Facilitating Dynamic Schedules for Healthcare Professionals

Cormac Driver, Éamonn Linehan, Mike Spence, Shiu Lun Tsang, Laura Chan and Siobhán Clarke.

Abstract—**Healthcare professionals working in highly dynamic hospital environments typically have correspondingly dynamic schedules that are difficult to manage. Emergent tasks and shifting priorities result in existing schedules becoming obsolete. Pervasive computing technology can aid healthcare professionals in organising daily activities by exploiting knowledge of planned and emerging tasks, of patients and colleagues, and of the general state of the working environment. The Hermes software framework at Trinity College Dublin supports the development of mobile applications for human activity management that exploits such environmental knowledge. Supported applications are based on the concept of a *trail*. This paper describes the Hermes framework and illustrates how it can be used to implement applications that can automatically reorganise the schedules of healthcare workers based on sensed changes in their environment.**

Index Terms—**context-aware scheduling, pervasive computing, healthcare, trails**

I. INTRODUCTION

HEALTHCARE professionals working in a hospital environment typically have many responsibilities contending for their time. With tasks ranging from providing medical care and monitoring patients to undertaking administrative responsibilities, it is often the case that healthcare professionals have a seemingly endless set of changing tasks to carry out. Consequently, they must manage their time by composing their activities into prioritised to-do lists. However, hospitals are inherently highly dynamic environments in which task interruptions and delays are commonplace. Additionally, previously unforeseen tasks can emerge that may require attention alongside the already scheduled tasks. In the face of such change, static paper or whiteboard-based to-do lists can become difficult to manage and, in the worst case, obsolete.

Recent advances in embedded sensor and mobile computing technology have given rise to a range of possibilities in pervasive healthcare. Among these is the opportunity to aid healthcare professionals by automatically managing their schedules in the face of significant contextual events that can negatively impact their schedule. In this paper we present Hermes, a software framework for developing mobile, context-aware

applications based on the concept of a *trail*, which is a contextually scheduled collection of activities. The framework contains reusable structure and behaviour common to mobile, context-aware activity scheduling applications, saving developers from repeatedly addressing common challenges. We illustrate how the framework can be used to develop an application to dynamically adapt the schedules of healthcare professionals based on their personal, environmental, and patient contexts.

The remainder of this paper is organised as follows: Section II motivates our work in the area of healthcare and Section III presents our model for facilitating change. Section IV provides details of the Hermes framework. Section V contains a discussion of pervasive healthcare-specific challenges. Section VI discusses related work and Section VII concludes the paper.

II. MOTIVATION

Hospital workers function in an environment characterised by mobility, ad-hoc collaboration, frequent interruptions and a high degree of communication [1]. As a result, time management is a major concern and is one of the biggest challenges faced by healthcare professionals today [2]. The dynamic nature of the working environment makes time management difficult, with wasted time having serious negative consequences. Additionally, patients typically want more time with their physicians than they are currently receiving [3] and therefore maximising consultation time through effective time management and task organisation is a worthy objective.

Established time management techniques include using a personal calendar and/or a prioritised static to-do list. To-do lists should comprise of a prioritised list of tasks to be accomplished within the period of a working day and should ideally be composed before the distractions of the day occur [4]. To-do lists used in hospitals are traditionally memory, paper, or whiteboard-based, and in recent years they have migrated to mobile computing devices such as Personal Digital Assistants (PDAs). While the use of to-do lists for time management is unarguably beneficial, the static nature of the to-do list reduces its effectiveness in a highly dynamic work environment such as a hospital. Static to-do lists do not adapt or reorganise in the face of the inevitable interruptions, delays and unforeseen emergent activities that are a feature of hospital life. For this reason, the prioritised ordering of a carefully considered, predefined to-do list can quickly become stale once work

commences.

In the following section we introduce the trails model for facilitating context-triggered change to user activity schedules. This model can be used to implement adaptive activity management applications for healthcare professionals operating in highly dynamic environments. In addition, such applications enable a number of the core care delivery functions for electronic health records identified by the United States Institute of Medicine [5]. Specifically, decision support and administrative process functions are facilitated.

III. THE HERMES FRAMEWORK & TRAILS - OVERVIEW

A software framework is a reusable implementation of all or part of a software system and is a set of classes (some abstract) with behaviours defining the way in which instances of those classes collaborate [6]. The Hermes framework provides generic components containing structure and behaviour common to mobile, context-aware trails-based applications [7]. Mobile, context-aware applications are those that run on devices such as PDAs and smart phones, and have an awareness of the physical and social situation in which they are deployed. Depending on the availability of appropriate sensor information, they can detect and reason about context such as patient vital signs and the location of colleagues. A trail is a set of activities or tasks, together with associated information (e.g., spatial and temporal information and user preferences) and a dynamically reconfigurable recommended visiting order. Trail order is affected by significant changes in personal and environmental context as they occur. Combining the trails concept with mobile, context-aware technology creates opportunities for innovative activity-based applications in the area of pervasive healthcare. An example is an application to optimally schedule ward rounds and related activities for physicians.

The following scenario illustrates how a mobile, context-aware trails-based application can be deployed to assist healthcare professionals in carrying out their daily activities in an optimal manner despite the frequent occurrence of unforeseen events. This scenario was conceived based on the authors' experiences as practicing physicians and computer scientists and considers only a subset of the task types and roles that can be supported by the Hermes framework.

A. Pervasive Healthcare Trails Scenario

Dr. Edward Phealan, a physician at Dublin City Hospital, arrives for work at the hospital and switches on his mobile device. The mobile device is running a trails-based healthcare application. The application retrieves a list of patients that Dr. Phealan has to visit (administrative tasks can also be retrieved). The patients on the list are selected specifically for Dr. Phealan and ordered in an optimal manner (illustrated in Figure 1) according to a stated policy e.g., most urgent or shortest path between patients. The application considers the following context when selecting and scheduling patient visits:

- Dr. Phealan's location and role. Initially, Dr. Phealan

is in his office (information likely to change), and his role is resident doctor (unlikely to change).

- Information stored in the hospital information system such as each patient's condition and medical history including all information recorded during their current stay in the hospital, the physician(s) they have seen previously (if any) and their location (bed/ward).

- The status of each patient, which is automatically sensed by monitoring equipment e.g., heart rate, temperature, blood pressure, whether they are awake.

- Dr. Phealan's personal preferences e.g., to be alerted when behind schedule, preference for visiting new patients first and minimal walking.
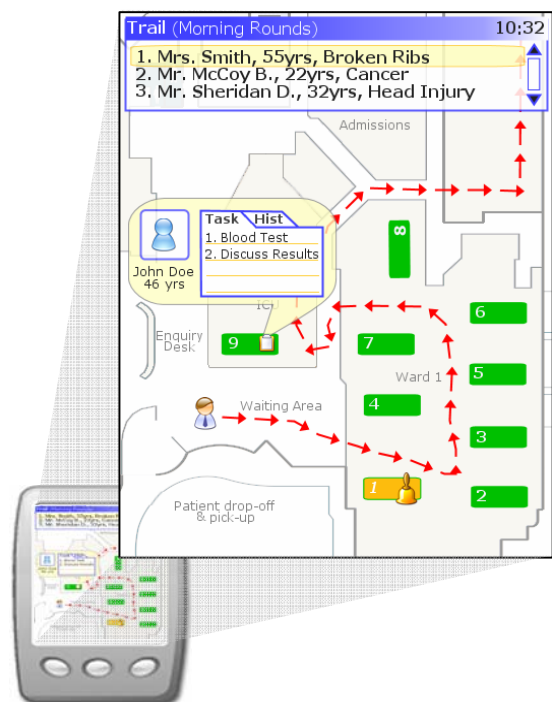

Figure. 1. Initial trail showing context-aware ordering and activity information.

Dr. Phealan consults his device and sets out to visit his first patient who has recently been involved in a road traffic accident (RTA). He uses his mobile device to review the patient's medical history while en route to the ward. Dr. Phealan carries out a patient examination and determines what actions need to be taken. He adds two tasks, an x-ray and a blood test, into the system. He specifies a medium level of priority for each task (as neither is life-critical), along with information about when the results are required, who needs to carry out the task, whether or not he needs to be notified about the results and whether a consultation with the person carrying out the task is required. The x-ray task is assigned to a radiologist's work trail (Dr. Stephenson) and the blood test is assigned to a phlebotomist's trail (Dr. Blake), both of whom are deemed by the application to have workloads that can cater for the addition of these tasks. Their trails are reconfigured to reflect the addition of the new tasks. Dr. Phealan's trail is up-

dated with the task of consulting with Dr. Stephenson later in the day to discuss the x-ray results. This interaction is illustrated in Figures 2 and 3.
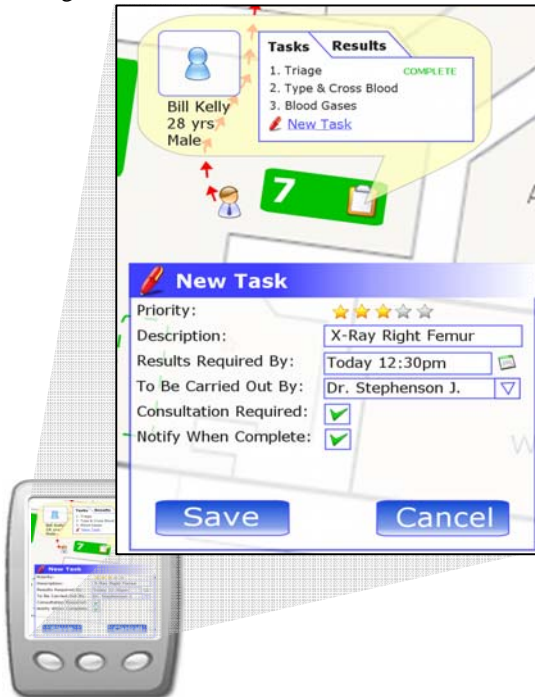

Figure. 2. Task created by Dr. Phealan and assigned to Dr. J. Stephenson.
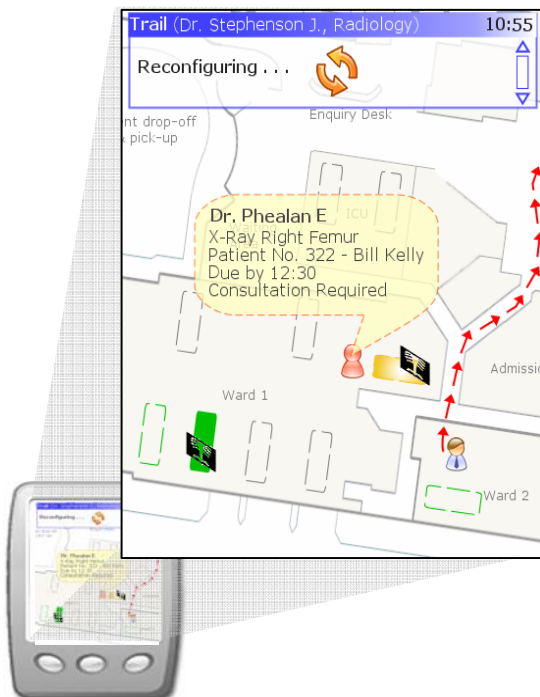

Figure. 3. A new task is received by Dr. J. Stephenson and her existing trail is reconfigured to cater for the new task.

Based on a predefined stated preference, Dr. Phealan's mobile device vibrates to alert him that he has been with the patient for the time he has allocated for that patient visit, and he is advised that he should proceed along the trail in order to stay on schedule. Dr. Phealan consults his mobile device and continues with his rounds.

While examining another patient later on in the day, Dr. Phealan is alerted that his trail has been reordered. The automatic monitoring equipment attached to his first patient, the RTA victim, has sent information to his mobile device indicating that the patient's vital signs have dropped to a critical level. Dr. Phealan's trail is reordered to take account of this new context information and his mobile device advises Dr. Phealan to return to the first patient immediately. This interaction is shown in Figure 4. Dr. Phealan returns to the patient and acts to return the patient to a stable state. He then resumes his rounds.
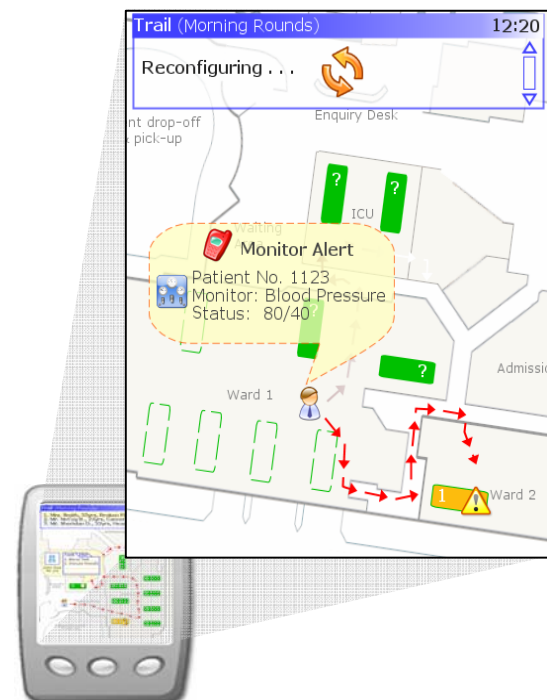

Figure. 4. An urgent alert triggered by a patient sensor causes Dr. Phelan's trail to be reconfigured.

While walking along the corridor on the way to the next patient on his trail, Dr. Phealan serendipitously meets the radiologist he is scheduled to meet later in the day. His mobile device recognises that he can take advantage of this situation and alerts him to this fact. Dr. Phealan discusses the x-rays with Dr. Stephenson and this task is removed from both physicians' trails.

IV. HERMES ARCHITECTURE

This section describes the Hermes software architecture. First, we discuss the technical challenges related to mobile, context-aware applications such as that described in the scenario. Second, we describe assumptions we are making about a hospital's technical environment. Finally, we describe the components of the Hermes architecture and some experiences with its use in trails applications.

A. Challenges

To implement the application described in the scenario there are number of core challenges that must be addressed. These challenges are not specific to the pervasive healthcare domain (though some may be exacerbated in this environment) and therefore must be addressed each time a trails-based application is implemented. The Hermes framework provides reusable and extensible software components that address these issues so that developers do not have to repeatedly tackle them. A summary of the prominent challenges follows below.

1) Collaborative Context

The more contextual information an application obtains, the clearer the picture of the surrounding environment, and the easier it is for an application to enact appropriate behavior for the current situation. Significant sources of contextual information are surrounding devices, which might contain up-to-date information such as the RTA patient's blood pressure or the location of the radiologist for a serendipitous meeting. We term contextual information acquired from these remote sources *collaborative context.*

The risk of exposing personal context to others through collaborative context necessitates measures be taken to limit the propagation of such privacy sensitive information. As wireless networks are a broadcast medium, it also becomes necessary to provide data security to ensure only the intended recipient can read the data. It is a challenge to ensure the maximum exploitation of the available context through collaboration while ensuring that no privacy sensitive information is unnecessarily or inadvertently disclosed. Challenges related to multi-sensor/device context fusion such as coping with variability in sensor availability, defining context integration policies and resolving conflicts between contradictory data of the same type must be addressed. Additionally, interpretation complications are one of the major impediments to using a piece of collaborative context. Context interpretation describes the meaning an application assigns to a piece of context, which in turn affects how it is used [8]. For example, an application might assume that a given position represents its current location if that position is considered in isolation from other contexts such as time. The position might actually be a stale sensor reading e.g., Dr. Phealan's location from ten minutes ago. The Hermes framework is addressing these issues with *context capsules* [9], which provide a means to augment context with enough information for its reuse.

2) Context Management

The management of context information within context-aware applications has implications for semantic interoperability, query languages, context data storage, and context interpretation. Expressive, extensible and interoperable context management systems are necessary to support the implementation of sophisticated context-aware applications [10 - 12].

Addressing such challenges on mobile devices adds even more complexity [13], [14]. There has been considerable progress on context discovery, acquisition, semantic modelling and reasoning [15 - 20], but little attention has been given to the fact that these operations will have to be distributed across many mobile devices which may have to function while disconnected from any computing infrastructure.

There is also the challenge of persisting context information on the mobile device. The persistence of context information on the mobile devices facilitates disconnected operation and increases application response times as round-trips to centralised repositories are not required. In addition to the data modelling and physical storage challenges there is the issue of determining what context should be retained and for what period of time. The challenge is to ensure that enough related context is stored to uniquely characterise a situation, while working within the limits imposed by resource-constrained devices.

3) Trail Generation

Trail generation is a combinatorial optimisation problem similar to the classic Travelling Salesman Problem (TSP). The most obvious solution to such problems is to generate all permutations of the activities on a trail (or cities on a tour etc.) and rate each one against a predefined notion of optimality. However, the number of permutations is factorial in the number of activities, and this brute-force solution becomes impractical for even a relatively small number of activities [21], [22]. Therefore, it is necessary to find a satisfactory solution by trading solution quality against application responsiveness. Using approximation algorithms (e.g., heuristic and randomised approaches) it is possible to achieve solutions with a high probability of being within 2-3% of optimal in a practical amount of time. However, determining what a practical time is on a per-application basis and achieving this level of efficiency on a mobile device are non-trivial issues.

The trail generation problem compounds the TSP by necessitating a more complex evaluation function. An evaluation function quantifies the optimality of a solution, essentially encoding a human notion of optimality within the trail generation algorithm. The TSP typically uses the distance between cities to assess the worth of a particular permutation of cities. However, when evaluating a trail composed of patient visits in a hospital there is a much wider range of factors to be considered e.g., patient condition, medical history, current location, scheduled appointments, physician's preferences and role etc. The implementation of the evaluation function therefore becomes a multi-attribute utility estimation problem [23], with the additional processing required by the evaluation function affecting the overall efficiency of the trail generation algorithm. Additionally, individual users will most likely have their own view of what constitutes the optimal trail and therefore the evaluation function must be sensitive to user preferences.

4) Route Generation

To facilitate trail generation it is necessary to produce context-aware pedestrian-level routes between activities on a trail. These routes constitute a level of spatial awareness that allows the trail generation component to order tasks with respect to the travel times between them. If these routes are to also be used to aid the physician find their way then the challenges of

personalisation, context-awareness and maintaining relevance must be addressed [24]. It is a challenge to reason over a multi-resolution spatial world model in order to provide efficient route planning and monitoring on mobile devices [25].

Key problems to be overcome include the ability to integrate different resolution geometric data sources into a searchable topological map, taking into account appropriate contextual
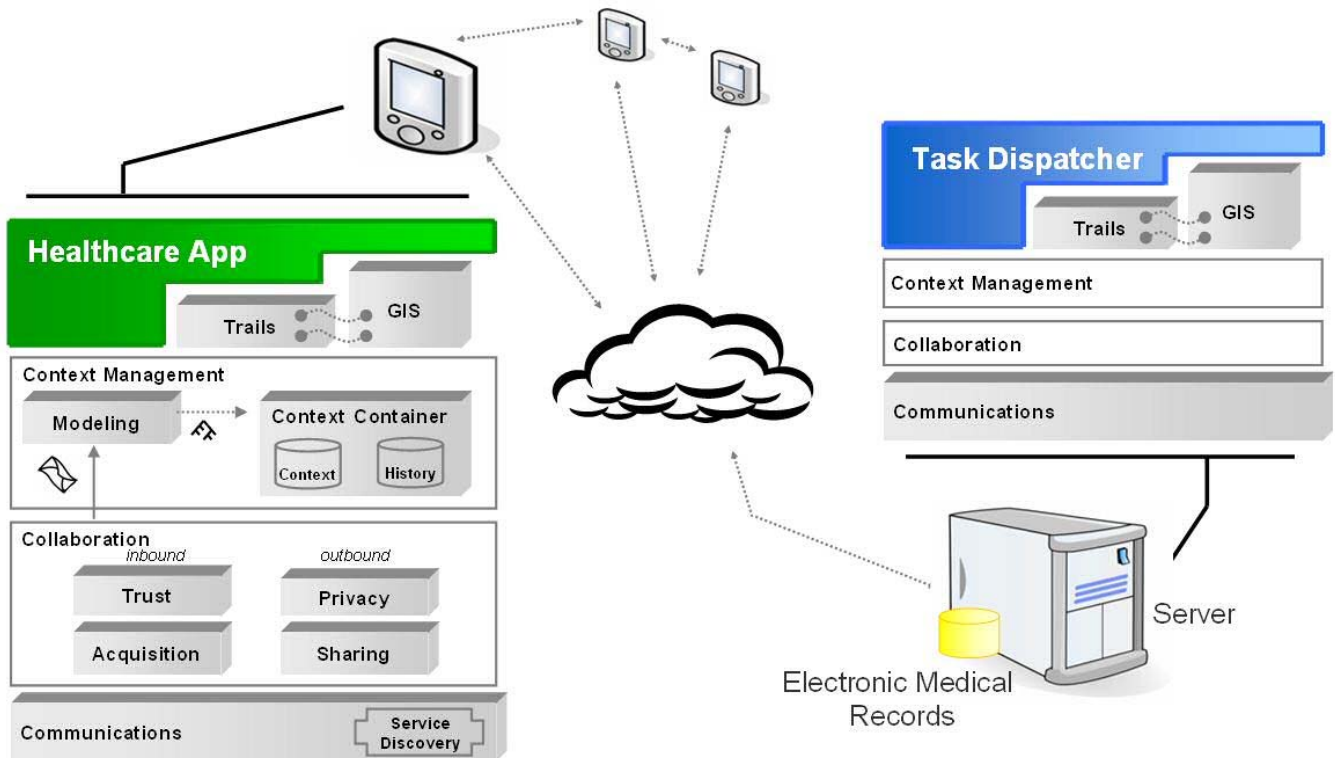

Figure. 5. The Hermes architecture showing the physical components and the layered software framework deployed on each system component

information during the planning process and monitoring the situation of the user and a subset of the world model for changes that may affect the appropriateness of a previously planned route. Additionally, incremental navigation should be supported [26]. This involves choosing an appropriate route presentation in order to minimise intrusive instructions to the user and maximise the user's ability to orientate themselves and to follow a route without undue cognitive load [27].

B. Assumptions

In order to support the implementation of the healthcare professional's schedule management application, we make a number of technical assumptions about the environment in which the application is deployed:

- *Mobile Devices* – Physicians have use of a mobile device.
- *Network Connectivity* – The mobile devices are capable of communicating with peer devices, sensors and centralised services across various networks. There is no assumption of continuous network coverage throughout the hospital as the frequencies commonly used for wireless networking may conflict with some medical devices resulting in loss of signal [28].

- *Context-Awareness* – Mobile devices are responsible for maintaining their own situational awareness.
- *Electronic Medical Records* – There is an existing, accessible electronic medical record system in the hospital.
- *Human Resource Management* – A computer-based system is used to manage staff rosters and roles/responsibilities. This information can be mined by the Hermes framework to determine the role each user plays in the hospital and to ensure tasks are assigned to appropriately qualified individuals.

C. Architecture

Figure 5 illustrates how the components of the Hermes framework enable the scenario presented above. The software elements are designed in such a way as to enable autonomous operation of the mobile devices. This adds robustness to the system by enabling the mobile devices to continue to operate in the presence of server outages or network unavailability. This is achieved by replicating many of the server-based services on the mobile devices and giving the mobile devices the ability to communicate directly with each other.

1) Mobile Device Software Architecture

At the bottom layer of the mobile device software archi-

tecture is the *Communications* component. This layer is responsible for managing communication with sensors, infrastructure and peer devices over a variety of networks. The Communications component is itself context-aware it is aware of the state of its network interfaces. When no suitable network interface is available, message delivery can be delayed or upper layers of the architecture can be immediately notified of the failure. The action taken is dependant on the message priority.

The *Collaboration* component divides its responsibilities between inbound and outbound considerations. On the inbound side (left) are the *Acquisition* and *Trust* components. The Acquisition component is responsible for acquiring specific types of context on behalf of an application e.g., patient vital signs from sensors, tasks received from the task dispatching service. The Trust component works with Acquisition to augment incoming messages with trust-related metadata. On the outbound side (right) are the *Sharing* and *Privacy* components. The Sharing component is responsible for responding to incoming requests for context and the Privacy component provides for the protection of personal data by limiting which types of context are exchanged with which peers. The Collaboration component, along with the Communication component, facilitates the opportune meeting of Dr. Phealan and Dr. Stephenson in the scenario. This meeting illustrates the establishment of an ad-hoc peer-to-peer connection between the devices and collaboration through exchange of context, enabling the devices to reconfigure their trails.

The *Context Management* component contains the *Context Container* and *Modelling* components. The Modelling component caters for fusion of context and the conversion of incoming context to the context manager's internal representation. The Context Container is responsible for storing current context. Once context values in the Context Container are overwritten they move to the context history. The Context Container makes context available by providing an interface supporting queries and supporting components registering to be notified of changes in particular contexts.

Hermes distinguishes itself from other context management software frameworks through the addition of two closely collaborating components that provide solutions for a number of technical challenges common to trails-based applications. These components are the *Trails* component and *GIS* component. The Trails component is responsible for the generation and management of trails. In the scenario the Trails component generates an initial trail when it first receives Dr. Phealan's list of tasks from the task dispatcher. The Trails component later reconfigures the trail as appropriate in response to acquired context. The GIS (Geographic Information System) component is responsible for maintaining a model of the structure and geography of the user's environment. Such a model supports intelligent spatial reasoning by the Trails component, improving the overall user experience of trails-based applications.

A healthcare application operates above this layer which is responsible for generating graphical user interfaces such as those proposed in the scenario. This application, as well as providing trails functionality to healthcare professionals can use the context services of the Hermes framework to implement other services such as proactive connection to the hospital's electronic medical record server and retrieval of relevant patient medical records in anticipation of the bedside visit.

The Hermes architecture is designed to support multiple simultaneously executing applications. All such applications use the components of the Hermes architecture to dynamically adapt their behaviour to the user's current situation.

2) Server Software Architecture

The responsibilities of the server illustrated in Figure 5 are the following:

- *Task Aggregation* – The server is responsible for collecting new tasks entered by users at existing stationary terminals or on mobile devices. The server is also capable of supporting decision making, ensuring compliance with best clinical practice, and by generating tasks in response to alarms from patient monitoring equipment.
- *Dispatch* – Having acquired the static context regarding user roles in the hospital the server will dispatch jobs to appropriate users based on its knowledge of their current trails.
- *Efficiency* – The server supports mobile devices by taking the burden of some of the more complex computational elements when possible.

The server executes instances of the Communications, Collaboration, Context Management, Trails and GIS components with the same responsibilities as instances of these components executing on the mobile device. Although their responsibilities have not changed, their implementation is optimised to take advantage of the increased computational abilities of the server and the more reliable network interfaces the server has at its disposal. For example, the Communication component can access the hospital's wired network infrastructure allowing more data to be transferred, such as high-resolution digital x-ray imagery, than would be possible on the wireless network. It is also capable of receiving large volumes of sensor data from patient monitoring equipment. Similarly, the Context Management component is capable of maintaining context history much further into the past than would be possible on a mobile device, facilitating the reporting requirements previously discussed. It is also capable of modelling the incoming patient monitor readings as patient context in the context model. The server determines when a sensor reading is critical and immediately notifies the appropriate physician. Since the server is in communication with a large number of mobile devices it can maintain a more accurate context model by aggregating context shared from each individual device. The GIS component on the server maintains a more

detailed spatial model of the entire hospital whereas the mobile devices only require knowledge of a user's immediate environment in order to aid navigation and determine proximity. The Trails component facilitates the off-loading of trail generation, a computationally intensive operation, in order to decrease application response time or enhance trail optimality. If a mobile device is disconnected from the server its trail component can execute a scaled-down trail generation algorithm.

A *Task Dispatcher* application, implemented at the top level of the server's software framework, performs the task aggregation and dispatching. The Task Dispatcher uses static information regarding hospital layout, staff roles and hospital policy in combination with dynamic context such as an individual staff member's current trail, location and device network connectivity to allocate new tasks to appropriate staff members. The purpose of this component is to ensure that workloads are balanced and that the maximum number of patients is treated in the most logical order, maximising the safety and wellness of all patients.

D. Applications

The trails-based approach to dynamic schedule management applications has been implemented and evaluated with encouraging results. For example, *Oisín goes to Trinity* is a context-aware application for people within the Trinity College Dublin (TCD) campus. The application provides campus-wide trails to different kinds of users. One class of user is the tourist. Founded in 1592, TCD has a number of world-famous attractions. Activities are optional, with time and tourist interests important scheduling factors. Another class of user is a student on his first day. Many activities are mandatory (e.g., register, attend course director's talk) and many are optional (e.g., join debating society). The trails are based on and affected by environmental and personal context and are dynamically reconfigurable. We completed a user study of the application involving a user trial with questionnaires and interviews both before and after the trial. 21 subjects took part in the study. 83% of subjects agreed with the trail reconfiguration decisions either all of the time or most of the time. Only 12% of subjects disagreed that the reordering decisions were better than those they could have made themselves. 83% of subjects thought that the amount of control the application had when reconfiguring their trail was acceptable. This result is considered positive as users were given no edit control over the trail and could only effect a change by purposely deviating from the trail. The study asked users "Can you imagine yourself using mobile, context-aware technology as part of everyday life?" both before and after the study. Answers to this question were slightly more positive after application usage.

Since this study, extensions to the architecture include additional support for user preferences, collaborative context, refined context-aware routes and trail generation algorithms, more advanced spatial reasoning and rendering, and improved form factor. To evaluate these extensions, we use a multiplayer riddle solving game, called *Riddle Hunt,* in which the application assists users by both presenting context information such as other players' locations and generating trails designed to help the player win the game based on stated preferences or strategies.

V. DISCUSSION

While trails-based applications can make an important contribution to facilitating dynamic schedules for healthcare professionals, employing the trails model in a hospital environment presents a number of issues.

A. Domain Knowledge

Automating many of the decisions in the scenario requires that domain knowledge be available and interpretable by the application. For example, the scenario assumes the application knows to match the role of the health professional to the appropriate activity. This would prevent a nurse or a brain surgeon from being assigned an activity like performing open-heart surgery. Additionally, it assumes that an alarm received from a blood pressure monitor means that a critical activity should be added to a physician's trail. Unfortunately, some of this assumed knowledge would require encoding the equivalent of much of the practical education and training of healthcare professionals and the policies of the particular hospital.

Existing medical knowledge bases are not comprehensive enough to encode the entire workings of a hospital. However, clinical decision support systems [29] exist that make use of domain knowledge for a particular medical field to facilitate physicians in decision making. Such systems could, if sufficiently comprehensive medical knowledge bases existed, determine the relative priority of activities for the purpose of trail generation. This priority, like other domain knowledge, can also be assigned by a medical professional upon inspection of each activity. Alternatively, it can be an interactive hybrid of both. This sort of interactive system is common in case-based reasoning [30] and other machine learning approaches [31].

B. Resource Coordination

The scenario considers a single physician's schedule while doing his rounds at a hospital. The schedule is made up of activities assigned by the Task Dispatcher application based on hospital policies and the activity load of the hospital staff. In reality, there must be schedule collaboration with several other entities within the hospital such as other medical personnel, patients, and hospital resources [32]. If these other entities are not considered, it could lead to a conflict in schedules, e.g., two physicians scheduled to see the same patient at the same time. A possible solution is to create trails for patients as well as limited resources such as operating rooms or expensive medical equipment. The fact that all of these schedules are dynamic and highly interrelated means that a minor change in one can cause a dramatic change in others. The MedPAge system [32] attempts

to address this problem through a multi-agent schedule negotiation system focused on the patient.

C. Network Connectivity

In the scenario there is no requirement for complete network coverage. However, issues arise when there is insufficient connectivity in some areas of a hospital.

If a trail is reordered on the mobile device and is not able to communicate that change to the server, the resource coordination necessary to schedule or free the appropriate resources e.g., patients and equipment, cannot take place. In addition, tasks may not be immediately delivered to out of range devices. This problem is currently addressed in hospitals by secondary means of communication such as wireless beepers or audible paging over an intercom. These different modes of assigning an activity could be integrated so that the system remains in a consistent state.

VI. RELATED WORK

A. Mobile Computing-Based Schedule Management

The viability and tangible benefits of deploying a mobile computing-based schedule management system in a hospital are described in [33]. The Blackberry mobile computing device was deployed throughout the Baylor Healthcare System, a network of hospitals employing 15,000 people in Dallas, Texas. 600 Blackberry devices were deployed to hospital staff, including physicians, nurses and administrators, who used them to update their calendars, retrieve email and access remote systems. The case study revealed that staff productivity was increased by four hours per week with resultant savings projected to reach $3 million annually. The schedule management application deployed was not context-aware and did not feature any trail generation or automatic dynamic reconfiguration capabilities.

In recent years there has been a significant amount of research into the development of mobile, context-aware tour guides [34 - 40]. In terms of schedule generation and management, P-Tour [41] and the Dynamic Tourist Guide (DTG) [42] are two of the most sophisticated.

P-Tour is a personal navigation system that allows tourists to compose a multi-destination schedule, taking individual preferences and time restrictions into account. The end-user application resides on a mobile device and communicates via WiFi with a remote server. The server executes a genetic algorithm to compute tours to within 2% of optimal in 15.5 seconds (for 14 destinations). The optimal tour is transmitted to the tourist's mobile device. The system can recognise when the user has deviated from the recommended path and either alerts the user or recomputes the schedule. In this case it can recompute the schedule more efficiently as it can use the previous solution as a basis.

The Dynamic Tourist Guide is a mobile agent that selects tourist attractions from a predefined database (based on elicited user preferences) and plans a tour of these attractions. The end-user application is deployed on a mobile device and the tours are generated on a remote server. The DTG can compute a tour of 16 activities to within 5% of optimal in a response time of 5.5 seconds. The server-based agent uses a directed depth-first algorithm that incorporates a number of heuristics including using an average duration estimate for all attractions and trading off attraction relevance against the cost (in time) of traveling to and exploring the attraction. These heuristics, while increasing the efficiency of the algorithm, can result in tours that maximise the number of attractions, hence achieving a higher overall value (based on their tour assessment method). However, such tours can have a lower average score per attraction meaning that tourists may not be that interested in many of the attractions and their favourites may be absent.

Both the P-Tour and DTG tour generation and recomputation algorithms are server-based and therefore assume an uninterrupted wireless network connection. Neither application provides support for disconnected operation in terms of tour management, something that we believe is necessary given the nature of wireless networks. It is unlikely either P-Tour or DTG could address this issue by deploying their tour generation and recomputation algorithms on the user's mobile device. The algorithms are designed to work on a powerful server platform and their performance would be significantly degraded by migrating to a resource-constrained platform such as a PDA or smart phone.

P-Tour and the DTG both acquire location context from an external source (GPS device). Temporal and attraction context are acquired internally to the application, from the mobile device or server. Given that these approaches have been implemented as end-user applications, it is unclear how much support is available for extending the range of internal and external context considered by the applications. The consideration of additional contexts from multiple diverse sources would be necessary if these approaches were to be deployed in a context-rich hospital environment.

Finally, it is unclear whether the DTG's heuristic approach would be feasible when deployed in a hospital environment. It may not be possible to make generalisations about physicians' tasks in the way that the DTG developers make generalisations about attractions e.g., using a single activity duration regardless of whether the attraction is a church spire or an art museum. The use of heuristics in a life-critical environment requires significant consideration.

B. Middleware for Pervasive Healthcare

Healthcare Aide [43] is a mobile device-based virtual assistant for physicians based on a framework for Middleware Adaptability for Resource Discovery, Knowledge Usability and Self-healing (MARKS) [44]. The Healthcare Aide application facilitates physicians in making patient notes, sending and retrieving information, securing data access and chatting with other physicians. The MARKS middleware contains components to support data privacy, continuous communication, application and energy efficiency, mobility, concurrent applications, and service discovery. A

user study involving a small number of physicians and medical professionals was conducted using a cognitive walkthrough strategy. Overall the Healthcare Aide application was positively received. The application does not feature support for schedule management, static or dynamic. The context processing and resource discovery components of the framework could potentially be used to provide context data to a trails component. However, implementing this extension to the framework would not be trivial given the challenges that must be addressed to provide trails support.

Bardram et al. have proposed a middleware for pervasive healthcare [45]. The middleware is intended to support the development of a range of healthcare applications e.g., a point-of-care consultation application using the patient's overhead television as the display and a medicine conference involving heterogeneous devices. The middleware is oriented towards adapting different configurations of computing devices and consequently does not support the development of dynamically reconfigurable trails-based applications. Bardram has also developed a Java Context-Awareness Framework (JCAF) which has been demonstrated to work in a pervasive healthcare environment [46]. JCAF can be used to satisfy the context requirements of a mobile, context-aware trails-based application but does not provide any trails-specific framework components.

C. Context-Aware Middleware

Muñoz et al. [47] present a mobile, context-aware system designed for use in a hospital environment. This work addresses the issues of managing communication in a hospital setting, typically involving different actor locations, work-hours and communication paths. Although this system does not make any attempt to manage workflow, it could be used as an informal tool for task management where physicians can make use of context-awareness in their instant messaging conversations. For example, physicians or nurses finishing a shift could send instructions describing patient treatment to any nurse on the next shift. In this system the mobile devices can perceive and reason about their own context and use that information to retrieve messages from a server. Similar to our approach, all users are required to be equipped with a mobile device. Although informal communication is possible there is no notion of scheduling of tasks and consequently it is possible that the server may deliver large numbers of instructions to a user in a very short space of time, because it is not aware of their schedule and has no comprehension of the tasks and task durations included in the informal instructions. This work also does not address the integration of other sources of context information within the hospital and does not consider contextual information from medical devices.

There are many examples of application or domain specific context-aware systems such as the hospital example above. In addition there are more flexible and extensible context-awareness systems that aim to support context-awareness in any class of application in any domain. Two

examples of such systems are described below.

The Context Toolkit [8] is a framework for developing context-aware applications. The Context Toolkit provides applications with access to context information while hiding the details of context sensing and management.

Gaia [48] is a distributed context-based middleware infrastructure. Gaia is described as, "a meta-operating system that aims at supporting the development and execution of portable applications for active spaces". Gaia achieves this by providing context-based services to users in a limited physical space, allowing applications access to their current context as maintained by the environment.

These systems have varying degrees of support for ad-hoc communication, context acquisition from external sensors, semantic context modeling, fusion of context from disparate sources and the data management of spatiotemporal contextual information. These systems provide a service to applications allowing them to easily retrieve situational information and often include some support for reasoning and presentation of the context.

VII. CONCLUSIONS

We have presented the Hermes framework for mobile, context-aware trails-based applications and illustrated how it supports context-aware scheduling in a hospital environment. The framework provides generic components that can be reused or extended to implement a pervasive healthcare application to address the existing problem of hospital task scheduling in the face of shifting priorities and emergent tasks. However, a number of issues remain to be addressed including domain knowledge management, resource coordination and the complications caused by intermittent network connectivity. Notwithstanding these issues we believe that Hermes is a useful technology that can be applied in the healthcare domain to facilitate improved management of healthcare professionals' schedules.

REFERENCES

[1] J.E. Bardram. "Activity-Based Computing: Support for Mobility and Collaboration in Ubiquitous Computing". Personal and Ubiquitous Computing, 9(5):312–322, July 2005.

[2] S. Cowap. "Time Trials". The Medical Journal of Australia, 183 (2): 72. The Australasian Medical Publishing Company, 2005.

[3] J. Ogden, K. Bavalia, M. Bull, S. Frankum, C. Goldie, M. Gosslau, A. Jones, S. Kumar and K. Vasant. ""I want more time with my doctor": a quantitative study of time and consultation". Family Practice 21(5):479-483. Oxford University Press, 2004.

[4] S.J. Parrillo and F.G. Wenger. "Time Management for the Academic Emergency Physician". eMedicine Clinical Knowledge Base. www.emedicine.com/EMERG/topic673.htm. 19 May 2006.

[5] P. Tang. "Key Capabilities of an Electronic Health Record System". Letter Report. Institute of Medicine Committee on Data Standards for Patient Safety. Board on Health Care Services. Washington D.C.: National Academies Press. July 2003.

[6] D. Roberts and R. Johnson. "Evolving Frameworks: A Pattern Language for Developing Object-Oriented Frameworks". ACM Pattern Languages of Program Design 3, pp. 471-486, 1996.

[7] S. Clarke and C. Driver. "Context-Aware Trails". IEEE Computer, Vol. 37, No. 8, 2004.

[8] D. Salber, A.K. Dey and G. D. Abowd. "The Context Toolkit: Aiding the Development of Context-Enabled Applications". In Conference

on Human Factors in Computing Systems (CHI '99), Pittsburgh, pp. 434-441, May, 1999.

[9] M. Spence and S. Clarke. "Preserving Context with Context Capsules". In Third International Workshop on Modeling and Retrieval of Context, Boston, July 2006.

[10] G. D. Abowd, M. R. Ebling, H.-W. Gellersen, H. Lei, G. Hunt (eds.), Special Section on Context-aware Pervasive Computing, IEEE Wireless Communications, Vol. 9, No. 5, Oct. 2002.

[11] S. Khungar and J. Riekki. "A context based storage system for mobile computing applications". ACM Journal of Mobile Computing and Communications Review 9(1):64-68, 2005.

[12] G. Banavar, A. Bernstein."Challenges in Design and Software Infrastructure for Ubiquitous Computing Applications". Communications of the ACM , Volume 45 , Issue 12, Pages: 92 - 96, 2002.

[13] K. Henricksen, J. Indulska, and A. Rakotonirainy, "Modeling context information in pervasive computing systems". In 1st International Conference on Pervasive Computing (Pervasive), volume 2414 of Lecture Notes in Computer Science, pages 167–180. Springer, 2002.

[14] K. Cheverst, N. Davies, K. Mitchell, A. Friday. "Experiences of Developing and Deploying a Context-Aware Tourist Guide: The Lancaster Guide Project". Proc. 6th Ann. Int'l Conf. Mobile Computing and Networking (Mobicom 00), ACM Press, 2000.

[15] D. Garlan, D. P. Siewiorek, A. Smailagic, and P. Steenkiste. "Project Aura: Toward Distraction-Free Pervasive Computing". IEEE Pervasive Computing, April, 2002.

[16] W. G. Griswold, R. Boyer, S. W. Brown, and T.M. Truong. "A component architecture for an extensible, highly integrated context-aware computing infrastructure (Active-Campus)". In International Conference on Software Engineering, pp. 363–372, Portland, Oregon, 2003.

[17] A. Dey, D. Salber, and G. D. Abowd. "A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications". Human-Computer Interaction (HCI) Journal, 16(2-4):97–166, 2001.

[18] G. Stevenson, P. Nixon, R. Ian Ferguson. "A General Purpose Programming Framework for Ubiquitous Computing Environments". Ubisys: System Support for Ubiquitous Computing Workshop, Ubi-Comp, Seattle, Washington, 2003.

[19] G. Biegel, V. Cahill. "A Framework for Developing Mobile, Context-aware Applications". Proceedings of the Second IEEE International Conference on Pervasive Computing and Communications (PerCom '04) pp. 361, 2004.

[20] W. Schilit. "A System Architecture for Context-Aware Mobile Computing". PhD Thesis, Columbia University, May, 1995.

[21] C. Driver, E. Linehan and S. Clarke. "A Framework for Mobile, Context-aware Trails-based Applications: Experiences with an Application-led Approach". Workshop 1 ("What Makes for Good Application-led Research in Ubiquitous Computing?"), Pervasive 2005.

[22] K. Cheverst, N. Davis, K. Mitchell and P. Smith. "Providing Tailored (Context-Aware) Information to City Visitors Source". In Proceedings of the International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems, pp. 73 – 85. Lecture Notes in Computer Science, Springer-Verlag, 2000.

[23] D. von Winterfeld and W. Edwards. "Decision Analysis and Behavioral Research". Cambridge University Press, 1986.

[24] A-M. Nivala, and L. T. Sarjakoski. "Need for Context-Aware Topographic Maps in Mobile Devices". In 9th Scandinavian Research Conference on Geographical Information Science, Espoo, Finland, pp. 15-29, June, 2003.

[25] D. Nicklas, M. Großmann, T. Schwarz, S. Volz, and B. Mitschang,. "A Model-based, Open Architecture for Mobile, Spatially Aware Applications". In Proceedings of the 7th International Symposium on Advances in Spatial and Temporal Databases, pp. 117–135. Springer-Verlag, 2001.

[26] C. Kray, K. Laakso, C. Elting and V. Coors. "Presenting route instructions on mobile devices". In Proceedings of the International Conference on Intelligent User Interfaces 2003, pp. 117-124, 2003.

[27] J. Baus, A. Kruger, and W. Wahlster. "A resource-adaptive mobile navigation system". In, International Conference on Intelligent User Interfaces, pp. 15–22, San Francisco, California, 2002.

[28] V. Gehlot and E.B. Sloane. "Ensuring Patient Safety in Wireless Medical Device Networks". IEEE Computer, vol. 39, no. 4, pp. 54-60, April, 2006.

[29] Coiera, E., "Guide to Health Informatics". 2nd ed: Hodder Arnold, 2003.

[30] D.W. Aha and L.A. Breslow. "Refining conversational case libraries". In Second International Conference on Case–Based Reasoning, pp. 267–278, Springer, 1997.

[31] C. Sammut and R.B. Banerji. "Learning concepts by asking questions". Machine Learning: An Artificial Intelligence Approach, Volume II, pp. 167-191, Morgan Kaufmann, 1986.

[32] T.O. Paulussen, N.R. Jennings, K.S. Decker, and A. Heinzl. "Distributed Patient Scheduling in Hospitals". In International Joint Conference on Artificial Intelligence, Acapulco, Mexico, 2003.

[33] "Baylor Healthcare Case Study". Available from Blackberry at www.blackberry.com/products/pdfs/BlackBerry_Baylor_CS.pdf. Research in Motion Limited 2004.

[34] K. Cheverst, N. Davies, K. Mitchell and A. Friday. "Experiences of Developing and Deploying a Context-Aware Tourist Guide: The GUIDE Project". In Proceedings of the Sixth Annual International Conference on Mobile Computing and Networking. ACM, 2000.

[35] G. Abowd, C. Atkeson, J. Hong, S. Long, R. Kooper and M. Pinkerton. "Cyberguide: A Mobile Context-Aware Tour Guide". ACM Wireless Networks - Special issue: Mobile Computing and Networking: selected papers from MobiCom '96, p. 421-433, 1997.

[36] J. Wohltorf, R. Cisseé, A. Rieger and H. Scheunemann. "Berlin-Tainment: An Agent-Based Serviceware Framework for Context-Aware Services". Workshop on Context Awareness. 2nd International Conference on Mobile Systems, Applications, and Services - MobiSys, Boston, 2004.

[37] B. Schmidt-Belz, A. Zipf, S. Poslad and H. Laamen. "Location-based mobile tourist services - first user experiences". Int. Congress on Tourism and Communications Technologies, Helsinki, Finland. Springer Computer Science, 2003.

[38] H. Anegg, H. Kunczier, E. Michlmayr, G. Pospischil and M. Umlauft. "LoL@: Designing a Location Based UMTS Application". ÖVE-Verbandszeitschrift e&i, Springer, Heidelberg, Germany, 2002.

[39] IST Project Fact Sheet. Mobile Tourism Guide. (m-ToGuide) http://icadc.cordis.lu/fep-cgi/srchidadb?ACTION=D&CALLER=-PROJ_IST&QM_EP_RCN_A=61826. June, 2006.

[40] Crumpet Project Overview, http://www.fit.fraunhofer.de/projekte/ crumpet/index_en.xml, June 2006.

[41] A. Maruyama, N. Shibata, Y. Murata, K. Yasumoto and M. Ito. "P-Tour: A Personal Navigation System for Tourism". In 11th World Congress on Intelligent Transport Systems, vol. 2, pp. 18-21, 2004.

[42] K. ten Hagen, M. Modsching and R Krammer. "A Location Aware Mobile Tourist Guide Selecting and Interpreting Sights and Services by Context Matching". In 2nd International Conference on Mobile and Ubiquitous Systems, pp.293-304, IEEE Computer Society 2005.

[43] M. Sharmin, S. Ahmed, S.I. Ahamed, M. Haque, and A.J Khan. "Healthcare Aide: Towards a Virtual Assistant for Physicians, Patients, Nurses and Resident Physicians Using Pervasive Middleware". In 1st Workshop on Ubiquitous and Pervasive Health Care (UbiCare 2006) in conjunction with Fourth Annual IEEE International Conference on Pervasive Computer and Communications (PerCom 2006), Pisa, Italy, pp. 490-495, March 2006.

[44] M. Sharmin, S. Ahmed, and S.I. Ahamed. "MARKS (Middleware Adaptability for Resource Discovery, Knowledge Usability and Self-healing) for Mobile Devices of Pervasive Computing Environments". In Third International Conference on Information Technology: New Generations (ITNG 2006), Las Vegas, Nevada, USA April, 2006.

[45] J.E. Bardram and H.B. Christensen. "Middleware for Pervasive Healthcare - A White Paper". Advanced Topic Workshop - Middleware for Mobile Computing, Heidelberg, Germany. In association with ACM Middle 2001.

[46] J.E. Bardram. "The Java Context Awareness Framework (JCAF) - A Service Infrastructure and Programming Framework for Context-Aware Applications". In 3rd International Conference on Pervasive Computing, L.N.C.S., Springer-Verlag, Munich, 2005.

[47] M.A. Muñoz, M. Rodríguez, J. Favela, A.I. Martinez-Garcia, and V.M. González. "Context-Aware Mobile Communication in Hospitals". IEEE Computer, vol. 36, no. 9, pp. 38-46, September, 2003.

[48] M. Roman, C.K. Hess, R. Cerqueira, A. Ranganathan, R.H. Campbell, and K. Nahrstedt. "A Middleware Infrastructure to Enable Active Spaces". IEEE Pervasive Computing, pp. 74–83, October, 2002.