Pervasive and Mobile Computing 4 (2008) 719–736

Contents lists available at ScienceDirect

Pervasive and Mobile Computing

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

An application framework for mobile, context-aware trails

Cormac Driver *, Siobhan Clarke

Distributed Systems Group, School of Computer Science and Statistics, Trinity College Dublin, Ireland

ARTICLE INFO

Article history:
Received 25 September 2007
Received in revised form 16 April 2008
Accepted 21 April 2008
Available online 5 May 2008

Keywords:
Trails
Context-aware scheduling
Pervasive computing
Software frameworks

ABSTRACT

In this paper we describe the design, implementation and evaluation of a software framework that supports the development of mobile, context-aware *trails*-based applications. A trail is a contextually scheduled collection of activities and represents a generic model that can be used to satisfy the activity management requirements of a wide range of context-based time management applications. Trails overcome limitations with traditional time management techniques based on static to-do lists by dynamically reordering activities based on emergent context.

1. Introduction

Time management strategies for planning and scheduling activities increase the efficiency and effectiveness of either personal or corporate time use. Supporting techniques are commonly based on the use of prioritised to-do lists [15]. A prioritised to-do list is an ordered collection of activities that the list owner must complete, with tasks crossed off as they are completed. While the use of to-do lists for time management is beneficial, their static nature reduces their effectiveness in dynamic environments where users are mobile and activity properties can change over time. The prioritised ordering of a carefully considered, predefined to-do list can quickly become obsolete as its owner begins addressing activities and unforeseen events occur.

Mobile context-aware computing is a computing paradigm in which applications can discover and take advantage of contextual information (such as user location, time of day, nearby people/computing devices and user activity) [44]. Example applications of mobile, context-aware computing include location-aware telephone call forwarding [53], situation-aware self-managing mobile phones [46], context-aware medication monitors [2] and weather-aware clothes hanger-based information displays [37]. Mobile devices can acquire context information from on-board local sensors such as GPS devices, accelerometers and infrared sensors, and from remote sensors that transmit data about the environment in which they are embedded e.g., temperature and pulse rate data can be acquired from sensors implanted in the human body. Mobile, context-aware applications automatically adapt to discovered context by changing their behaviour as appropriate to better suit the user. This paradigm facilitates the automatic adaptation of a mobile user's schedule of activities so that it accurately reflects the reality in which the user exists and maintains utility despite the occurrence of unforeseen events. Automatic, context-based schedule adaptation is at the core of a range of applications for the mobile user who has a set of activities that may or should be carried out throughout the day at different locations.

Implementation of a mobile, context-aware activity scheduling application involves addressing two challenges common to this type of application. First, an application must be capable of automatically ordering a list of activities in an effective manner with respect to relevant context. Existing approaches to mobile, context-based activity ordering are either constrained in the number of activities they can cope with because of device limitations, or are server-based and subject

* Corresponding author. Tel.: +353 18961543.
E-mail addresses: cormac.driver@cs.tcd.ie (C. Driver), siobhan.clarke@cs.tcd.ie (S. Clarke).

1574-1192/$ – see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.pmcj.2008.04.009

to wireless network disconnection. Second, an application must be capable of identifying when it is necessary to reorder a list of activities to ensure that the list order maintains utility in the face of context change. A balance is required between avoiding the execution of unnecessary reordering processing (a resource-intensive task) on a resource-constrained mobile device and ensuring the list order is always the 'best' one. Existing techniques for identifying when it is necessary to reorder a list of activities are generally based on periodically assessing the ordering. This approach raises the possibility of an activity ordering becoming temporarily out of sync with the user's reality.

To date, mobile, context-based activity scheduling applications have typically been designed and implemented in an application-specific manner. Consequently, developers have had to repeatedly tackle the challenges inherent to this class of application, hindering progress in areas that researchers are primarily interested in such as application deployment and user evaluation. In this paper we present an application framework for the development of mobile, context-aware *trails*-based applications [14]. A trail is a contextually scheduled collection of activities and represents a generic model that can be used to satisfy the activity management requirements of a wide range of context-based activity scheduling applications. The framework provides structure and behaviour to support context-based activity schedule composition (a process know as *trail generation*), identification of whether or not schedule reordering is required following context change (*trail reconfiguration point identification*) and subsequent automatic schedule reordering as appropriate (*trail reconfiguration*). The framework supports the development of trails-based applications in a generic, extensible manner, enabling developers to both reuse common application components and extend the framework to support application-specific behaviour. Consequently, the development of a diverse range of trails-based applications is more accessible to software developers.

The remainder of this paper is organised as follows. Section 2 presents a short perspective on the trails concept and illustrates its applicability. Section 3 presents our application framework for mobile, context-aware activity scheduling. Section 4 describes the evaluation of the framework in terms of (a) reusability and extensibility (b) performance and (c) human opinion on trail quality. Section 5 reviews related work and Section 6 provides a summary.

2. A perspective on trails

The relevance of trails-based applications is evidenced by the existence of many commonly assumed business-related roles in which activity scheduling in a dynamic environment is an inherent requirement. Context-based activity scheduling is an aspect of the work conducted by individuals in workplaces such as hospitals (scheduling patient rounds and administrative tasks), warehouses (managing the order in which requests for items are fulfilled), hotels (managing the order in which rooms are serviced/cleaned) and prisons (managing the order in which inmates are monitored by guards). Those working in professions that involve greater mobility e.g., mobile salespeople and tradespeople (plumbers, electricians, office equipment technicians), on call care givers (doctors, veterinarians), taxi drivers and mobile delivery personnel (parcel/food/flower delivery couriers) also manage their working lives by using relevant context to schedule their pending and emergent activities. Away from the business world, context-based activity scheduling is used informally by many. At the simplest level, people use context to manage their day-to-day activities. The implementation of such an application, a trails-based day planner, is presented in Section 4.1.1 as part of our evaluation. Context-based activity scheduling is also used by people in more specific leisure-related situations such as when sightseeing, attending a music festival, playing treasure hunt-type games, visiting a theme park or going shopping at a particularly busy time e.g., Christmas time. We discuss trails-based applications that support visitors to music festivals and theme parks as part of our evaluation in Sections 4.1.2 and 4.1.3 respectively. In addition, much of the related work analysed in Section 5 is concerned with mobile, context-aware tourist guide applications.

The pervasiveness of mobile devices (particularly the mobile phone) and their recent technical advancement, combined with the trails-based application development support described in this paper, creates an environment in which computer support for context-based activity scheduling can be realised to support users in both business and leisure scenarios.

3. Application framework

A software framework is a reusable implementation of all or part of a software system expressed as a set of classes (some abstract) with behaviours defining the way in which instances of those classes collaborate [42]. The term 'application framework' is used to describe a software framework that constitutes a generic application for a specific domain area [39]. In this section we introduce the Hermes project which is concerned with providing framework support for mobile, context-aware applications in general. We then discuss the challenges in mobile, context-aware trails management in particular and describe in detail how the application framework presented in this paper supports trail generation and trail reconfiguration point identification.

3.1. The Hermes project

The Hermes project[1] at Trinity College Dublin is investigating extensible, generic components for mobile, context-aware applications. A screenshot from one such application, a trails-based mobile game called RiddleHunt developed using the

[1] http://www.dsg.cs.tcd.ie/hermes