

Real Time Dynamic Mobile Scheduling System

Thesis proposal

For the degree of Master of Science in Computer Science

At Southern Connecticut State University

RamaKrishna Kolli

July 2003

Thesis Advisor:Dr.Hrvoje Podnar

Major-Field Approval – The advisor and the department chairperson

Advisor

Date

Chairperson

Date

A. Title

3-Tier Real Time Dynamic Mobile Scheduling System.

B. Statement of Purpose

Traditional applications have not accounted for the fact that employees leave their desks. Effective mobile computing aims to solve this problem by delivering the information users require remotely. This approach is usually followed by increase in their productivity.

One application that can be used is remote scheduler. Scheduler permits easier scheduling of activities and meetings involving individuals from other departments in the organization. It also provides an audit trail for verifying productivity of each individual. Implementing scheduling system on mobile phones would help individuals to schedule the meetings instantaneously. The goal of this project is to develop and implement meeting scheduler application for mobile phones that allows temporal assignment of meetings to resources (e.g. individuals and rooms) where a number of conditions have to be considered. The project focuses on the predictive scheduling and the correction of an existing schedule due to actual events in the planning-scheduling environment (reactive scheduling or rescheduling). In order to utilize the scheduling system the user interface should be developed using J2ME (Java™ 2 Platform, Micro Edition) technology. This technology provides a range of configurations and virtual machine technologies optimized for embedded application. Algorithms and methods to be used are required to conform to mobile phone specifications.

A scheduling problem can be described by the tuple (R, HC, SC) with:

- a set of resources $R = \{R_1, \dots, R_r\}$ like rooms, individuals.
- a set of hard constraints $HC = \{H_1, \dots, H_h\}$ (e.g. high priority meetings), that have to be fulfilled, and

- a set of soft constraints $SC = \{S_1, \dots, S_s\}$ (e.g. meeting due dates), that should be fulfilled but may be relaxed.

Result of scheduling is a schedule (or plan) showing the temporal assignment of operations. This system will support a scheduling algorithm and tools required for information retrieval and processing. The platform should also support utilities for data analysis and performance measurement.

Scheduling and, in particular, rescheduling are such difficult problems that no today's algorithmic approach can solve in a satisfactory manner and in reasonable time. Therefore, the objective of system approaches should not be the replacement of the human experts, but their support in order to extend their problem solving capabilities. Thus a computer-based scheduling system supporting the meeting planner in a user-friendly interactive manner should combine a graphical user interface with the scheduling method.

Different kinds of knowledge are used in solving a scheduling problem (Sauer,1993). This solution methods can be divided into the areas of domain, situation, scheduling, and meta-scheduling. The domain knowledge contains static information about the application environment, which is reflected in the structure of the scheduling problem, e.g. the set of groups and the available resources. The situation knowledge represents the current state of scheduling, e.g. the existing schedule, the remaining capacities of resources etc. The scheduling knowledge is divided into static and dynamic approaches for scheduling and rescheduling. The static scheduling knowledge contains complete algorithms, e.g. operations research algorithms but also knowledge based ones. These algorithms are firmly implemented into the scheduling structure. The dynamic scheduling knowledge denotes chunks of heuristic planning knowledge, which can be used together or separately in finding solutions to scheduling problems. It provides the

possibility for adequate description and integration of multiple algorithmic approaches in one system. The meta-scheduling knowledge contains the information necessary to determine the “best” algorithm to the current scheduling problem. This includes the decision on which algorithms are applicable for which tasks (goals) and which ones are appropriate for which events of orders to the machines that shall be used (Sauer,1994).

With any mobile solution, remote access to data is useless if the data is not accurate and timely (Pop Art, 2002). Currently, the United States is in its second generation, called 2G wireless networks. 2G devices typically use circuit-switched technology so that the device only connects to the network when needed. 2G protocols include CDMA, TDMA and GSM. Data transfer ranging from 9.6 to 19.6 kilobits per second.

The amount of time that handheld device is transmitting information should be kept to the minimum. Long transmission are prone to errors and certainly will reduce the battery life.

The component list for this application includes many entities:

- The “*User-Interface*” provides a user-friendly and problem adequate graphical presentation of the information currently in the knowledge base, e.g. a Gantt chart representation of the existing schedule, and multiple tools for interaction. Final code could be implemented using Oracle9i JDeveloper that has seamlessly integrated the J2ME technology. The J2ME extension allows you to reach and use any J2ME toolkit (Oracle9iDS Jdeveloper,2002).
- The “*Meta-Scheduling*” component realizes the selection and interpretation of the appropriate scheduling strategies. The new scheduling problem is analyzed, the appropriate skeletons are identified and refined to solution algorithms, the alternative algorithms (static as well as dynamically created ones) are judged and the applicable approaches are ranked with respect to their appropriateness for the current problem.

To identify appropriate heuristic the input data are compared against the predetermined list of constraints. The “best heuristic” (where most of the constraints are met) is applied first. If the interpretation of a heuristic succeeds, a new plan is generated. If the interpretation of a heuristic fails, the next heuristic is chosen until a solution is found or no appropriate heuristic is available. Because of the mobile device memory limitations all the heuristics will be implemented and performed on the server side.

C. Literature Review And Current State-Of-The-Art:

Traditional scheduling systems like Lotus notes (IBM Lotus Notes,2002), Outlook (Mac Outlook, 2001) exist with Client-Server architecture in the market. These systems are not dynamic in nature in the sense that, the schedule does not adjust automatically based on the work/meeting/individual priority. Lotus Notes 6 displays the conflict if any, but does not try to resolve the conflict based on the static information available like priority. Most importantly, these systems are not available on the go where you cannot access the desktop or laptop.

Mobile schedulers that are available in the market are personal schedulers and do not share the information real time. It seems that a majority of scheduler applications including traditional scheduling systems are enterprise products and are not publicly available.

Dynamic schedulers with mobile technology are currently not available in the market. It seems that a majority of scheduler applications developed are non-dynamic.

D. Methodology

The proposed project will combine complex 3-tier architecture depicted in Figure 1. The first tier consists of a MIDlets installed on mobile phones. The second tier consists of server side application programs implemented in JAVA and running on TOMCAT servlet engine. The third

tier utilizes ORACLE (Oracle9iAS Wireless, 2002) relational database with JDBC connection calls between server side application and the Database.

The Java 2 Platform Micro Edition, (J2ME) Wireless Toolkit (Sun Microsystems, 2002) is a set of tools that provide application developers with the emulation environments, documentation and examples needed to develop Java technology applications targeted at Connected Limited Device Configuration /Mobile Information Device Profile compliant mobile phones.

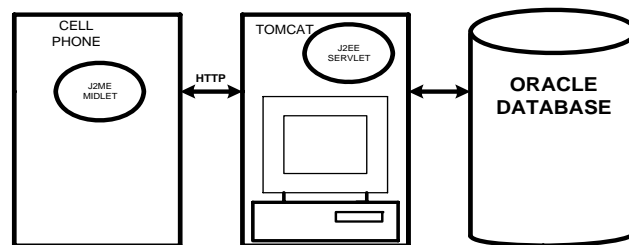


Figure 1. Pictorial representation of the 3-tier architecture.

Writing a world-aware application that runs in the stripped-down environment of J2ME (Muchow, W. J. 2001) can be a real challenge because most of the J2SE(Java™ 2 Platform, Standard Edition) support classes simply aren't in J2ME. It's important to understand how MIDlets can use HTTP to talk to a TOMCAT server (Giguere, 2000) and to remember that with J2ME, memory is always at a premium—the device might have very little space available for storing and running Java applications. This issue will be resolved by implementing the business rules on the server side. The MIDlets call the functions in the TOMCAT with parameters and based on these parameters server application derives the appropriate function to execute.

The scheduling system takes as input, a set of participants, their user preferences, and a set of Calendar slots across a time range. The scheduling system would provide the following functions to help the initiator resolve conflicts:

- A function to compute if everyone is available for a meeting on a particular day.
- A function to find the most preferred date within date range.

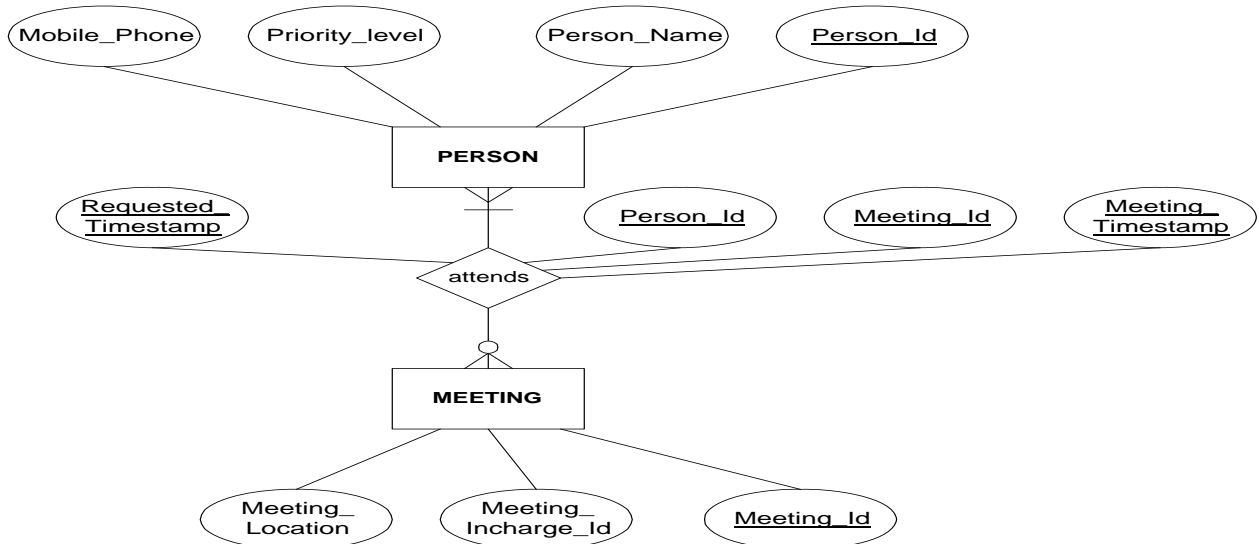


Figure 2. Mobile Real time Scheduler Entity-Relationship diagram (Hoffer, J. A., & Prescott, M. B., & McFadden, F. 2002).

Figure 2 shows a detailed, logical representation of the data for the meeting scheduler business environment. This Entity-Relationship model is expressed in terms of entities in the business environment, the relationships (or associations) among those entities, and the attributes (or properties) of both the entities and their relationships. Entities are represented by the rectangle symbol, while relationships between entities are represented by the diamond symbol connected by the lines to the related entities. Each entity type has a set of attributes associated with it. An attribute is a property or characteristic of an entity type that is of interest to the organization. Placing its name in an ellipse with a line connecting it to its associated entity represents attribute.

The entities in Figure 2 are:

PERSON - A person in the organization who initiated the meeting or attends the meeting.

Attributes of PERSON are person id, person name, mobile phone, and priority level

MEETING – Group of individuals scheduled for the meeting.

Attributes of MEETING are meeting id, location, and in-charge or person initiated.

E. Contributions

Majority of scheduling applications are static. This means that if resources are not available user intervention is needed to resolve the conflict. There are some scheduling systems that perform real time dynamic scheduling .If resources are not available the system proceeds with rescheduling algorithm. Because of immense computing demand of those systems usually run only on the powerful servers without remote input.

The Real Time Dynamic Mobile Scheduling System will provide remote capability for managing scheduling needs. This in turn will increase employee productivity, performance, efficiency and satisfaction. Given the today's boom in wireless technology it is expected that a RTDMS system will be utilized frequently.

F. References

Giguere, E. (2000). Making HTTP connections with MIDP.

Web site: <http://wireless.java.sun.com/midp/ttips/httpcon>

Jeffrey Hoffer (2002). The Database Environment in Prentice Hall 6E *Modern Database*

Management (Ch 1-5). Copy Right: 2002 Prentice Hall

IBM Lotus Notes.(2002) (n. d.) Calendaring & Scheduling for Lotus Notes 6 -- "seeing is knowing". Retrieved June 18, 2003, from

Web site:

<http://www.lotus.com/news/news.nsf/0/7042369bf2c9130a85256c3d005a039b?OpenDocument>

Mac Outlook. (2001) (n. d.) Calendar/Scheduler and Tasks. Retrieved June 18, 2003, from

Web site: http://www.med.wayne.edu/msis/computing/software/mac/mac_sch.pdf

Muchow, W. J. (2001). *Core J2ME Technology & MIDP*. (1st ed.) Prentice Hall PTR.

Oracle9iAS Wireless (2002). *Creating a Mobilized Business An Oracle White Paper*. Retrieved June 18, 2003, from

Web site: http://technet.oracle.com/products/iaswe/pdf/9iaswe_twp.pdf

Oracle9iDS Jdeveloper. (2002) - *J2ME Extension*. Retrieved June 18, 2003, from

Web site: <http://otn.oracle.com/products/ids/daily/jul11.htm>

Pop Art, Inc (2002). *White Paper: Mobile/Wireless Services*. Retrieved June 18, 2003, from

Web site: <http://www.popart.com/whitepapers/wireless-applications.pdf>

Rajpathak, D., & Motta, E., & Roy, R. (2001). *A Generic Task Ontology for Scheduling Applications*.

Web site: <http://kmi.open.ac.uk/people/dnyanesh/Publications/IC-AI'2001-1/finalversion.pdf>

Sauer, J. (1993). *Meta-Scheduling using Dynamic Schedule Knowledge*.

Web site: <http://www-is.informatik.uni-oldenburg.de/publications/669.pdf>

Sauer, J. (1994). *Scheduling Meta-Scheduling*.

Web site: <http://www-is.informatik.uni-oldenburg.de/publications/667.pdf>

Sun Microsystems. (2002). *Java 2 Micro Edition, Wireless Toolkit*, Sun Microsystems

Web site: <http://java.sun.com/products/j2mewtoolkit>