Chiu, D. K. W., Choi, S. P. M., Wang, M., & Kafeza, E. (2008). Towards Ubiquitous Communication Support for Distance Education with Alert Management. *Educational Technology & Society*, *11* (2), 92-106.

Towards Ubiquitous Communication Support for Distance Education with Alert Management

Dickson K.W. Chiu
Dickson Computer Systems, 7 Victory Avenue, Kowloon, Hong Kong // dicksonchiu@ieee.org

Samuel P.M. Choi
School of Business and Administration, The Open University of Hong Kong // schoi@ouhk.edu.hk

Minhong Wang
Faculty of Education, University of Hong Kong // magwang@hkucc.hku.hk

Eleanna Kafeza
Department of Marketing & Communications, Athens University of Economics & Business, Greece // kafeza@aueb.gr

ABSTRACT

In distance education, communications among students, educators, and administrators have been one of the most important problems. The distance education programs in The Open University of Hong Kong (OUHK) span not only Hong Kong but also many cities over a large area in China. To improve and monitor the quality of communications among students, tutors, and staff, we propose a communications management infrastructure based on an alert mechanism. Requests and messages in these systems (referred to as alerts) have to be delivered and handled timely to provide ubiquitous distance education communication support and management. We also propose using this platform for the integration of partners in China with the OUHK offices. Presently, most systems cannot address urgency and alerts are often handled in an ad-hoc manner. In this paper, we propose a sophisticated alert management system (AMS) for ubiquitous communications management in distance education under various requirements. We develop a model for managing alerts, in which alerts are associated with communication tasks and a set of parameters are captured for message routing and urgency management. The AMS matches the specialties of the educators and staff who receive an alert, based on the alert specification. We then propose a routing mechanism that is initiated when the alert message is not acknowledged or handled within the deadline, so that the alert can be re-routed if necessary. Monitoring is especially essential to ensure timeliness and availability of consultation or decision, otherwise suitable exceptions should be raised and handled. We outline our implementation framework for communications among education administrators, education staff, students, and system administrators, enhanced by alter management via various communication devices for ubiquitous support of communication in distance education.

Keywords

Distance education, Ubiquitous communication, Alert management, Conceptual modeling, System design architecture

Introduction

Distance education refers to a mode of learning in which students and educators are situated in different place and/or time (Bates, 2005). In the past, distance education is delivered by printed materials and audio tapes via postal service. Students usually have to study independently and in isolation. With the advances in telecommunications technologies, distance education can now take place through multiple media, such as radio broadcasts, video-conferencing, and online learning platforms. Interactions among students and teachers now become possible: the temporal and spatial barriers are therefore alleviated. In recent years, the advent of mobile technologies have been offering cheaper and more convenient communications so that students and educators can access information and communicate with one another anytime anywhere ubiquitously by using various mobile devices (Chang et al., 2007). Mobile technologies bring both new opportunities and challenges to distance education.

Retaining awareness, accessibility, and responsiveness among students and educators is one of the main concerns for modern distance education (Tuckman 2005). With the increasing mobility of students and educators as well as the increasing number of busy professionals undertaking life-long learning, merely online learning platforms and web-based administrative services are no longer adequate (El-Bishouty et al., 2007). Neither traditional practices of using cellular phones and pagers for communications, nor isolated electronic means like email or instant messenger are adequate for seamless integration with existing and future learning platforms. Multi-channel ubiquitous communications support now becomes necessary (Fano & Gershman, 2002; Chang et al., 2007). At the same time, tutors and staff are easily overwhelmed by the increasingly large amount of messages and may overlook some important or urgent ones. We refer to these important or urgent messages as *alerts* (Kafeza et al., 2004; Chiu & Choi, 2005).

As such, a number of issues must be considered in order to support effective ubiquitous communications for distance education. For instance, alert management should include various alert types and parameters that qualify the service provider to receive an alert. Apart from service suitability, application specific considerations like costs, waiting time, service time may also be important. In addition, routing, monitoring, and logging the alerts are also mandatory functionalities for ubiquitous communications and their automated management. Based on our experience in alert management for healthcare applications (Chiu et al., 2004) in which the management of urgent and important communications is also the key to success, we propose to adopt an alert management system (AMS) as a key driver software module for communications management in distance education.

The contributions of this paper include: (i) an enhanced conceptual model for managing alerts based on the requirements of communications management in distance education and a set of routing parameters; (ii) a practical architecture for the AMS based on contemporary Web Services for programmatic interactions, together with multiple-platform ubiquitous support for human users; (iii) a mechanism for (re-)routing alerts and increasing their urgency level when alerts are not acknowledged or processed within a deadline; (iv) a demonstration of the applicability of this approach with a case study of the Open University of Hong Kong.

The rest of our paper is organized as follows. First, we discuss an overview of our methodology and the requirements for ubiquitous communications management in distance education. Then, we compare related work. Next, we describe our alert conceptual model that captures both data and process integration requirements, followed by our system architecture, highlighting the AMS and its mechanisms for monitoring and ubiquitously routing the alerts. Before we conclude with our future work direction, we discuss the applicability of our alert mechanism with respect to the requirements of the major system stakeholders.

Background and Methodology Overview

The Open University of Hong Kong (OUHK) is the first education institution in Hong Kong that offers government-recognized undergraduate and post-graduate degree programs via open and distance education. Figure 1 summarizes the main stakeholders in our case study.

Unlike other conventional universities, OUHK students have the flexibility to plan their learning schedule towards their degrees according to their own pace. Students study through the well-developed self-instructional materials with other academic supports such as tutors and online learning platforms. Most of the students are mature and have their full-time job while studying part-time. They are often at senior management positions and have their family to take care of. Many of them also need to travel frequently. Therefore, as introduced earlier, ubiquitous communication support is particularly useful to them.

For every course, tutors are assigned to provide students with assistance in the study by answering the students' questions via telephone tutoring or online platforms. The tutors also mark the assignments with constructive comments and conduct tutorial classes once a month at one of the many teaching locations. As most tutors are also part-time, they have similar communication requirements as the students.

To meet the growing demand for higher education after China's accession to the World Trade Organization, OUHK offers a number of postgraduate programs in Mainland China. OUHK currently collaborates with 13 Mainland partner institutions (such as the Institute of Continuing Education in the Graduate School of the Chinese Academy of Social Sciences and the State Statistical Bureau) to provide continuing education in over 22 cities. Up to now, more than 6,000 students in Mainland have enrolled and over 3,000 students have graduated. In August 2003, OUHK opened a new office in Shenzhen with a team of 15 staff to undertake the administration and coordination work of the Mainland operations. In addition, the Shenzhen office provides support for multimedia education, corporate training, consultancy services, and the production of learning materials.

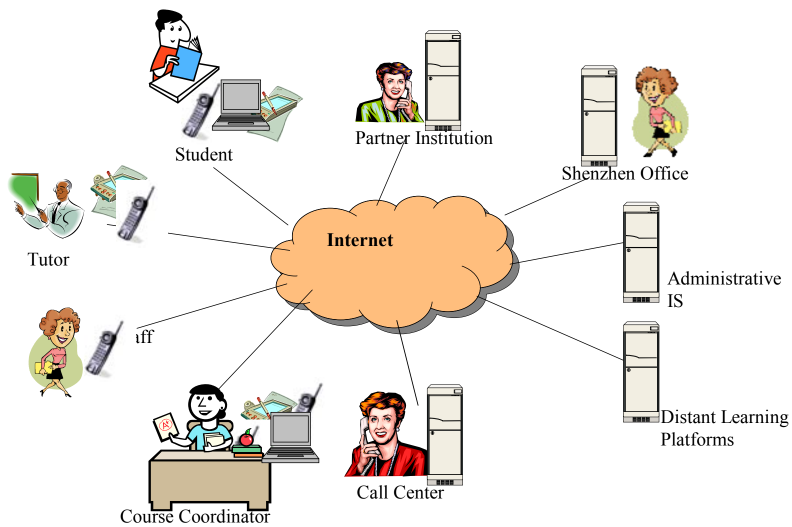

Figure 1. Stakeholders of distance education at the OUHK case study

With the rapid growth of the number of students and programs in OUHK, the importance of IT support in course delivery has become increasingly apparent. In the past few years, OUHK has invested a large amount of resources to develop IT systems to support learning, such as an electronic library, an online learning platform, and other web-supported administrative services. Students benefit from such IT systems all the way from enrolment to graduation.

However, enhancing communication support still remains a priority issue to be further improved. Efficient, seamless, and ubiquitous communications among the tutors, students, and the university staff are essential for maintaining high quality of teaching in distance education. While the online learning platform is clearly useful, we observe that some education activities cannot be effectively conducted in pure online mode without other communication channels, such as direct voice interaction or conferencing. For instance, students have difficulties in preparing their assignments for the courses that require many mathematical formulas or drawings. Students who have difficulties in reading and typing Chinese may find online discussion troublesome. In particular, communication requests towards assignment deadlines, tests, and examinations become urgent and increase greatly in number. So, tutors and staff are easily overwhelmed by the large amount of messages and may overlook some urgent or important ones. Further complication may arrive when ubiquitous support of multi-device, anytime, anywhere communications are desired. Another related problem is that some students and tutors might login to the online learning platform very irregularly.

During the past years, the education programs at Hong Kong institutions have been established using an efficient mechanism to carry out business processes via e-mails, online platforms, and phones. Nevertheless, such practices cannot be directly applied to the Mainland programs due to the geographical scattering of the involved parties. With the increasing mobility of students and tutors as well as the increasing number of busy professionals undertaking life-long learning, a pure Internet-based solution is inadequate. Any party may be out of Internet access from time to time; the efficient and effective management of multiple-channel communications is therefore necessary.

Motivated by the general lack of alert management systems, we started off our study by gathering the objective and requirements from various stakeholders (as summarized in Figure 1). This led to our proposal towards an alert management system, which is *robust, efficient, cost effective, simple,* and *user friendly* to improve and manage ubiquitous communication. Based on these objectives, detailed requirements were elicited and formulated into an alert conceptual model. Then we sketched an overall system architecture for the call center, with focus on the AMS design. We then worked out the detailed mechanisms for each component of the system. In the design, we paid more attention to flexibility so that alert management policies could be adapted to various situations for various partners. According to these designs, we then developed an AMS prototype (Chiu et al., 2004) for evaluation.

As for deployment, we plan to split it into phases. The first phase is to establish an automatic AMS to manage all the alerts and integrate with existing online learning platforms. After getting used to the new arrangements and fine tuning of the alert management policies, the second phase is to extend the system to connect to partner institutions. In the third phase, we plan to include further intelligence into the system, in particular, with advanced capability reasoning (Chiu et al., 1999), scheduling with mobile location dependent information, service negotiation, and the integration with traffic routing.

Related Work

Recent advances in telecommunications technologies have enabled a global platform for organizations and individuals to communicate among one another, conduct various activities, and provide value-added services (Lin & Chlamtac, 2000). The use of Personal Digital Assistants (PDA) and mobile phones for ubiquitous computing are getting popular and can now support Internet accessibility. In addition, new mobile "smart devices" featured with different software and hardware capabilities, such as Wireless Application Protocol (WAP) and Short Message Service (SMS) have been introduced into the market.

Weiser (1991) first proposed the idea of ubiquitous computing: specialized elements of hardware and software, connected by wires, radio waves, and infrared, would become so ubiquitous that no one would notice their presence. Different from virtual reality, ubiquitous computing techniques bring computing devices into everyday life. This ubiquitous environment provides an opportunity, which permits human-computer interactions away from any workstations. That is, the user may interact with multiple devices such as cell phones, Personal Digital Assistants (PDAs), touch-sensitive screens, and notebook computers within a session which may be operating in different environments.

There exist many e-learning platforms in the market, such as Blackboard (http://www.blackboard.com), Lotus Learning Management System (http://www.ibm.com/software/tw/lotus/product/t22.html), and WebCT (http://www.webct.com). Most of them are designed to support learning at the course-activity level, such as course materials publishing, discussion forums, and assignment submission. Recently, Cesarini et al. (2004) have proposed a workflow approach to expand the scope to the learning process level. The proposed system, in particular, allows instructors to define structured courses and to specify the study paths to guide students throughout their learning. However, these systems have not yet address the current need for mobile and ubiquitous communication support.

The ubiquitous learning environment seamlessly connects learning collaborators, learning contents, and learning services (Cheng et. al., 2005; Yang, 2005). This is particularly important for distance education because students and educators are situated in different places at different time. Sakamura & Koshizuka (2005) review some enabling technologies for ubiquitous learning. Joiner et al. (2006) explore the design of situated educational and compelling experiences using ubiquitous technology. While most studies explore extensions to classroom environments for full-time students and educators, none of the existing work on ubiquitous learning explicitly addresses the need of distance education.

In ubiquitous learning, communications, interactions, and collaborations are key success factors. In particular, collaboration heavily relies on the interaction and communications among the collaborators (Zhang et al., 2005, Kajita & Mase, 2006). For example, Zhang et al. (2005) propose a framework of social interaction support for ubiquitous learning, but without detailed implementation of the communication aspects. Yang (2005) describes how context aware ubiquitous learning environments help peer-to-peer collaborative learning. However, these studies do not adequately address the problem of communication management problem for distance education.

On the other hand, related researches on communication management have been rather concerned in healthcare applications, which pioneer the study of managing urgent and important messages as alerts. Ammenwerth et al. (2000) report one of the major issues, which is mobile technologies, can help in hospitals is communication and reachability management. Communications that include the patient, the message sender, and the urgency are useful. Hripcsak et al. (1996) preliminarily identify the need for event monitors, and describe some of the requirements of such monitors including tracking healthcare events, looking for clinically important situations and sending messages to the providers. Eienstadt et al. (1998) further categorize messages into *alerts, results,* and *replies*. The limitation of their approaches is that they only focus on alerts that can be handled by two-way pagers. Ride et al. (1994) argue that the problem of figuring out to whom the message should be sent is a difficult one. They only suggested some ad hoc solutions, e.g., sending a message to whoever has recently examined the patient electronic record.

Suomi and Tähkäpää (2003) study the requirements of a contact center for public healthcare with a case study in Turku, Finland, and point out that contact routing is the main system functionality. They also provide a good survey of call centers that run with older technologies, which we believe, is also applicable to distance education services. In our related study on alert management in distance education, we proceed further to detailed system design and prototyping, with a focus on urgency requirements for alert routing, by employing additional mobile technologies and integrating processes of partner institutions.

Although information integration issues are not new in database research communities (Sheth & Larson, 1990), applying workflow technologies in different application domains has many unique properties that entail special integration design considerations such as Sheng & Chen (1990) and Cesarini et al. (2004). In the context of workflow management systems (WFMS), we have proposed to separate user alerts from user sessions with the WFMS for flexibly integrating them in our ME-ADOME system (Chiu et al., 2002). More recently, we have proposed an agent-based cognitive approach to dynamic workflow management, especially by enhancing the interaction with the environment and incorporating environmental information into real-time decisions on workflows or processes (Wang and Wang, 2006a).

As an extension to existing process models such as Sheng & Chen (1990) and Cesarini et al. (2004), our process model abstracts information regarding roles and their schedules of service providers taking these roles. A preliminary version of our process model in the context of WFMS is available in Chiu et al. (1999). We have employed a bottom-up data-driven methodology to extend information systems into Web Services (Chiu et al., 2003). We further incorporate alerts and their routing in this paper. To our knowledge, there are no other WFMS employing this approach. Further, there has no other work on alert-based communication management for distance education currently.

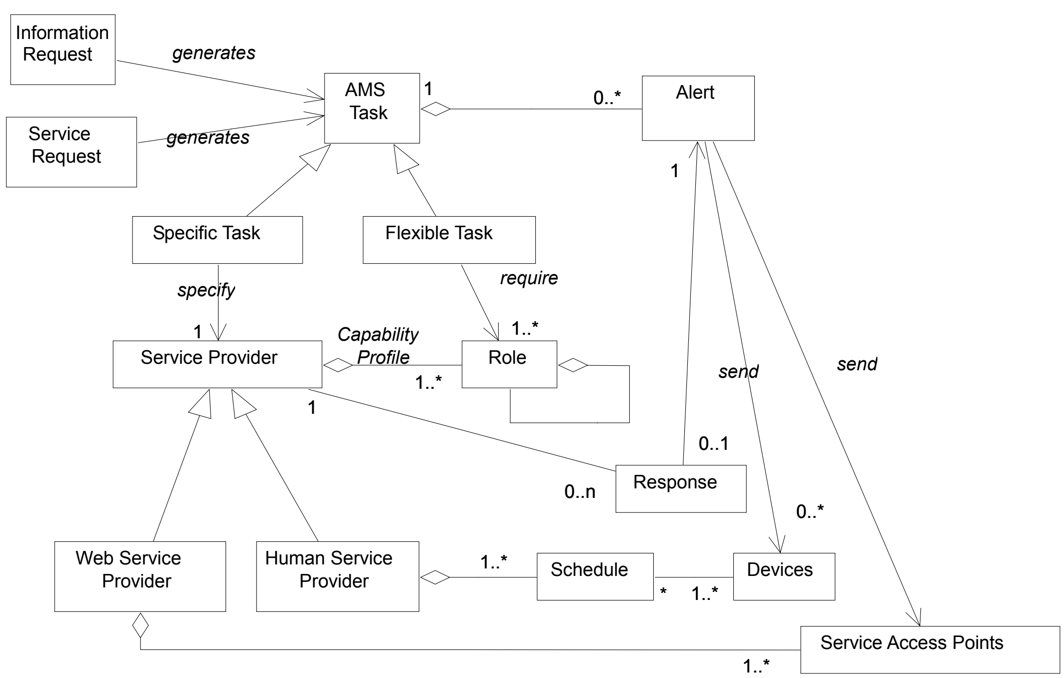

Figure 2. UML Class Diagram for alerts in AMS

Alert Conceptual Model

Based on the requirements discussed in the previous section, we design a unified alert mechanism for both data and process requests. *Figure 2* depicts our *alert conceptual model* in Unified Modeling Language (UML) Class Diagram. When a user (e.g., student) requires a service request (e.g., discussion with a tutor) or information request (e.g., accessing data), the AMS generates an *AMS task* to monitor the enactment of the request. The AMS task is called a *specific task* if it requires a specific service provider. Otherwise if only some criteria for a task are specified, it is called a *flexible task* as the AMS may have the flexibility to use matchmaking techniques (as discussed in the next section) to search for suitable alternative *service providers*. These criteria are specified in terms of roles, which can be used to represent the capability, position, and/or specialties of the service provider. A match is valid if the set of roles required is a subset of the service provider (Chiu et al., 1999).

The AMS task generates *alerts* that are routed to matching service provider(s), which can be *human service providers* (e.g., tutors) or *Web Services providers* (e.g., data service interfaces of partner institutions). Sometimes, an alert might need to be sent to several service providers (e.g., several tutors) with the similar capabilities, if the request is very urgent (e.g., near assignment deadlines). Besides identification information, an alert has an urgency level, which could be a function of time. Normally when an alert is not responded in a specified time, the AMS increases alert urgency with time, as further explained in the next section. Although traditionally alerts are small messages, we generalize the concept of alerts to include any additional information urgent, important, or that can better justify the request (new mobile devices can now support multimedia messages). For example, an audio file or an image could answer a student's question better than text. Further, to support ubiquitous communications, different devices of each human provider that one can access have to be tracked with respect to his/her schedule.

A *response* is a service providers' reply to an alert; the service provider may finish, confirm, or reject the request. For short enquiries to human service providers or simple processing of data by web service providers, the service providers may finish the request and respond with the required results right away. If more time is required, they reply with a confirmation response to reflect their commitment to the request. Should a service provider feel unable to serve the request or do so on time (e.g., a tutor is sick), he/she/it could reject the request.

<table>
<tr><td colspan="2">Distance Education Platforms and Workflows</td></tr>
<tr><td>Event/Exception Handling</td><td>Data/Process Requirements</td></tr>
<tr><td colspan="2">Alerts Managed by AMS</td></tr>
<tr><td colspan="2">Web Services and Mobile Devices</td></tr>
</table>

Figure 3. Alert Management in distance education applications

On the other hand, the AMS needs to deal with those alerts that are not responded by the deadline. From reported practice, we observe that as long as an alert is not responded, the service provider may be changed. As a result, the AMS can revise the matching between the alert and the service provider (Chiu et al., 1999). In addition, as the time passes without any acknowledgement, the urgency of the alert should increase as well. Thus, this may in turn change the service provider to be requested. In our model, we propose a flexible approach, in which a strategy can be defined by the administrator (see Section 5.3).

Figure 3 further summarizes the conceptual architecture of alerts (Chiu et al., 2004). The essence of alerts is to capture urgency requirements, as required by some distance education platforms or workflow. It should also be noted that exceptions are subclasses of events (Chiu et al., 1999; 2001). Exceptions often, but need not always, have urgency implications. Different from general events, alerts have more specific attributes, in particular, urgency level and service requirements. Different from exceptions, alerts need not be related to abnormal behaviors. That means, alerts can be (i) triggered *asynchronously* to handle an event or exception, or (ii) generated synchronously to satisfy the service requirements. Alerts received by a service provider can be handled by its international systems, or processed by human service providers, and/or forwarded to other external service providers.

System Design and Implementation

In this section, we first present our overall system architecture and then detail the mechanism of the AMS, which supports sophisticated alert monitoring and routing.

System Architecture

Figure 4 depicts the overall implementation architecture of our system. As the AMS only manages the alert, domain-specific application logic (in this case, the distance learning platform and workflow) is required for a complete system. When communications are required, the distance learning platform generates alerts with the necessary specification to the AMS. This approach separates the complex logic of alert management from the normal workflow, which follows the models we proposed (Kafeza et al., 2004; Wang and Wang, 2002). Any subsequent processing that depends on the response or the result of the external service has to wait till it finishes (as signaled by the AMS); otherwise the workflow can continue.

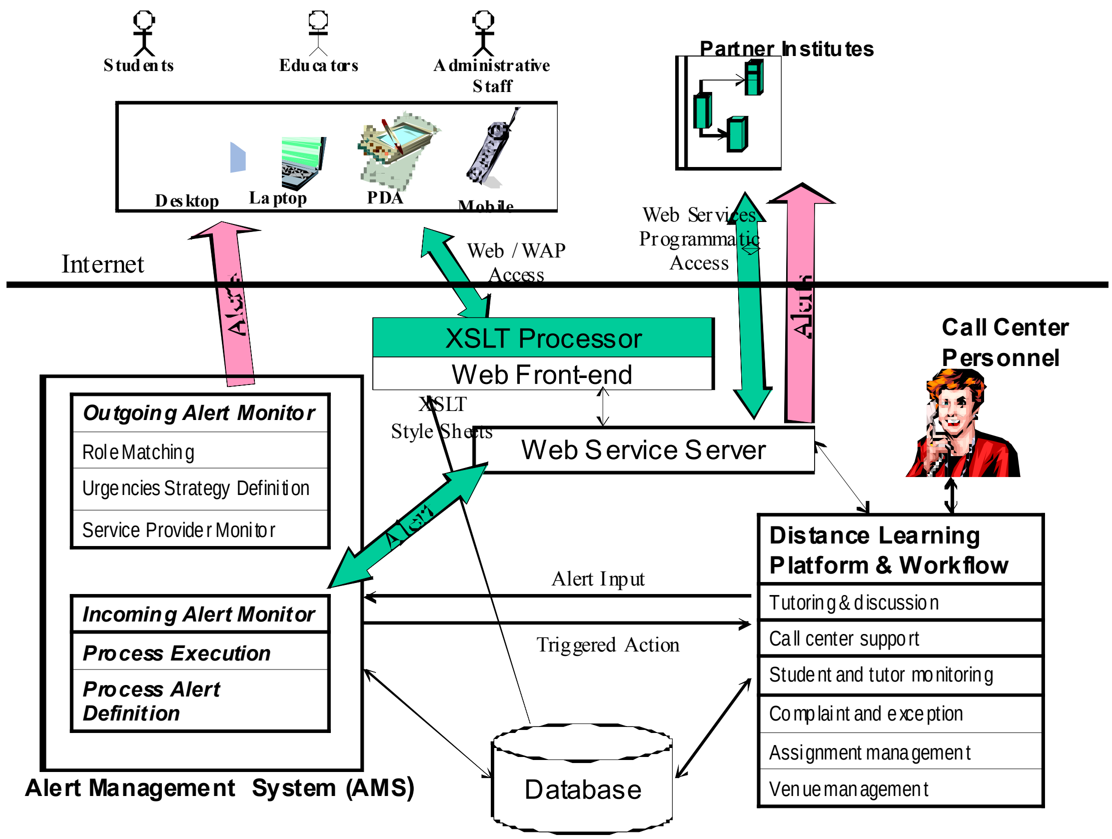

Figure 4. Overall system architecture highlighting the AMS

On the other hand, functions of the distance learning platform can be triggered by the Process Execution Module of the AMS to carry out timely appropriate actions in response to incoming alerts. In addition, the application logic supports an administrative Web front-end for the administrators or Call Center personnel. We plan to upgrade or add the application logic in the following systems to support alert management: tutoring and discussion system, call center support system, student and tutor monitoring system, complaint and exception management system, assignment management system, and venue management system. We detail how such mechanisms function in subsequent subsections.

To extend the accessibility for users on different platforms, eXtended Markup Language Stylesheet Language (XSL) technology is employed (Lin and Chlamtac, 2000). For example, different Hypertext Markup Language (HTML) outputs are generated for Web browsers on desktop PCs and PDAs respectively, while WAP Markup Language (WML) outputs are generated for mobile phones. We can then build an *alert response form* for a tutor through WAP on a mobile phone and a PDA browser respectively.

AMS Mechanisms

As the AMS supports both the sending and responding of alerts for communications, a person or organization can be a service provider and requester at the same time. The AMS consists of two major parts (see *Figure 4*). Let us explain how the AMS functions with a running example, in which a student requests consultation service from a tutor.

The *Incoming Alert Monitor* is responsible for receiving and queuing alerts and enacting the corresponding services (processes). In addition, the *Process and Alert Definition* module supports a tool, with which administrators may pre-define the tasks and their associated alerts according to the AMS model. So, a student can enter a request through the Call Center's Web portal or a mobile device (e.g., PDA or SMS message), by selecting the subject, subject module, interaction type (e.g., email, chat-room, phone discussion), problem category, urgency level, etc. Alternatively, the student may phone the Call Center and the operator enters the request. The request becomes an incoming alert with the parameters supplied by student, and is received via web interface, SMS message, or web service. Then, the alert triggers the appropriate handlers in the distance learning platform for execution by the *Process Execution* module. In this scenario, the handler is in the Tutoring and Discussion system. After validating the incoming request, the handler will in turn trigger an outgoing alert, which attempts to find an appropriate tutor to response to the student's enquiry in time.

The *Outgoing Alert Monitor* subsystem is responsible for sending the alerts to the corresponding service providers (persons, programs or organizations), and monitoring their responses. Human service providers (e.g., tutors) are communicated with ICQ (Weverka, 2000), SMS, and email too. In this way, a service provider supporting only manual interactions may still participate in data and process integration through an *alert response form*, through which the required response can be entered and send to the requestor. For example, a tutor may enter a simple response through such a form to respond to the student immediately.

The *Outgoing Alert Monitor* subsystem consists of three modules: the *Urgencies Strategy Definition*, the *Role Matching*, and the *Service Provider Monitoring* modules. The *Urgencies Strategy Definition* module enables the administrators to specify the policies that will be followed if the alert is not acknowledged within the deadline (e.g., send the alert to another tutor of the subject). The *Role Matching* module is responsible for identifying the service providers to which the alert will be forwarded (e.g., select a suitable tutor "intelligently" as explained in the next subsection). The *Service Provider Monitoring* module is responsible for applying the strategies defined at the urgencies strategy definition by executing the actions specified by the administrators. Its functions include sending alert messages, receiving response, maintaining alert status, and logging information. For every response message received, the *Service Provider Monitoring* module updates the status information of the associated alert, and tags the alert as "taken care of". If the alert message has been sent to several service providers (e.g., tutors) for a very urgent request, the first one who confirms is assigned to the task while the others will receive a cancellation message instead. Then for every alert in the *active alert table* with its deadline expired, the module checks the *urgency strategy table*, executes the associated action, and updates the status information accordingly.

Figure 5 depicts a typical life cycle of an alert in UML Activity Diagram. All alert processing and messaging for an alert is logged ("Log alert" node) for auditing purposes. If the alert is a *specific* one (e.g., when a student requests enquiry with a specific tutor), there is no room for matchmaking. Otherwise, if the alert is a *flexible* one, a matching algorithm ("Find matching service provider" node) is invoked to search for a suitable service provider (e.g., when a student request enquiry with any tutor of a particular subject). The "Determine device / web service access point" node determines the device for a human or the Web Service access point for a Web Services provider respectively. This feature accommodates future system advancement opportunity such as supporting agent-assisted communication and artificial intelligent based tutoring. Then, the "Send alert" node sends the alert accordingly. If the "Check if response received by deadline" node fails, the AMS will increase the alert urgency level, thereby triggering the alert message to be resent to either the same service provider or a different suitable one (as discussed in the next subsection). The last tolerance level is guided by "Check if service performed upon service due" node. If the service is not performed within a deadline (e.g., a tutor does not response to a student), then the AMS generates a new alert to the relevant administrator to notify this exception. As such, additional manual or system-assisted exception handling processes (Chiu et al., 2001) such as compliant or disciplinary process can be carried out.

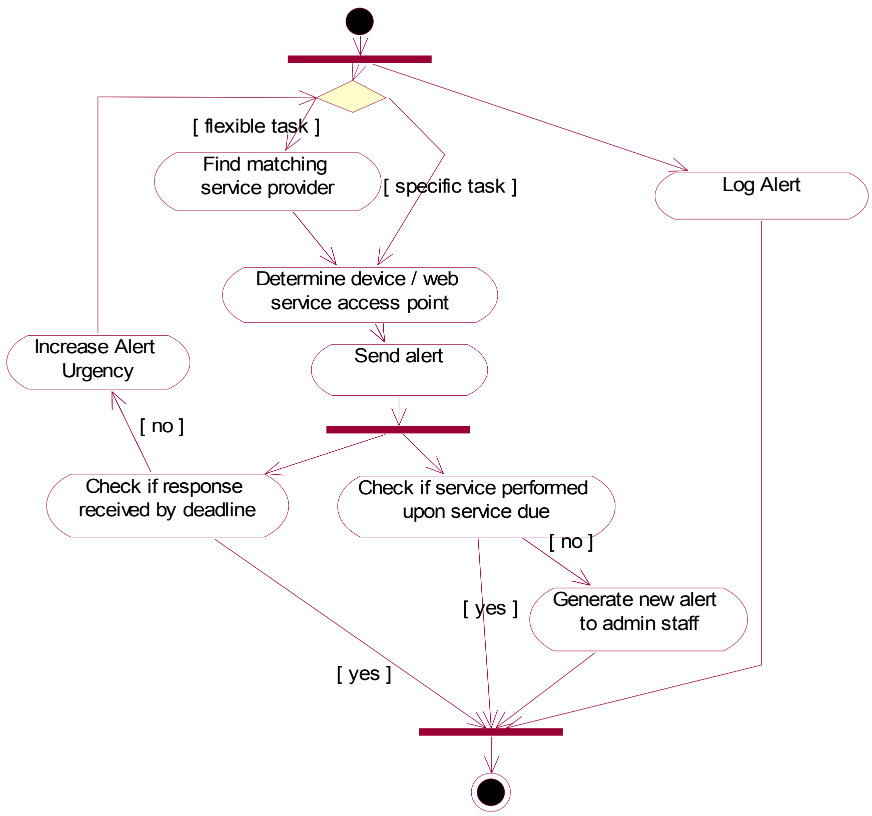

Figure 5. Typical life cycle of an alert in UML activity diagram

Service Provider Matching and Urgency Strategies

The service provider matching module is responsible for searching a service provider for each alert. The service provider matching algorithm (Kafeza et al., 2004) searches for those service providers that can play the role required by the alert. In this scenario, typically, any tutor of the involved subject can match a non-specific request for tutor consultation. The algorithm then selects those having a response time less than the deadline (e.g., by considering to pending requests of each tutor and their schedules in the system). This further restricts the set of service providers that can receive the alert. If the matching is successful, one service provider is selected according to an administrator-supplied cost function. In this application, the cost function can be based on the time required for service, hourly-rate of the service provider, etc. In case no matching is available (i.e., there exists no service provider with the requested role that can meet the deadline), the algorithm upgrades the alert by expanding the roles whenever possible (e.g., request a senior tutor or lecturer instead of a tutor). After the matching, an *active alerts table* keeps all instantiated alerts and the information whether the alert has been acknowledged or not.

If an alert is re-sent, the service provider for matching algorithm will take into account of the urgency strategy definition. The urgency strategy definition module is a tool for defining the policies according to which the urgencies of the alert will evolve. Moreover, this module is responsible for keeping and updating status information for the alerts. In our alert model, every alert is associated with an urgency value and a deadline, while every service provider is associated with an average response time for every service that it provides. During the specification phase, the administrator has to specify the *urgency strategy tables*. An *urgency strategy table* defines the policies for every urgency increase and the additional actions that should be taken. The administrator may define different urgency strategy tables for different types of alerts. For example, we could define the urgency values from the ordered set {*Urgent, Very Urgent, Critical, Very Critical*} and a default *urgency function* as follows:

$$U_{002}(t) = \begin{cases} \text{Urgent} & t \le \text{T (default)} \\ \text{Very Urgent} & \text{T} < t \le \text{T} + \text{dt}_1 \\ \text{Critical} & \text{T} + \text{dt}_1 < t \le \text{T} + \text{dt}_1 + \text{dt}_2 \\ \text{Very Critical} & \text{T} + \text{dt}_1 + \text{dt}_2 < t \le \text{T} + \text{dt}_1 + \text{dt}_2 + \text{dt}_3 \end{cases}$$

Table 1. Example Urgency Strategy Table

Urgency002	**Action**
Urgent	Default
Very Urgent	Submit a second alert to the tutor, notifying about the approaching deadline
Critical	Redirect the alert to another tutor that has the best expected response time
Very Critical	Send the alert to several tutors and accept the results of the one that response first, notify an administrator

Table 1 shows an example of urgency strategy table. Here, let us consider the association of an alert with this table. The default level for the alert is *Urgent*. When the priority increases to *Very Urgent* because the selected tutor provides no response, the AMS creates another alert message to notify the tutor about the eminent deadline. If there is still no response, the AMS tries another tutor of the same subject who has the best expected response time. If this step also fails, the priority further increases to *Very Critical*, where several tutors of the subject will receive the alert, while an administrator is notified.

Applicability and Discussions

In this section, we discuss the applicability of the AMS to distance education in the perspective of various stakeholders through our case study, focusing on communications management aspects. We also discuss from the perspective of the system developer and administrator. Then, we discuss the configuration and usability issues with a focus on the limitations of this study.

Students' Perspective

As most students are part-time and mobile, their main requirement is the ubiquitous availability of flexible, anytime, anyplace help from tutors, though quality responses are often more important than the response time. The AMS helps route their questions to the tutors and also helps find alternative tutors for assistance. Similarly, the AMS can facilitate ubiquitous communications among students for peer-to-peer collaborative learning (Yang, 2005).

The AMS maintains other messages for the students and enables ubiquitous access to these messages through different devices. For example, as tutorial classes may not have a fixed venue due to various constraints, students (as well as tutors) have to be reminded on the tutorial times and places. Similarly, the AMS can maintain different kinds of notifications and reminders, such as pickup of the assignments or course materials, assignment due date, and examination venues. The AMS can also maintain the acknowledgments regarding the receipt of assignments, scores, and results, as well as the tutor's comments on the assignments, etc.

Upon exceptions (Chiu et al., 2001), students can also use the AMS to communicate with the course coordinator for special arrangements, such as extending assignment deadlines, attending another tutorial class, changing the tutorial group, or even deferring examinations and studies. With improved communication facilities for students, we expect the students can learn more effectively and more efficiently.

Tutors' Perspective

Tutors in the Mainland are usually lecturers in local universities or professionals in related subjects, living in nearby cities. They have to closely communicate with their students, their corresponding partner institution, as well as the course coordinator. As they are working part-time for distance education, phone tutoring hours need to have restrictions since part-time tutors are not always available after the assigned office hours. Therefore, they have the similar ubiquitous communication requirements as the students. For example, acknowledgements managed by the AMS can avoid failures in receiving students' assignments due to lost mails.

Further, the AMS help keep track of the messages so that more attentions will be paid to important or urgent messages and overwhelmed by the large amount of messages from the students can be avoided. In addition to student profiles (such as past scores), the AMS provides a record of all the students' messages and can therefore help track the students' study progress and performance. This also enables the tutors to answer students' queries more effectively.

The students' requests for consultation usually become more urgent towards assignment deadline and examinations. The AMS now makes it possible to effectively arrange alternative tutors for attending these requests, should a tutor fail to respond within a reasonable amount of time.

Course Coordinators' Perspective

Course coordinators are full-time academics in Hong Kong. They manage a number of courses concurrently. However, they may not be familiar with the operations in the Mainland. They are responsible for the development of the whole course, such as course material, assignments, marking guides, tutorial agendas, examinations, etc. The AMS therefore helps them organize and schedule tutor meetings, day schools, and tutorials as well as manage the messages from all other parties. For example, students' applications for assignment extensions are usually near the deadline and are therefore urgent. Correction in the course material, assignments, and even examination questions represents an increasing degree of urgency. For example, problems in an examination paper have to be handled immediately. On the other hand, handling students' deferment or withdrawal of study is very important and has to be tracked.

Another main contribution of the AMS to the course coordinator is to monitor the performance of the tutors and the students. For example, tutors have to mark assignments and return comments to the students within a short period of time. Alerts can raise the attention to marking lateness. Alerts can also raise the attention to those students who often submit their assignments late or have low accumulative assignment scores. In this way, problems can be discovered early for remedies.

Partner Institutions' Perspective

Partner institutions are educational institutions in various Mainland cities. They conduct local administrative work, such as the recruitment of local students and tutors. They also distribute course materials and assignments, arrange for tutorial classroom, and organize day schools and other activities for the students. Therefore, the AMS mainly coordinates the partner institutions' communication with the students and the tutors, especially upon exceptions such as changes in tutorial venue and tutor absences or substitutions. In addition, by forming a tutor pool to answer student queries, we can expect a better response time and a better substitution upon exceptions. The AMS may also help build the software for operating the hot lines or call centers for the students.

Because the partner institutions monitor the tutors' performance such as the quality of the assignment marking and collect feedback from students and tutors regarding the course, the AMS can pass the alerts of unsatisfactory or exceptional results as well as complaints to the course coordinators and other administration staff. Other exceptions such as resignations of tutors, reallocation of students to other partner institutions, and withdrawal of studies can be reported and tracked in the same way.

Administrators' Perspective

The Shenzhen office in Mainland has full-time administrative staff for coordinating different partner institutions in Mainland and communicating with the course coordinators and administrators at the Hong Kong head office. They publish course materials, assignments, and examinations for all the Mainland operations. They conduct tutor meetings and seminars as well as manage the overall payment to the tutors based on their workload.

The Hong Kong head office is responsible for the overall management. It keeps all the records of the students and the tutors, issues certificates, and liaises with the course coordinators. For example the administrators often need to organize meetings for standardization and award issues.

In addition to ubiquitous communications management, the administrators need data integration with the partner institutes. The AMS can help enforce deadline for information submission and can coordinate data amendment requests such as score amendment and student personal information update.

Management's Perspective

A major concern of the management is the costs against the benefits of the new system. In particular, if any of the improvements as discussed in the previous subsections can significantly help improve the productivities of the tutors and staff or help the students learn more effectively, the costs can be justified. Our approach provides tangible benefit for organizations by the improved communications and services, which also indirectly improves the relationships among the students, tutors, and staff.

The disparity of heterogeneous organizational applications and distance education platforms has created inflexible boundaries for communicating and sharing information and services among the stakeholders. Therefore, the AMS provides a standardized way for managing communications as well as an underlying mechanism for information and process integration. With an AMS mechanism built on top of existing systems and facilities, cost reduction on program, new features, and even the elicitation of management knowledge can be realized. Also, expertise to handle practical exceptions in a timely manner can be incorporated into the AMS. Through software reuse, a reduction in not only the total development cost but also training and support cost can be achieved.

System Developer's Perspective

System developers are often concerned about the system development cost and subsequent maintenance efforts. These concerns can be addressed by systematic fine-grained requirements elicitation of all the stakeholders. With effective support of an AMS, the workflows involved in communications can be simplified because the urgency requirements, associated communications, and monitoring required can be systematically captured into an AMS, instead of scattering in different systems. Thus, the overall system complexity can be managed using the loosely coupled software modules that orchestrate the effective AMS communications. This can also significantly shorten the system development time and meet the high management expectations.

Recent advances in technologies have resulted in fast evolving mobile device models and standards. Our approach can further help reduce uncertainties through adequate testing and experimentations of new technologies, taking advantage of loosely coupled system modules. Upgraded systems that support alerts through the AMS can be gradually integrated instead of a sudden switch-over. Moreover, a distributed solution favors not only privacy but also scalability because we are expecting constant growth in the number of partner institutions and students. It further eases the programming of captured knowledge as explained in the previous subsections.

Configuration and Usability

While the use of urgency strategy tables enables very flexible configuration for the evolution of alerts and related actions, it is not obvious how suitable parameters (e.g., time between elevations) can be. Although the above example illustrates how existing manual policies can be translated into program configuration, but such policies may not exist. Therefore, cautious experimentations with parameter values based on current experiences and/or estimations should be carried out first. However, as the AMS logs all the requests and responses, analysis of such results can be easily facilitated, which provides hints to the adjustments of such values. In addition, the general urgency of requests and requirements on response time vary according to different situations and events of a subject. For example, according to our experience, time requirements are usually tighter for assignment deadlines and examinations, and students normally expect immediate response to queries about examination questions. Therefore, different urgency tables are required to cater for different situations. This is one direction of our continuing research.

As for the main issue of usability, we have to make sure that human participants are not overwhelmed by excessive alert messages. This is necessary to retain the awareness, accessibility and responsiveness as introduced earlier, especially because many participants (tutors and students) are part-time. While the AMS can effectively limit the frequency of alerts at different urgency levels for a particular user, the tolerance level depends not only on the personality of the participant, but also on the participants' *context* (Hong et al.; 2007; Yang, 2005; Schilit et al., 1994). Context refers to the set of environmental states and settings that either determines an application's behavior or seems interesting to the user. The application of context-awareness on alert-based communication management is another direction of our continuing research based on our previous work on adaptive workflow monitoring (Wang et al., 2005). Besides limiting the alerts on the AMS, future possibility is to employ agents at the participants' devices to filter and manage the alerts (Chiu et al., 2003) according to the participants' context and preferences.

Finally, the distance learning platform has to validate requests, especially those initiated by students, before passing them to the AMS to avoid misuse. For example, quota or credit systems as well as complaint and reward systems may help reduce the misuse or overuse of high urgency requests (by requiring a higher credit for more urgent requests). Further studies are required to figure out user behaviors towards these issues and integrate the behavior issues in the setting of relevant parameters.

Conclusions

In this paper, we have looked into the problem of supporting and managing ubiquitous communications under urgency constraint for distance education. We have proposed a framework of an alert management system (AMS) that supports communications of both human (e.g., educators) and Web Services (e.g., partner institution) providers. This framework introduces a flexible alert conceptual model that allows administrators to specify tasks, alerts, roles, and their inter-relations. We have also presented our AMS architecture and described the alert monitoring and routing mechanisms involved. We have further proposed a matchmaking approach, which can be customized for different urgency strategies for different application requirements. As such, an effective AMS can be built and plugged into various distance education systems that require sophisticated ubiquitous communication support. By using AMSs, partner institutions, tutors, and the administrative offices, can form a *learning grid* (Gentzsch, 2002) to improve education service quality while cutting down operation costs at the same time. We have illustrated the applicability of our approach with a detailed case study of the OUHK, and explained the key advantages and limitations with respect to the main stakeholders of distance education.

It should be noted that an AMS targets for urgent, asynchronous, unstructured, or even ad-hoc tasks (such as exception handling). Therefore, it is complimentary to conventional workflow management systems (WFMS) that target at regular synchronous task flows. In fact, the motivation of AMS evolves from the exception handling and user-interface mechanisms of our ME-ADOME WFMS (Chiu et al., 2002) by factoring out and extending, in particular, urgency requirements. The physical execution of individual tasks of regular processes is outside the scope of the AMS and is captured in the application logic of individual information systems (as illustrated in *Figure 4*), which can be a WFMS as well.

For continuing and future work, we first focus on the usability study, based on the feedback of the students and tutors as evaluation, while we are working on the integration of the AMS to the existing web-based learning platform. We are exploring various settings of the urgency tables in cooperation with the administrators. We are also investigating the application of context-awareness in ubiquitous communication management, based on our experience in enterprise services (Hong et al., 2007). Moreover, the complexity involved in the communication processes will be further analyzed and managed based on our previous work on an adaptive approach to complex process management in dynamic situations (Wang and Wang, 2006b). On the other hand, we are also interested in further possibilities in user communication management with agents (Chiu et al., 2003). We are also interested in the impact of cancellations, other possible exceptions, and the tradeoff between quality and costs.

Acknowledgement

This research is partially supported by a UGC CERG research grant (No. RGC/HKU7169/07E) from the Hong Kong SAR Government.

References

Ammenwerth, E., Buchauer, A., Bludau, B., & Haux, R. (2000). Mobile information and communication tools in the hospital. *International Journal of Medical Informatics*, *57* (1), 21-40.

Bates, A. W. (2005). *Technology, E-Learning and Distance Education (Studies in Distance Education)*, 2nd Ed., London: RoutledgeFalmer.

Chang, B., Lim, M., Ham, D.-H., Lee, Y., Kang, S., & Cha, J. (2007). Supporting Multi-device for Ubiquitous Learning. *Lecture Notes in Computer Science, 4469*, 457-467.

Cesarini, M., Monga, M., & Tedesco, R. (2004). Carrying on the e-learning process with a workflow management engine. *Paper presented at the ACM symposium on applied computing*, March 14-17, 2004, Nicosia, Cyprus.

Chiu, D.K.W., Cheung, S.C., & Kafeza, E. (2002). Three-tier View-based Support for Mobile Workflow. *Paper presented at the 1st International Conference on Mobile Business*, July 2002, Athens, Greece.

Chiu, D.K.W., Cheung, S.C., Kafeza, E., & Leung, H.-F. (2003). A Three-Tier View Methodology for adapting M-services. *IEEE Transactions on Systems, Man, and Cybernetics, Part A, 33* (6), 725-741.

Chiu, D.K.W., & Choi, S.P.M. (2005). Alert Driven Communications Management for Distance Learning. *Paper presented at the 2005 IEEE International Conference on e-Technology, e-Commerce and e-Services* (EEE05), March 29 – April 1, 2005, Hong Kong.

Chiu, D.K.W., Kwok, B., Wong, R., Kafeza, E., & Cheung, S.C. (2004). Alert Driven E-Services Management. *Paper presented at the HICSS37***,** January 5-8, 2004, Hawaii.

Chiu, D.K.W., Li, Q., & Karlapalem, K. (1999). A Meta Modeling Approach for Workflow Management System Supporting Exception Handling. *Information Systems*, 24 (2), 159-184.

Chiu, D.K.W., Li, Q., & Karlapalem, K. (2001). Web Interface-Driven Cooperative Exception Handling in ADOME Workflow Management System. *Information Systems*, 26 (2), 93-120.

Eisenstadt, S., Wagner, M., Hogan, Pankaskie, W., Tsui, F-C., & Wilbright, W. (1998). Mobile workers in healthcare and their information needs: are 2-way pagers the answer? *Paper presented at the AMIA Annual Fall Symposium*, November 7-11, 1998, Orlando, FL.

El-Bishouty, M. M., Ogata, H., & Yano, Y. (2007). PERKAM: Personalized Knowledge Awareness Map for Computer Supported Ubiquitous Learning. *Educational Technology & Society, 10* (3), 122-134.

Fano, A. & Gershman, A. (2002). The Future of Business Services in the age of Ubiquitous Computing. *Communications of the ACM, 45* (12), 83-87.

Gentzsch, W. (2002). Grid Computing: a New Technology for the Advanced Web. *Lecture Notes in Computer Science, 2326*, 1-15.

Hong, D., Chiu, D.K.W., Cheung, S.C., Shen, V.Y., & Kafeza, E. (2007). Ubiquitous Enterprise Service Adaptations Based on Contextual User Behavior. *Information Systems Frontier, 9* (4), 343-358.

Hripcsak, G., Clayton, P., Jenders, R.A., Cimino, J.J., & Johnson, S.B. (1996). Design of a Clinical Event Monitor. *Computers and Biomedical Research, 29*, 194-221.

Joiner, R., Nethercott, J., Hull, R., & Reid, J. (2006). Designing educational experiences using ubiquitous technology. *Computers in Human Behavior, 22* (1), 67-76.

Kajita, S., & Mase, K. (2006). uClassroom: Expanding Awareness in Classroom to Ubiquitous Teaching and Learning. *Proceedings of the 4th IEEE International Workshop on Wireless, Mobile and Ubiquitous Technology in Education* (pp.161-163), Los Alamitos: IEEE Computer Society.

Kafeza, E., Chiu, D.K.W., Cheung, S.C., & Kafeza, M. (2004). Alerts in Mobile Healthcare Applications: Requirements and Pilot Study. *IEEE Transactions on Information Technologies for Biomedicine*, *8* (2), 173-181.

Lin, Y.-B., & Chlamtac, I. (2000). *Wireless and Mobile Network Architectures*, John Wiley & Sons.

Milrad, M., & Spikol, D. (2007). Anytime, Anywhere Learning Supported by Smart Phones: Experiences and Results from the MUSIS Project. *Educational Technology & Society, 10* (4), 62-70.

Ride, D.M., Safran, C., Philips, R.S., Wang, Q., Calkins, D.R., Delbanco, T.L., Bleich, H.L., & Slack, W.V. (1994) Effect of Computer Based Alerts on the Treatment and Outcomes of Hospitalized Patients. *Archives of Internal Medicine, 154*, 1511-1517.

Sakamura, K., & Koshizuka, N. (2005) Ubiquitous computing technologies for ubiquitous learning. *Proceedings of the IEEE International Workshop on Wireless and Mobile Technologies in Education* (pp.11-20), Los Alamitos: IEEE Computer Society.

Schilit, B. N., Adams, N., & Want, R. (1994). Context-aware computing applications. *Proceedings of the IEEE Workshop on Mobile Computing Systems and Applications* (pp.85-90), Santa Cruz, CA: IEEE.

Sheng, O. R. L. & Chen, G. H. M. (1990). Information Management in Hospitals: An Integrating Approach. *Paper presented at the Ninth Annual International Phoenix Conference on Computers and Communications*, March 21-23, 1990, Scottsdale, AZ, USA.

Sheth, A. & Larson, J. (1990). Federated database systems. *ACM Computing Surveys, 22* (3), 183-236.

Suomi, R. & Tähkäpää, J. (2003). Establishing a Contact Centre for Public Health Care. *Paper presented at the 36th Hawaii International Conference on System Science*, January 6-9, 2003, Hawaii.

Tuckman, B.W. (2005) The effect of motivational scaffolding on procrastinators' distance learning outcomes. *Computers & Education, 49* (2), 414-422.

Yang, S. J. H. (2006). Context Aware Ubiquitous Learning Environments for Peer-to-Peer Collaborative Learning. *Educational Technology & Society, 9* (1), 188-201.

Wang, M., & Wang, H. (2006a). From Process Logic to Business Logic - A Cognitive Approach to Business Process Management. *Information & Management, 43* (2), 179-193.

Wang, M., & Wang, H. (2006b). Enhancing Interaction with Dynamic Environments: An Adaptive Approach to Process Management. *International Journal of Technology and Human Interaction, 4* (2), 37-53.

Wang, M., Wang, H., and Xu, D. (2005). The Design of Intelligent Workflow Monitoring with Agent Technology. *Knowledge-Based Systems, 18* (6), 257-266.

Wang, M. and Wang, H. (2002). Intelligent Agents Supported Flexible Workflow Monitoring System. *Lecture Notes in Computer Science, 2348*, 787-791.

Weverka, P. (2000). *Mastering ICQ: The Official Guide*, IDG Books.

Weiser, M. (1991) The computer for the 21st century. *Mobile Computing and Communications Review*, *3* (3), 94-104.

Zhang, G., Jin, Q., & Lin, M. (2005). A framework of social interaction support for ubiquitous learning, *Paper presented at the 19th International Conference on Advanced Information Networking and Applications*, March 28-30, 2005, Taipei, Taiwan.