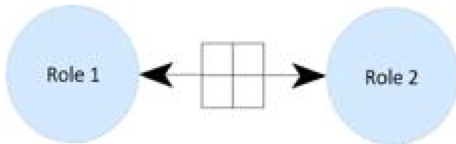
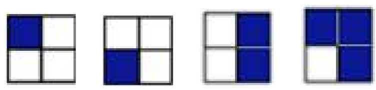


Fig. 1. Characterization of the interactions between two roles.



(a) Group synchronous work. (b) Group asynchronous work. (c) Autonomous work. (d) Multi-synchronous work.

Fig. 2. Representation of the mobile work styles using the MCM language.

Fig. 3. MSW for construction inspections.

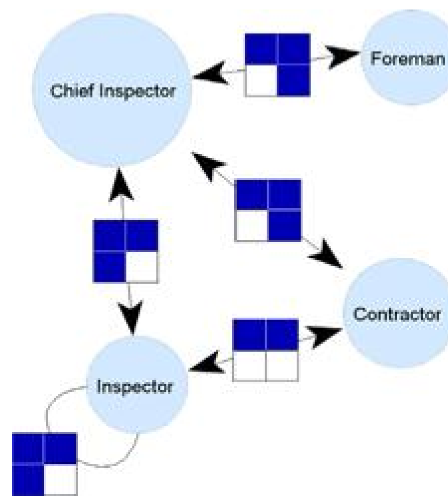


Fig. 4. Mobile collaboration in construction inspections.

2. Work scenario characterization

Mobile work can be classified into four types or styles depending on the way in which nomadic users interact: *synchronous*, *asynchronous, autonomous* and *multi-synchronous* [20]. Each style imposes particular interaction requirements to the MSW designed to support it. The interactions involved in each work style can be represented using the Mobile Collaboration Modeling (MCM) language [21]. The language represents roles and interactions between them. Roles are depicted as graph nodes, and the interactions as labeled edges (Fig. 1). Each interaction type between two roles, which is potentially present in a mobile work, is represented as a label on the edge (i.e. the square in Fig. 1).

In this case, we use the MCM language to represent the four styles of work nomadic workers use. Synchronous work is only possible when collaboration services and connectivity are always available for the participants. Synchronous work between two roles is represented with an edge having a colored upper left quarter on its label (Fig. 2(a)).

Users can perform asynchronous work using an interim component as a synchronization point (i.e. a server). The edge label in this case shows a colored lower left quarter (Fig. 2(b)).

Nomadic workers do not interact with other nodes (such as peers or servers) during autonomous work. They just use local data and services to perform the activity. Autonomous work is represented with an edge having one or two colored right quarters on its label, depending on the simultaneity of the activities carried out by the autonomous workers. Fig. 2(c) shows one possibility of autonomous work.

Finally, multi-synchronous work can be divided in a sequence of several autonomous and group activities. This style can be represented with an edge that includes in its label any combination of the three previous work styles. For example, Fig. 2(d) represents an activity involving autonomous work and also synchronous interactions between two participants. It could represent, for example, the activity carried out by a nurse during a ward round in a hospital. Typically, she inspects patients, delivers medicine and records related information on the patients' health record (i.e. she performs autonomous work). However, she needs to synchronize the patients' health information before and after each round (i.e. she performs synchronous interactions with a server or other nurses).

Multi-synchronous (or loosely coupled) work is present in most mobile collaboration scenarios [13]. These scenarios usually share a set of common requirements [8]. For each of these scenarios, we define a graph using the MCM language, which helps visually represent the involved roles and the type of interaction that takes place between them. All the interaction requirements involved in a mobile collaborative process can be identified by just analyzing the MCM graph representing such activity. It helps deal with the problem of identifying hidden requirements that usually appears in the development of mobile shared workspaces [8].

The following subsections describe and compare the characteristics of mobile work in three different application domains: construction inspections, healthcare, and emergency responses.

2.1. Construction inspections

Typically, construction projects involve a main contractor and several subcontracted companies. These companies are in charge of a specific part of the project (i.e. infrastructure, electrical network, gas or painting). Nomadic workers (inspectors) carry out the inspection process in teams to identify contingency issues (problems) in the infrastructure. Each inspection team is composed of a coordinator (chief inspector) and several inspectors. All inspectors must report results to the team coordinator, who is in charge of unifying and reporting the contingency issues to the main contractor's headquarters. The coordinator must also report the results to the foreman of the construction site as a way to keep a record of the contingency issues.

The authors have proposed to use an MSW running on tablet PCs to support this loosely coupled work (Fig. 3) [19]. This application is used by inspectors not only to conduct the reviewing process but also to interact among them and with the foreman/main contractor. Fig. 4 shows the interaction graph of these actors. The graph was specified using the MCM language.

Analyzing the graph it is possible to see that services supporting synchronous interactions among mobile users are mandatory (e.g. data synchronization or users location); however, most roles work in an autonomous way. The graph also indicates that just inspectors require remote access to shared resources. In addition, when a mobile worker decides to interact with another actor of this scenario, s/he must be able to do it. This is because these interactions involve co-located (or physically near) collaborators. Nevertheless, the chief inspector requires additional interaction mechanisms because s/he needs to interact with remote persons and actors working in another work shift. A characterization of the work context, the problem to be solved and the requirements for the solution mentioned above are presented below.

Work context. This work scenario involves several geographically distributed inspectors with the following needs: (a) they jointly work on diagnosing the project resources, (b) they need to know the other inspectors' locations and their current activities to collaborate on-demand, (c) they are not always able to use a fixed communication infrastructure and (d) some of them need to share information with other actors, e.g. with inspectors, the main contractor or the foreman.

Problem to be addressed. Inspectors need support for their interactions as represented in Fig. 4. Such interactions involve sharing messages and data resources. Each mobile worker does not know where and when such interactions will take place; therefore s/he cannot assume that specific resources will be available in the physical collaboration environment when s/he requires them, for example, wireless communication support.

Solution. A possible solution is to have a personal workspace able to interact with others on-demand. Such interactions must be supported regardless of the physical location of the people involved. These work conditions impose a number of requirements for the design of MSWs in terms of interaction support. The most important requirements are the following ones:

– *Autonomy of data and services*. Nomadic users must be able to perform the inspection even if they are isolated in the construction site. The information and services required to complete the task must be available in the mobile worker's computing device.
– *Ad hoc communication*. If two inspectors meet in the construction site and decide to collaborate, the MSW must provide mechanisms to support their interactions regardless of the workers' physical location.
– *Awareness of user availability/location*. Since loosely coupled work involves on-demand collaboration, when a mobile worker wants to contact a teammate he needs to know his/her location and/or availability. In fact, availability could be inferred from the teammate's location.
– *Messaging*. This service is useful for short and simple interactions between mobile workers. For example, asking an inspector for an urgent meeting in the foreman's cabin. Typically, messaging is used to promote collaborative work sessions.
– *Information sharing*. Inspectors record contingencies in their local workspaces. They have to integrate that information after the reviewing process in order to get a complete view of the project's state. Depending on the type of representation used for this information (i.e. if the information is reconcilable or irreconcilable [17]), the sharing process will require (1) just file transfer (in case of irreconcilable information) or (2) file transfer and data synchronization (in any other case).

2.2. Healthcare services

Healthcare services, particularly those provided in hospitals, involve collaborating nomadic workers performing several roles: doctors (e.g. physicians and therapists), medical residents, and nurses. They carry out periodic inspection rounds to monitor the patients' health condition. The medical personnel report the information to the doctor in charge of a work shift and also to the hospital central services at the end of a ward round. Such information is then used to evaluate each patient's condition and also the global situation at the hospital.

Fig. 5. MSW supporting health service delivery.

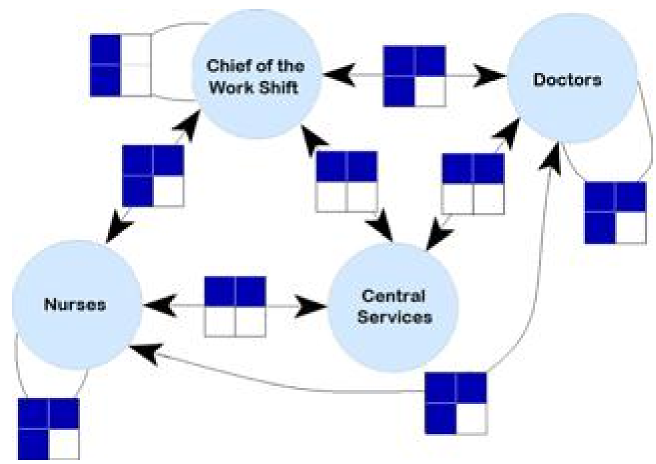


Fig. 6. Mobile collaboration in hospital work.

For example, a mobile worker (e.g. a nurse) may detect a problem with a patient during a round. This event triggers a set of actions, for example locating the physician in charge of that patient, retrieving the patient's medical record, or calling a meeting with a specialist to analyze the case.

A workplace study conducted by the authors in a public hospital showed that the staff spends 53% of their time out of their base location [22]. While on the move, other staff often locates and contacts them to request shared information (e.g. laboratory results) or to discuss issues related to patients' assessment.

In case of an emergency, it is important to notify the appropriate supporting personnel in order to provide a response according to the situation. Notification must be fast and effective because the patient's life could depend on it.

An MSW running on tablet PCs or PDAs can be used by doctors, therapists and nurses to examine and update a patient's medical record (Fig. 5) [14]. These tools can also be used to deliver messages/alarms, locate medical personnel and share patient information.

Fig. 6 shows the interaction graph typical of hospital work. There, we can see that most of the work is performed in a synchronous way; although the participants perform mainly autonomous work. Therefore services for on-demand collaboration are mandatory. Furthermore, the hospital personnel must count on an intermediary actor (e.g. a shared space or server) in order to be able to contact persons who are temporarily unavailable. For example, a doctor may be unavailable while s/he is in a surgery room. Next, we present the work context, the problem to be addressed and the requirements for the solution mentioned above.

Work context. The work conducted by healthcare professionals involves geographically distributed personnel with the following characteristics: (a) they jointly work on diagnosing and taking care of patients, (b) they need to know the location and current activities of their colleagues because they collaborate on-demand, (c) they are not always able to use a fixed communication infrastructure and (d) they need to share information with other nomadic workers, hospital managers and their computer center. Unlike the previous scenario, hospital work involves occasional time-critical activities (e.g. during emergencies or surgical interventions).

Problem to be addressed. Patients are usually highly distributed over different locations (rooms, floors and buildings). Therefore there is an ample space where people collaborate on-demand. Like in the previous case, mobile workers have uncertainty about where and when such interactions will take place. However, they must be able to carry out the interactions represented in Fig. 6, independently of their physical location. Some of these interactions can be time-critical.

Solution. An MSW for medical personnel can assist mobile workers in a hospital. The most important requirements associated to an MSW for this setting are the following ones:

- *Autonomy of data and services*. It is not possible to assume accessibility of the information in the hospital systems or availability of wireless networks during time-critical activities. Therefore, the MSW must be as autonomous in terms of data and services as possible.
- *Ad hoc communication*. The MSW requires an ad hoc communication service since access to wireless communication cannot be ensured at all times and everywhere. This increases the capability to locate and interact with teammates and shared resources.
- *Awareness of user availability/location*. These are two of the most frequently used services. Therefore counting on them makes a difference in the resulting mobile work. Furthermore, these services become critical during emergencies.
- *Messaging*. This is a basic service, which is required to locate and coordinate mobile workers. It is also required to deliver alarms and support information sharing.
- *Information sharing*. This service simplifies the process of sharing documents (e.g. medical records) and reporting results of several regular activities, such as patients' inspections or nurses' rounds. Depending on the type of the data involved (e.g. text documents, images, or structured files), there is a more appropriate mechanism to share it [17]. Usually, these mechanisms involve file transfer and/or data synchronization.

2.3. Routine urban emergencies

Typically, firefighters are at the forefront of responses to urban emergencies, such as fires, car accidents, gas leakages, collapses, and chemical/water spills. The time required to take control of an emergency has a direct consequence on injuries to citizens and damages to the civil infrastructure.

Whenever an alarm center receives an emergency call, it typically delivers two or more fire trucks (belonging to different companies) to the site of the emergency, in order to ensure the resilience of the response process. Additionally, it could ask medical personnel or police for assistance.

Once firefighters arrive at the emergency site, they trigger the response process. This process requires coordinating first response activities in order to reduce the time needed to take control of the emergency. An Incident Commander is in charge of coordinating the efforts in the field. However this person is typically only able to make the macro-decisions, due to the typical limitations of radio channels [23]. Several decisions are also made in parallel by firemen who need to deal with time-critical situations; e.g., (1) decide where to park a fire truck to get fast and safe access to the building's water network during a fire, or (2) determine which is the best route to rescue a victim.

Fig. 7. MSW to support emergency responses.

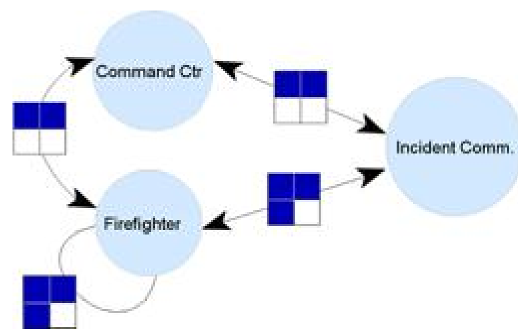


Fig. 8. Mobile collaboration during emergencies.

Making on-time and effective decisions requires on-demand access to key information and sometimes to validate such decisions with a partner. Decisions that can affect other first response teams must be communicated as a sort of awareness mechanism. Alarm delivery is also an important service that must always be available when responding to an emergency.

The authors have proposed an MSW running on a PDA or smartphone as a complement to the current radio systems used by firemen (Fig. 7). This workspace is used just by decision makers in the field and it allows them to retrieve important information from the command center or other decision makers in the field; e.g., building blueprints, maps of the affected area, evacuation routes, or location of other fire trucks.

The MSW can also be used to deliver messages/alarms, to locate key personnel (e.g. a civil engineer who may evaluate the affected physical infrastructure) or special resources (e.g. a concrete breaker).

Fig. 8 shows the interaction graph for a typical urban emergency. Such graph indicates the three types of actors require services to conduct synchronous interactions (e.g. notifications and file transfer). However, the incident commander and firefighters can get isolated if there are communication problems in the emergency place. In such case, they must count on a mobile application that embeds the data and services required to perform autonomous work. Next, we present the work context, the problem to be addressed and the requirements for the previously mentioned solution.

Work context. The emergency response process involves geographically distributed first responders with the following characteristics: (a) they cooperate to mitigate the emergency situation and try to reduce the damage to human lives and property, (b) they need to know the location and current activities of their colleagues because they need to collaborate on-demand, (c) they also need to locate resources available for the emergency mitigation (e.g. a chain saw or a concrete breaker), (d) they are not always able to use a fixed communication infrastructure and (e) they need to share information with other first responders and also with the command center. In this work scenario most of the activities are time-critical.

Problem to be addressed. Firemen must be able to carry out the interactions represented in Fig. 8, independently of the physical characteristics of the emergency site. Their life and perhaps other persons' lives could depend on it. Therefore, firefighters have to assume that no communication support will be provided in the site and the interaction capabilities must be embedded in the MSW.

Solution. An MSW has shown to be an appropriate tool to support the decision-making process in the field during emergencies [16,23]. The most important requirements associated to MSW dealing with this activity are the following ones:

– *Autonomy of data and services*. It is highly recommended for the MSW to be as autonomous as possible, considering the large number of decisions that are made in parallel during an emergency, and the risk represented by the dependence on centralized components.
– *Ad hoc communication*. It is important to embed ad hoc communication capabilities in each MSW since the communication media could be unavailable at a certain place or for a period of time. The communication support represents the basis for any other coordination or collaboration service; therefore it must be available when required.
– *Awareness of firemen availability/location*. The incident commander changes periodically the deployment of firefighters in the field, depending on the evolution of the emergency situation; therefore s/he must know the firemen's availability and location. These two services are critical, not only to coordinate the response process but also to protect the firefighters' lives.
– *Awareness of resource availability/location*. These services are very important as they help reduce the time required to access the response resources, which accelerate the response process.
– *Information sharing*. It is required to retrieve and report information about the emergency or the affected area (e.g. pictures, maps, or hydrant locations). Information sharing could help make fast and accurate decisions. It improves the effectiveness of the mitigation process.
– *Messaging*. This service is required by firemen mainly to deliver orders and alarms. However, the messaging services are the basis for many other services, such as file transfer or awareness of users' location.

The features common to the described mobile work activities are characteristics of loosely coupled work, and not only of a particular work scenario. Therefore, it should be possible to define an abstract MSW containing the core services to support loosely coupled work. Then, each particular MSW could be designed as an instance of the abstract representation. This design reuse would help reduce the cost and complexity of the development of appropriate solutions to these design problems; and it might help improve the quality of the final product.

3. Related work

Various research initiatives have reported the implementation and usage of mobile collaborative applications in several work scenarios, such as education, healthcare, business, productive processes and disaster relief [24,25,16,22,13,26]. Unfortunately, they do not describe the design of strategies used to support loosely coupled work [27]. Thus, the potential design solutions behind such applications cannot be reused in future developments.

Some of these initiatives model particular design issues, such as awareness [28,29], work context [30], computing support [15,31] or data synchronization [32,17]. Others focus on the communication or coordination support required to carry out loosely coupled work [27,33]. However, these papers do not report on the design decisions required by the MSW to support loosely coupled (or multi-synchronous) work.

A notable exception is the work by Su and Mark [34], whom defined a general model to describe the behavior of nomadic workers. Their model involves three main concepts: assemblage of actors, searching for resources and integration with others. These components consider the communication and coordination capabilities, and also on-demand interactions. However, the model does not consider autonomous work, which is a significant element of loosely coupled activities.

The authors have already proposed a couple of comprehensive solutions for designing MSW [12,18]; however, this proposal is a step forward towards the definition of a domain-specific architecture for MSWs supporting loosely coupled work.

Regarding reuse of design knowledge in collaborative systems, several strategies have been proposed. However, the most successful approaches include groupware design patterns, pattern languages and frameworks.

A groupware pattern is a structured description of a solution to a recurrent problem in a particular groupware context [35]. These patterns use specific examples, state the groupware problem that they address, and deliberately scope their context of application. However, taken in isolation, patterns only represent unrelated good ideas; thus, a pattern language is required to provide coherent support for design generation. A pattern language provides a taxonomy to help designers to find patterns (and related reusable designs), evaluate the problem from various perspectives, and develop new solutions [36].

The proposed MSW abstract structure is similar in spirit to a pattern; because it describes in abstract form and by means of examples, a solution to the recurrent problem of designing mobile workspaces for loosely coupled work. Also, it explicitly establishes the context of use, the problem to be addressed, and a proposed solution. Furthermore, the proposed abstract structure is also similar to a pattern language as it aggregates an ensemble of proved solutions to enable designers to select the most adequate configuration of communication, coordination and interaction space. The main difference however, is that our proposal provides these elements in a more tightly integrated manner that makes it more specific to the application domain (i.e. mobile workspaces for loosely coupled work).

Groupware frameworks provide a structure to organize thinking about particular aspects of support, a vocabulary for analyzing activities during collaboration and for comparing solutions on these aspects, and a set of starting points from which initial solutions could be further developed and refined [37]. There are high-level frameworks that describe particular aspects of groupware support at a conceptual level, such as component elements, mechanisms used to provide actual support and its uses for collaboration (WHAT to provide, HOW to provide it, and WHEN to provide it). Also, there are low-level frameworks that include implementations on specific languages and object hierarchies to facilitate development (e.g. DOORS [38] and Manifold Framework [39]), and design constructs such as multi-level architectures that separate concerns to deal with specific aspects of the domain one at a time (e.g. presentation layer, domain logic, and collaboration logic). Despite our proposal for MSWs provides a structure to organize thinking and a vocabulary for designing mobile workspaces for loosely coupled work, it cannot be used directly to evaluate specific MSWs due to the abstract nature of the model; thus it lacks the comprehensiveness of a framework.

4. The MSW abstract structure

The abstract structure of an MSW is presented considering the context where an instance application can be used, and also the problem the collaborative system is able to address. Thus, it is possible to know in which situations it is possible to reuse the proposed design in order to obtain particular instances of this abstract structure. The following subsections describe these three aspects of this reusable design.

4.1. Context

Most mobile workers carry out loosely coupled (or multi-synchronous) work [13]. Their style of work involves a sequence of several autonomous and on-demand collaboration activities. These activities can be synchronous or asynchronous, and they can involve three types of actors: peers, formal points and interim points. Peers are other mobile workers; formal points are centralized permanent components (e.g. a server); and interim point are centralized resources that are available at certain periods of time (e.g. the employees of a commercial office). Interim points are stationary or have micro-mobility (i.e. they move around a relatively small area, for example, within an office).

Typically the work context includes features of the physical environment where the user is located (e.g. wireless network availability or physical proximity of other users). Since peers are on the move while they carry out the assigned task, their work context is typically dynamic and unpredictable.

4.2. Problem to be addressed

Nomadic workers must be as autonomous as possible, in terms of the data and services required to perform a task while on the move. In addition, they must be as connected as possible, because loosely coupled work typically involves on-demand collaboration.

In an ideal situation, a nomad worker should be able to interact with others when required, independently of the type of collaborator (i.e. peers and/or interim/formal points) or the physical location of the involved actors.

In addition, a mobile worker usually needs to locate a particular resource or collaborator, a task that could be supported by appropriate services provided by the application.

4.3. Proposed solution

The proposed solution considers two basic components to support loosely coupled work: (1) an appropriate mobile device to carry out the specific kind of nomadic work (e.g. a smartphone or a laptop), and (2) a mobile shared workspace providing the functionality required to perform the activity and collaborate on-demand.

An MSW is a mobile collaborative system with three main components: the workspace, and communication and coordination services (Fig. 9).

The *workspace* is the portion of the office that contains the resources (i.e. data and services) required to conduct the mobile activity. Typically, this workspace must be designed to support personal work; however it also has to be able to provide access and manipulation of shared resources (e.g. a digital blueprint or a medical record). In addition, this component has to provide support for on-demand collaboration. The collaboration

Fig. 9. Abstraction of an MSW.

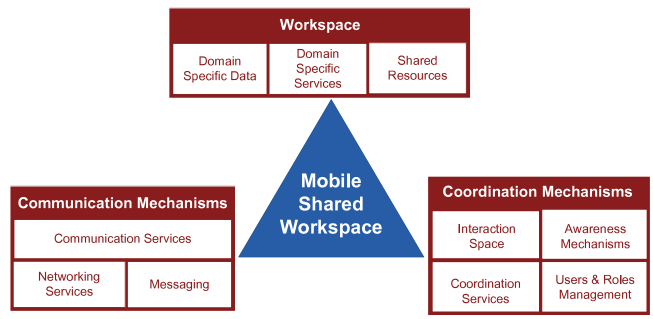


Fig. 10. Abstract structure of an MSW.

mechanisms are implemented using the communication and coordination services (Fig. 10).

The *communication* component provides the connectivity required by nomad workers. It establishes the minimal operational requirements to support computer supported loosely coupled work. Grigoras and Riordan propose a cost-effective mobile ad hoc network manager to support this type of mobile communications situations [40].

Three groups of services are involved in this component: networking, messaging and communication. Networking is in charge of keeping a user visible for the rest of the peers. Messaging provides point-to-point or group messages delivery. Communication uses the previous services to allow more complex communication services, such as peers' dynamic discovery, resources location, or alarms delivery.

The *coordination* component uses communication services to provide advanced interaction support to mobile workers. This component includes four sub-components: management of users and roles, coordination services, an interaction space, and awareness mechanisms.

User and role management is used to identify nomadic workers and assign them grants to access shared resources. The coordination services are those used to manipulate or distribute shared resources (e.g. file transfer or data synchronization).

Since the proposed abstract structure is an architectural design component, the communication and coordination services to be embedded into a MSW could be modeled using particular design patterns. Some of those communication and coordination patterns have been proposed by the authors in previous works [27,33].

The interaction space provides services to support the interaction between the three types of participants (i.e. peers, interim points or formal points). The awareness mechanism is used to know the location and availability of other participants.

Fig. 11 summarizes, through a class diagram, the abstract structure of an MSW. It is worth noting that the strength of this abstract model is its high-level descriptive and generative ability, which allows modeling both current and future MSW.

This approach can be used as a general instrument to evaluate the structure of a MSW. However, it must not be used to evaluate

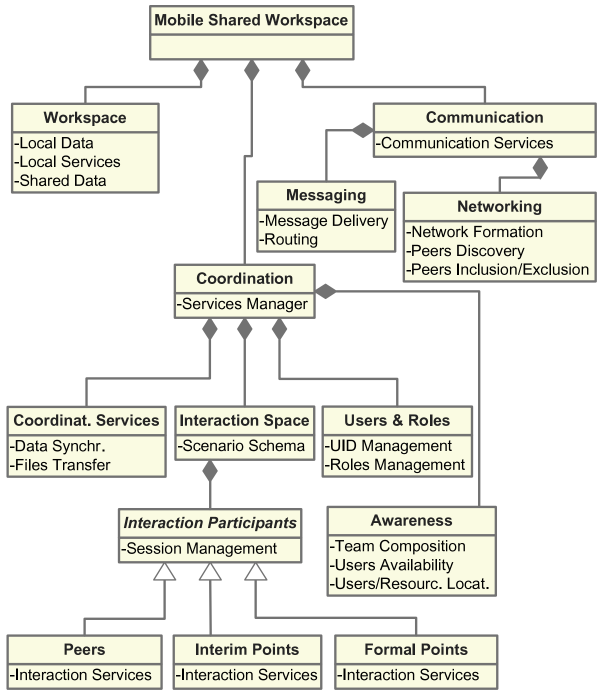


Fig. 11. Structural design of an MSW.

Fig. 12. User interface of the MSW that support construction inspections.

specific functions to be embedded in a particular MSW, because such functions depend on the results of the instantiation process performed on the proposed abstract model.

5. Instances of the proposed abstract structure

The proposed abstract structure is the result of a bottom–up process, in which the authors analyzed and conceptualized the design behind three already developed MSWs. In this section we show how to instantiate the proposed structure to design specific MSWs that support mobile work in the three described work scenarios.

5.1. MSW for construction inspections

The MSW for construction inspections allows inspectors to manage several construction projects, each one composed of several digital blueprints. The inspectors can do hand-written annotations on those maps (i.e. local resource linked to a particular blueprint) indicating contingency issues (i.e. problems). The annotations of several inspectors need to be communicated and integrated in order to diagnose the project state. Eventually, these annotations have to be removed (or marked as completed) when the corresponding subcontractor has addressed the problem.

Fig. 12 shows the main user interface of COIN (Construction Inspector) [19], which is the MSW developed by the authors to support this activity. COIN has been evaluated in a real scenario

Workspace

Domain Specific Data: *Projects, digital blueprints, annotations.*

Domain Specific Services: *Annotate blueprints, navigate the blueprints, change the current project/blueprint, show/hide annotations, etc.*

Shared Resources: *Projects, digital blueprints, annotations*

Fig. 13. Local workspace to support construction inspections.

Coordination Mechanisms

Users & Roles Management

Roles

Peers: *Inspector, coordinator.*

Interim Points: *Foreman.*

Formal Points: *Contractor.*

Users

- *Users' authentication.*
- *Management of users' grants.*

Interaction Space

Synchronous: *Inspector-Contractor, Inspector-Inspector, Inspector-Coordinator, Coordinator-Foreman, Coordinator-Contractor.*

Asynchronous: *Inspector-Contractor-Inspector, Inspector-Coordinator-Inspector, Inspector-Inspector-Coordinator.*

Autonomous: *Inspector, Coordinator.*

Awareness Mechanisms

Users' connection and location, users' availability for collaboration, users' role, reviewed areas, inspection status, annotations owner and timestamp.

Coordination Services

File transfer, on-demand information synchronization, session management.

Fig. 14. Coordination mechanisms of the MSW.

and the obtained results show the design to be appropriate in supporting the inspectors' loosely coupled work [19]. In particular, COIN has shown to reduce the effort spent in coordination activities. Experimental results indicate the inspection time using COIN is less than 10% of the time used with the paper-based process.

The architecture for this MSW can be obtained by instantiating the proposed abstract structure. The process consists of instantiating each design component based on the interaction graph (Fig. 4) and the information from construction inspection scenario. In this case, the workspace can be instantiated as shown in Fig. 13 in order to provide the autonomy of each nomad worker.

The coordination mechanisms must consider interactions among the four roles represented in the graph shown in Fig. 4. The instantiation of this component is presented in Fig. 14.

The communication component provides the basic support for interaction between roles. This component assumes the data of an inspection can be shared synchronously between two actors, or asynchronously using an interim/formal point as intermediary. Fig. 15 presents the instantiation of this component.

5.2. MSW for healthcare

Physicians and nurses will be the users of the MSW to support hospital work. The use of PDAs or smartphones is recommended since hospital work involves high mobility and a low data entry rate [31], and they have indeed been widely adopted in such settings. Fig. 16 shows the main interface of an MSW supporting medical staff during ward rounds [14].

Communication Mechanisms

Networking Services: *Peer detection, on-demand formation of mobile ad-hoc networks, peer inclusion/exclusion, connection to remote resources.*

Messaging Services: *Messages passing, unicast/multicast message delivery, routing.*

Communication services: *Synchronous/asynchronous notifications, Web services provision and consumption*

Fig. 15. Mobile connectivity component.

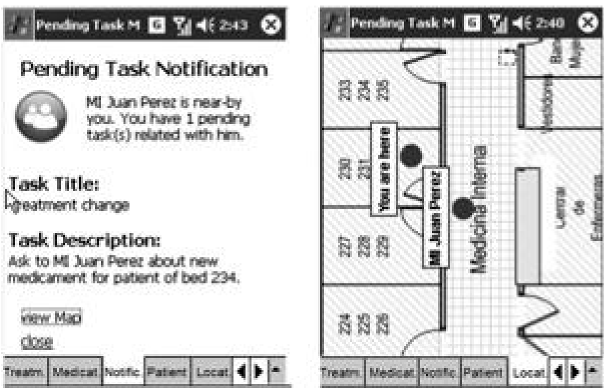


Fig. 16. MSW to support hospital work.

The menu at the bottom of the user interface shows the options that are accessible to physicians and nurses. In particular, the left panel shows a message received by a nurse, while she assists a patient, notifying her of the proximity of a colleague whom she needs to contact for information or assistance. The right panel shows the location of these two workers on the hospital map.

The workspace component of this application is in charge of supporting personal work and providing autonomy during hospital work. Fig. 17 shows an instantiation of this component for such scenario.

Fig. 18 shows the coordination mechanisms involved in the MSW. These mechanisms consider the roles involved in hospital work, and their interactions. The hospital central services act as information repository and also as a coordination element.

Similar to the previous case, mobile workers need to do both autonomous and group work. The users mainly need an updated copy of the patients' medical record and/or medication schedules for autonomous activities. On the other hand, the physicians typically require more information for group activities, such as meetings to analyze medical cases.

The instantiation of the communication component is similar to the previous case. The main differences are the nature of the information to be shared, the type of updates that can be done on such data, and the need to deliver alarms (i.e. broadcast) in case of emergencies.

5.3. MSW for urban emergency responses

Firefighters can use MSWs to support distributed decision-making and also coordination during emergency response. Fig. 19 shows three user interfaces of MobileMap [16], an MSW developed by the authors to support such activities. This application is an autonomous Geographical Information System running on a PDA or smartphone. MobileMap uses GPS to locate resources, such as fire trucks, firefighters and the emergency site.

The system allows on-demand interaction between a fireman and the command center in order to retrieve/report information about an emergency. It also provides ad hoc communication

Workspace

Domain Specific Data: *Patients census, medical records, nurse sheet, medication schedules.*

Domain Specific Services: *Update the patient health record and medication schedules.*

Shared Resources: *Patients census, medical records, nurse sheet, medication schedules, and patient studies.*

Fig. 17. Workspace supporting hospital work.

Coordination Mechanisms

Users & Roles Management

Roles

Peers: *Doctors in charge of a work shift (Chief), Doctors (physicians & and therapists), Nurses.*

Formal Points: *Hospital Central Services.*

Users

- *User authentication.*
- *Management of users' grants.*

Interaction Space

Synchronous: *Chief-Nurse, Chief-Doctor, Chief-Central Services, Chief-Chief, Doctor-Nurse, Doctor-Central Services, Doctor-Doctor, Nurse -Central Services, Nurse-Nurse.*

Asynchronous: *Chief-Central Services-Chief, Chief-Central Services-Doctor, Chief-Central Services-Nurse, Doctor-Central Services-Doctor, Doctor-Central Services-Nurse, Nurse-Central Services-Nurse.*

Autonomous: *Chief, Doctor, Nurse.*

Awareness Mechanisms

Users' connection and location, users' availability for collaboration, users' role, reviewed patients, patient's annotations (with owner and timestamp).

Coordination Services

File transfer, on-demand information synchronization (e.g. medical treatments and records, medication schedules), session management, alarms delivery.

Fig. 18. Coordination mechanisms supporting hospital work.

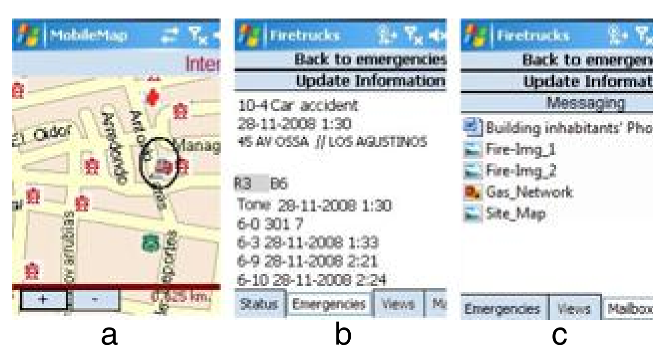


Fig. 19. MobileMap user interfaces.

and data sharing in the emergency site. It helps the incident commander, who is in charge of the emergency response process, to make and communicate her/his decisions.

Fig. 19(a) shows the current location of a fire truck that is arriving at the emergency site. The interface also indicates the location of key resources that can be used to support the response process, such as hydrants, hospitals and police stations. Fig. 19(b) presents a general characterization of the emergency situation

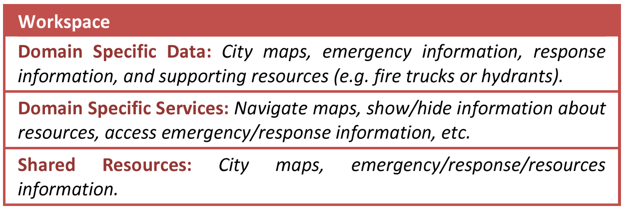


Fig. 20. Local workspace to support urban emergency responses.

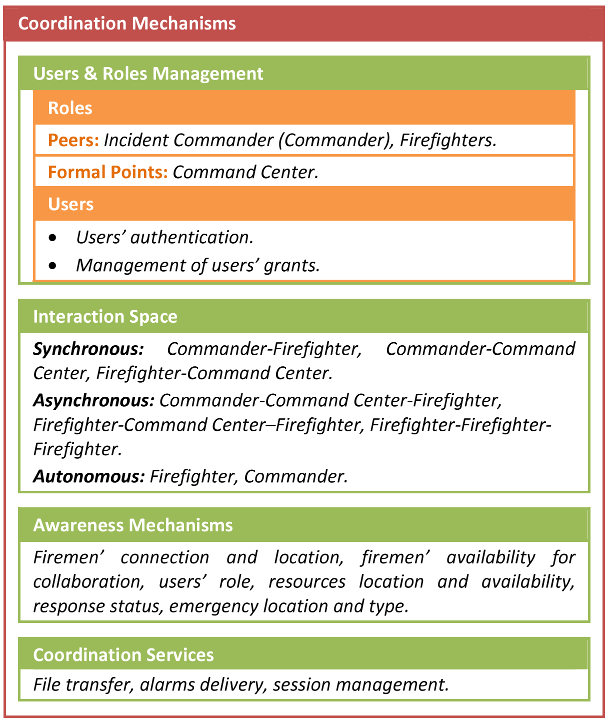


Fig. 21. Coordination mechanisms to support urban emergency responses.

to be addressed. Fig. 19(c) shows the information resources related to that emergency, which are available to firemen. This information could be useful to trigger the response process even if the firefighters have not already arrived at the emergency site. For example, in case of a flood or a fire, police officers may trigger the evacuation of the most vulnerable buildings nearby.

The workspace embedded in MobileMap provides autonomy to firemen during emergencies. Fig. 20 shows an instantiation of this component to support urban emergency responses.

The coordination mechanisms must consider interactions specified in the interaction graph for each application domain (Fig. 8). The instantiation of this component is presented in Fig. 21.

The instantiation of the communication component is similar to the previous work scenarios. In fact, this component depends mainly on the work style to be supported. Therefore it will be very similar since the three scenarios presented involve multi-synchronous work.

6. Conclusions and future work

Mobile shared workspaces can offer significant assistance to nomadic workers; both in terms of increased productivity [24,41,13,26,42], as well as in the quality of the work performed [24,25,13,14]. The development of these applications, however, can be a significant challenge and to our knowledge no previous research has been made to integrate this design knowledge to ease the development of tools aimed at supporting particular loosely coupled activities [27].

This article proposes an abstract structure that represents the essence of an MSW design. The structure was obtained after analyzing and generalizing the designs behind three already developed and tested MSWs. The results obtained from the use of these MSWs show they were successful in supporting loosely coupled work in the respective application domain.

This abstract structure can be instantiated to design particular MSW for specific work scenarios. The instantiation process is simple, facilitating its use by designers of collaborative mobile solutions. Typically, this practice also allows them to reduce project risk and complexity, produce a positive impact on the quality of the final product, and reduce the cost and time spent in the design process.

We are currently designing an MSW for a new domain, namely, the work of engineering undergraduate students. The MSW will allow the students to do various activities, such as write notes on the lecture slides, share the notes with other students, and integrate the notes of various students on the slides. The authors are using the abstract structure proposed in this article for this design activity. The future results in this application domain will allow us to have additional evidence of the usefulness of the design abstraction process.

The next step in this abstraction process is to formally define a domain-specific architecture for MSWs supporting loosely coupled work. This will considerably increase the applicability of this proposal.

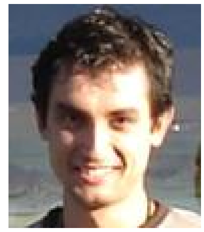
Acknowledgements

This work was partially supported by Fondecyt (Chile), grant Nos. 11060467 and 1080352, and LACCIR grant Nos. R0308LAC004 and R0308LAC005.

References

[1] L. Bucciarelli, Designing Engineers, MIT Press, Cambridge, 1994.
[2] C.M. Eckert, A.M. Maier, C. McMahon, Communication in design, in: P.J. Clarkson, C.M. Eckert (Eds.), Design Process Improvement—A Review of Current Practice, Springer, London, 2005.
[3] V. Hubka, E.W. Eder, A scientific approach to engineering design, Design Studies 8 (1987) 123–137.
[4] C. Alexander, S. Ishikawa, M. Silverstein, A Pattern Language: Towns, Buildings, Construction, Oxford University Press, New York, 1977.
[5] F. Buschmann, R. Meunier, H. Rohnert, P. Sommerlad, M. Stal, Pattern-Oriented Software Architecture, Volume 1: A System of Patterns, John Wiley & Sons, New York, 1995.
[6] E. Gamma, R. Helm, R. Johnson, J. Vlissides, Design Patterns: Elements of Reusable Object Oriented Software, Addison-Wesley, Reading, 1994.
[7] D. Roberts, R. Johnson, Patterns for Evolving Frameworks. Pattern Languages of Program Design III, Addison-Wesley, Reading, 1997.
[8] V. Herskovic, S.F. Ochoa, J.A. Pino, A. Neyem, Mobile collaborative systems: behind the user interface, Journal of Universal Computer Science, 2010 (in press).
[9] H. Schaffers, T. Brodt, M. Pallot, W. Prinz, The Future Workplace — Perspectives on Mobile and Collaborative Working, Telematica Instituut, The Netherlands, 2006.
[10] M. Divitini, B. Farshchian, H. Samser, UbiCollab: collaboration support for mobile users, in: ACM Symposium on Applied Computing, Nicosia, Cyprus, 2004, pp. 1191–1195.
[11] V. Herskovic, D.A. Mejía, J. Favela, S.F. Ochoa, J.A. Pino, A.L. Morán, Increasing opportunities for interaction in time-critical mobile collaborative settings, in: CRIWG'09 Conference, Douro, Portugal, in: Lecture Notes in Computer Science, vol. 5748, 2009, pp. 41–48.
[12] D. Mejia, A. Morán, J. Favela, Supporting informal co-located collaboration in hospital work, in: CRIWG'07 Conference, in: Lecture Notes in Computer Science, vol. 4715, 2007, pp. 255–270.
[13] D. Pinelle, C. Gutwin, A groupware design framework for loosely coupled workgroups, in: ECSCW'05 Conference, Paris, France, 2005, pp. 119–139.
[14] M. Tentori, J. Favela, Collaboration and coordination in hospital work through activity-aware computing, International Journal on Cooperative Information Systems 17 (4) (2008) 413–442.

[15] J. Favela, M. Tentori, L. Castro, V. González, E. Moran, A. Martinez-Garcia, Activity recognition for context-aware hosp. appl.: issues and opportunities for the deployment of pervasive nets, Mobile Network and Applications 12 (2–3) (2007) 155–171.
[16] A. Monares, S.F. Ochoa, J.A. Pino, V. Herskovic, A. Neyem, Mobile computing in urban emergency situations: improving the support to firefighters in the field, Expert Systems with Applications, 2010 (in press).
[17] A. Neyem, S.F. Ochoa, J.A. Pino, A strategy to share documents in MANETs using mobile devices, in: 8th IEEE Int. Conf. on Advanced Communication Technology, ICACT'06, Phoenix Park, Korea, 2006.
[18] A. Neyem, S.F. Ochoa, J.A. Pino, Integrating service-oriented mobile units to support collaboration in ad-hoc scenarios, Journal of Universal Computer Science 14 (1) (2008) 88–122.
[19] S.F. Ochoa, J.A. Pino, G. Bravo, N. Dujovne, A. Neyem, Mobile shared workspaces to support construction inspection activities, in: P. Zaraté, J.P. Belaud, G. Camilieri, F. Ravat (Eds.), Collaborative Decision Making: Perspectives and Challenges, IOS Press, Amsterdam, 2008, pp. 270–280.
[20] L. Chateigner, S. Chabridon, N. Sabri, G. Bernard, Service de reconciliation pour la synchronization de copies, in: 1st French-speaking Conference on Mobility and Ubiquity Computing, Nice, France, 2004, pp. 107–114.
[21] V. Herskovic, S.F. Ochoa, J.A. Pino, Modeling groupware for mobile collaborative work, in: CSCWD'09 Conference, Santiago, Chile, IEEE Press, 2009, pp. 384–389.
[22] E.B. Morán, M. Tentori, V.M. Gonzalez, J. Favela, A.I. Martínez-Garcia, Mobility in hospital work: towards a pervasive computing hospital environment, International Journal of Electronic Healthcare 3 (2007) 72–89.
[23] S.F. Ochoa, A. Neyem, J.A. Pino, M.R.S. Borges, Supporting group decision making and coordination in urban disasters relief efforts, Journal of Decision Systems 16 (2) (2007) 143–172.
[24] J.H.E. Andriessen, M. Vartiainen (Eds.), Mobile Virtual Work: A New Paradigm? Springer, Berlin, Heidelberg, 2006.
[25] M.C. Brugnoli, F. Davide, R. Slagter, The future of mobility and of mobile services, in: P. Cunningham, M. Cunningham (Eds.), Innovation and the Knowledge Economy: Issues, Applications, Case Studies, IOS Press, Amsterdam, 2005, pp. 1043–1055.
[26] P. Tarasewich, Designing mobile commerce applications, Communications of the ACM 46 (12) (2003) 57–60.
[27] A. Neyem, S.F. Ochoa, J.A. Pino, A patterns system to coordinate mobile collaborative applications, Group Decision and Negotiation, 2010 (in press).
[28] A. Ferscha, Workspace awareness in mobile virtual teams, in: 9th IEEE International Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises, IEEE Press, 2000, pp. 272–277.
[29] C. Gutwin, M. Roseman, S. Greenberg, A usability study of awareness widgets in a shared workspace groupware system, in: Proc. of the ACM Conference on Computer Supported Cooperative Work, 2006, pp. 258–267.
[30] R. Alarcon, L. Guerrero, S.F. Ochoa, J.A. Pino, Design of mobile collaborative applications using contextual elements, Computing and Informatics 25 (6) (2006) 469–496.
[31] L.A. Guerrero, S.F. Ochoa, J.A. Pino, C. Collazos, Selecting devices to support mobile collaboration, Group Decision and Negotiation 15 (3) (2006) 243–271.
[32] F. Hupfeld, M. Gordon, Using distributed consistent branching for efficient reconciliation of mobile workspaces, in: Collaborative Computing: Networking, Appl. and Worksharing Conference, IEEE Press, 2006, pp. 1–9.
[33] A. Neyem, S.F. Ochoa, J.A. Pino, Communication patterns to support mobile collaboration, in: CRIWG'09 Conference, Douro, Portugal, in: Lecture Notes in Computer Science, vol. 5784, 2009, pp. 270–277.
[34] N.M. Su, G. Mark, Designing for nomadic work, in: 7th ACM Conf. on Designing Interactive Systems, Cape Town, South Africa, 2008, pp. 305–314.
[35] A. Dearden, J. Finlay, Pattern languages in HCI: a critical review, Human Computer Interaction 21 (1) (2006).
[36] S. Lukosch, T. Schümmer, Groupware development support with technology patterns, International Journal of Man–Machine Studies 64 (7) (2006) 599–610.
[37] C. Gutwin, S. Greenberg, A descriptive framework of workspace awareness for real-time groupware, in: Awareness in CSCW, Computer Supported Cooperative Work: The Journal of Collaborative Computing, 11 (3–4) 411–446 (special issue).
[38] N. Preguica, J.L. Martins, H.J.L. Domingos, S. Duarte, Integrating synchronous and asynchronous interactions in groupware applications, in: Proc. CRIWG '05, Springer Verlag, 2005, pp. 89–104.
[39] I. Marsic, An architecture for heterogeneous groupware applications, in: Proc. ICSE '01, IEEE Press, 2001, pp. 475–484.
[40] D. Grigoras, M. Riordan, Cost-effective ad hoc networks management, Future Generation Computer Systems 23 (8) (2007) 990–996.
[41] M. Perry, K. O'hara, A. Sellen, B. Brown, R. Harper, Dealing with mobility: understanding access anytime, anywhere, ACM Transactions on Computer–Human Interaction 8 (4) (2001) 323–347.
[42] Y. Wang, E. van de Kar, G. Meijer, Designing mobile solutions for mobile workers: lessons learned from a case study, in: Proceedings of 7th International Conference on Electronic Commerce, ICEC'05, Xi'An, China, 2005, pp. 582–589.

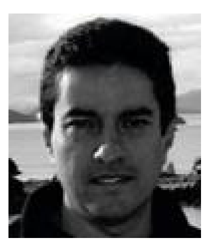


Juan Rodríguez-Covili received a Bachelor degree in Computer Science from the University of Chile, where he is currently pursuing a Master degree in the same field. His research interests include computer supported collaborative work, mobile networks and applications, social networking and video games. All his work is enhanced by his experience as IT consultant for many projects at the industrial production processes, and the financial and educational sector.

Sergio F. Ochoa is assistant professor of Computer Science at the University of Chile. He received his Ph.D. in Computer Science from the Catholic University of Chile. His research interests include computer supported collaborative work, educational technology and software engineering. Dr. Ochoa is a member of IEEE, ACM and the Chilean Computer Society and sits on the Steering Committee of the LACCIR (Latin American and Caribbean Collaborative ITC Research Initiative). He currently serves as an IT consultant for a number of public and private organizations in Chile.

José A. Pino is full professor of computer science at the Universidad de Chile. His research interests include computer supported collaborative work, human–computer interaction, and software industry studies. He has served as President of the Chilean Computer Science Society (SCCC) and President of CLEI (the Latin American Association of Universities Concerning Information Technology). He has co-authored six books and published research papers in international conferences and journals, including Journal of the ACM, Communications of the ACM, Decision Support Systems, Interacting with Computers, and Information Technology and People.

Valeria Herskovic is a Ph.D. candidate in Computer Science at the University of Chile. Previously, she obtained her Bachelor's and Engineering degree from the same university. Her research interests include mobile computing, collaboration, emergency support and human–computer interaction. She is currently in the last stages of her thesis work concerning evaluation of mobile collaborative software.



Jesus Favela is a professor of computer science at CICESE, Mexico, where he leads the Mobile and Ubiquitous Healthcare Laboratory. His research interests include ubiquitous computing, medical informatics and human–computer interaction. He holds a B.Sc. from the Universidad Nacional Autónoma de Mexico (UNAM) and M.Sc. and Ph.D. from the Massachusetts Institute of Technology (MIT). He is a member of the ACM and of the Sociedad Mexicana de Ciencia de la Computacion, of which he is former president (SMCC).

David A. Mejía, is a Computer Science Ph.D. student, at Centro de Investigación Científica y de Educación Superior de Ensenada (CICESE), Baja California, México. He is also a Professor of Computer Engineering at the Universidad Autónoma de Baja California (UABC), Mexico. His research interests include computer supported collaborative work, human–computer interaction, medical informatics, ubiquitous computing and workflow systems.

Alberto L. Morán is a professor of computer science at UABC, Mexico, where he leads the Technologies for Intelligent Environments Group. His research interests include CSCW, ubiquitous computing, and medical informatics. He holds a B.Sc. from UABC, a M.Sc. from CICESE, and a Ph.D. from the Institut Nationale Polytechnique de Grenoble (INPG). He is a member of the ACM and IEEE Computer Society.