Pervasive and Mobile Computing 6 (2010) 281–296

Contents lists available at ScienceDirect

Pervasive and Mobile Computing

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

Context modelling and a context-aware framework for pervasive service creation: A model-driven approach☆

Achilleas Achilleos [a,*], Kun Yang [a], Nektarios Georgalas [b]

[a] Pervasive Systems Research Group, School of Computer Science and Electronic Engineering, University of Essex, Colchester, CO4 3SQ, United Kingdom
[b] Centre for Information and Security Systems, British Telecom Innovate, Ipswich, IP5 3RE, United Kingdom

ARTICLE INFO

Article history:
Received 28 February 2008
Received in revised form 12 April 2009
Accepted 1 July 2009
Available online 19 July 2009

Keywords:
Pervasive service creation
Context-aware framework
Context modelling
Model-driven development
Domain specific modelling

ABSTRACT

Pervasive service creation entails a complex process that involves a diversity of development aspects. Context-awareness is an important facet of pervasive service creation, which deals with the acquisition, rendering, representation and utilisation of context information. In this paper we tackle context-awareness at the application level dealing with the representation and utilisation of context by services. We propose a model-driven approach that facilitates the creation of a context modelling framework and simplifies the design and implementation of pervasive services. To conclude, we demonstrate the benefits of our model-driven approach via the creation of a pervasive museum service and its evaluation using selected software metrics.

1. Introduction

Conventional services represent software applications that can be deployed on a specific device and platform to support the execution of particular computing tasks. In contrast, pervasive services refer to software applications that can operate in a dynamic environment and have the capability to run anytime, anywhere and on any device with minimal user attention [1]. A pervasive service provides users with a specialised and personalised behaviour that allows performing dynamic computing tasks.

One characteristic feature of pervasive services is context-awareness, which depicts the necessity to react in accordance to certain predefined rules or on the basis of intelligent stimulus. This denotes the capability of the service to utilise context information in order to adapt dynamically its behaviour. Context has acquired a variety of meanings over the course of research [2]. In this work, we define context as: "*Any information relevant to the interaction of the user with the service, where both the user and the application's environment are of particular interest*". Understanding which information is termed as context, how to model and utilise them is particularly important in order to simplify the creation of pervasive services.

Service creation is a complex process, which involves a set of activities for the rapid analysis, design, implementation and validation of services [3,4]. The process is usually supported by a service creation framework, which aims to simplify service creation. Many technology-specific frameworks [3,5] have been developed to realise this objective. None of them though provides a clear-cut solution, due to the technology-specific complexities introduced. These frameworks aid technology experts but certainly do not assist novice users [4]. Hence, we argue that a model-driven generic framework [6] is required to provide solutions to these issues.

☆ The work presented in this paper is partly supported by British Telecom under the Model-driven Component-based Systems Engineering (MOSE) project and the UK Engineering and Physical Sciences Research Council (EPSRC) under project PANDA (Policy-based Model-driven Pervasive Service Creation and Adaptation).

* Corresponding author. Tel.: +44 0 78400 65217; fax: +44 0 1206 872900.
E-mail address: aaxilleas79@yahoo.gr (A. Achilleos).

1574-1192/$ – see front matter © 2010 Published by Elsevier B.V.
doi:10.1016/j.pmcj.2009.07.014

When dealing with pervasive services the complexity of the process is augmented due to the diversity of sources from which context information is obtained. In conventional services information comes mainly as input from the user and this manually supplied information drives the service execution. Pervasive services though rely on information that arises from a variety of sources; *e.g. sensors, repositories, users*. Therefore, the capability to effectively represent and manage context must be provided, in order to aid the creation of pervasive services.

Context-aware service creation has been studied during the course of research following two complementary directives. Several approaches [7,8] have been proposed that follow an infrastructure-based solution to the problem. These approaches provide an infrastructure capable of sensing, gathering and processing context information required by the pervasive service [9]. Although the process is simplified, the necessity to tailor the service implementation in accordance to the infrastructure's implementation arises. Consequently, these approaches restrict the developer to a specific implementation technology.

The complementary directive introduces approaches operating at the application level [10]. These approaches do not consider how information is acquired, gathered and processed to obtain an abstract context description. The primary requirement is the representation of context information in a format that can be realised and utilised by context-aware services. In principle these approaches are termed as context modelling techniques [10]. They deal with management tasks such as representation, administration and distribution of context information to services to achieve their adaptation.

Context modelling primarily tackles the representation of context information in the form of an abstract context model. The model is defined via the use of a context modelling framework, which comprises a modelling language and a supporting editor with drag and drop capabilities. Subsequently, the mapping of the modelling language to an implementation technology is defined, to facilitate the transformation of context models. The context model drives the generation of the implementation, which acts as the bridging point (e.g. similar to an API: Application Programming Interface) that allows context to be utilised by services [11]. The generated implementation typically serves tasks for managing a context repository such as querying, administrating and distributing context information to services.

An ideal context model should go head to head with the service creation framework into which it is to be implemented. A common software engineering technology that underpins both context modelling and a pervasive service creation framework can naturally bring context-awareness into pervasive services at the stage of service creation. One such technology is the Model Driven Architecture (MDA) [12,13] paradigm from the Object Management Group (OMG) [14]. In our previous work, a preliminary MDA-based service creation framework has been proposed and verified [15]. The work in this paper follows on our previous research outcomes to bring context-awareness into service creation via a model-driven technology, in particular, OMG's MDA. MDA's many advantageous features such as high-level abstraction and platform independence not only facilitate but also simplify the process of context modelling and its eventual implementation to particular programming languages [16].

The terms pervasive and context-aware are used interchangeably in this work. This is because we exclusively consider context-awareness as the prime characteristic feature of pervasive services. The paper promotes the thought of incorporating context-awareness into pervasive services at the static compile time, i.e., service creation stage. These mechanisms built in at the service creation stage will be triggered at the service execution phase to provide inherent and therefore much enhanced service adaptability. The proposed approach complements the main-stream service adaptation methodology that is largely based on a complicated middleware infrastructure.

In this paper two main technical contributions are introduced. First, we propose a model-driven methodology based on the MDA paradigm that facilitates the service creation process. We utilise MDA for context modelling and consider context modelling as part of the whole process of pervasive service creation. Second, we practise the methodology to design and develop a Context Modelling Framework (CMF) and verify its effectiveness using a pervasive service scenario. Note that the CMF is integrated into the generic framework as one of its components to facilitate the design, validation and implementation of pervasive services.

The remainder of this paper is structured as follows: Section 2 presents related research work on context-aware service creation. Section 3 introduces the model-driven methodology and presents the generic framework's software components and architecture. In Section 4 we perform the necessary requirements analysis of the context domain and introduce the proposed CMF, which is integrated into the generic framework to comprise the Context-Aware Pervasive Service Creation Framework (CA-PSCF). Section 5 demonstrates the applicability of the CA-PSCF for the creation of a pervasive museum service. An evaluation of the approach is also presented in this section using selected software metrics. Finally, in Section 6 we present the conclusions and future work.

2. Related work

Initial efforts on context-aware service creation focused on an infrastructure-based solution to the problem. In their work Dey and Abowd [7] define an architecture and present a Java-based Context Toolkit that simplifies context-aware service creation. The toolkit provides three abstract architectural components namely widgets, interpreters and aggregators. These components are responsible for the acquisition of context information from sensors as raw data and the processing of those data to obtain a high-level representation. As a result, this context information can be utilised by context-aware services to achieve their adaptation.