

Chapter 46

A Mobile Fleet Application Case Study Using SyD Middleware

Janaka Balasooriya

Arizona State University, USA

Sushil K. Prasad

Georgia State University, USA

Michael Weeks

Georgia State University, USA

ABSTRACT

This chapter describes the implementation of a large fleet application based on SyD (System on Mobile Devices), a middleware technology for mobile devices. Although one is made to believe that a collaborative application that runs on a collection of heterogeneous mobile devices can be easily developed with the existing middleware technologies, the reality is that they require too many ad-hoc techniques as well as cumbersome and time-consuming programming. The authors' SyD middleware has a modular architecture that makes such application development very systematic and streamlined. Features of SyD are illustrated here through the prototype application, a complex communication system for a trucking fleet that operates a package delivery system. The fleet system has been implemented employing three technologies, SOAP, JDBC, and SyD. Implementation experience shows that SyD greatly simplifies development and implementation by allowing heterogeneous devices, peer-to-peer communications, group transactions based on event triggers, and mobility support.

1. INTRODUCTION

To develop an application that requires integrating a set of devices with different capabilities (e.g. mobile phone, PDA, PC, and server) with data distributed on these devices requires solving

many problems. For each device, the programming may be different for both data storage and communications; for example, the selection and capabilities of database packages that work with small devices like a PDA is limited. Groups of devices may need to be formed spontaneously,

DOI: 10.4018/978-1-61520-655-1.ch046

and deal with unexpected situations. Finally, weak connectivity means that data may not be available when needed. In short, the application requires the following technical needs to be met:

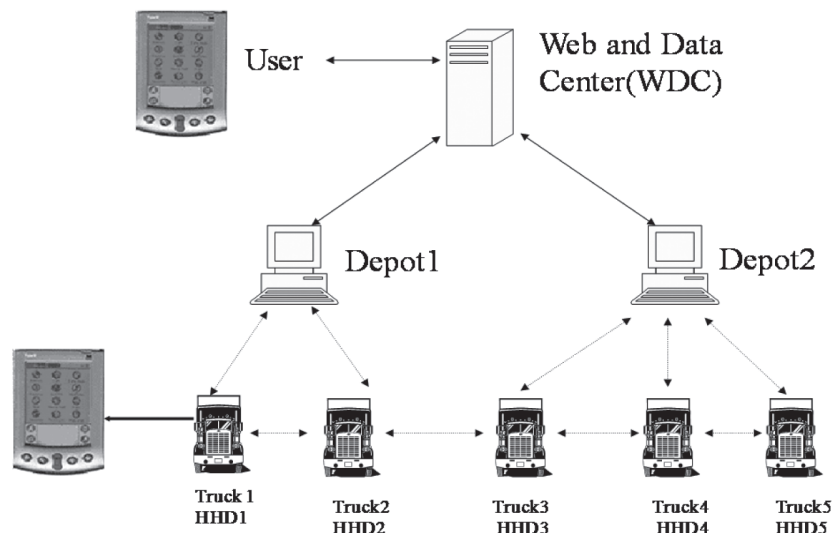
1. Heterogeneous device, data format and language support,
2. Peer-to-Peer computing over dynamic groups,
3. Group transactions: Autonomous Database Synchronization based on triggering events; group querying, and,
4. Mobility support through proxies and directory service.

The current technology for development of a distributed collaborative applications (hereafter we call them collaborative applications) over a heterogeneous set of wired or wireless devices and networks limitations (Bellavista, P., Corradi, A., Giannelli, C., 2009), (Prasad, S. K., Madiseti, V., Navathe, S., B., et al., 2004). Such application development requires tedious programming on each kind of device, both for data access and for data communication. The application code is governed by the type of device, data format, and

the network. The database server is typically a centralized logical entity providing only a fixed set of database services, with little flexibility for user-defined ad hoc services or the ability of user applications to dynamically configure a collection of independent data stores. Applications running across mobile devices are complex because of the lack of persistence of their data due to their weak connectivity.

A practical application that demonstrates the technical problems mentioned above is a truck fleet that operates an automated package delivery system. Figure 1 illustrates the communication architecture of our fleet application. In the fleet application, a user would connect through the Web and Data Center (WDC) web interface, request delivery of a package, and give information pertaining to the package, such as where the package can be picked up, its priority, and where it should be delivered. The WDC passes this on to a depot in the general geographic area of the package, which schedules a pick-up. A truck will arrive according to the schedule set by the depot, and take the package to its next stop, which could be the package's destination, or some intermediate point between the package's source and its destination.

Figure 1. Communication in the fleet system



21 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:

www.igi-global.com/chapter/mobile-fleet-application-case-study/66502

Related Content

Boosting the Competitiveness of Organizations With the Use of Software Engineering

Mirna Muñoz (2022). *Research Anthology on Agile Software, Software Development, and Testing* (pp. 1838-1856).

www.irma-international.org/chapter/boosting-the-competitiveness-of-organizations-with-the-use-of-software-engineering/294547

A Framework for Homogeneous Cross-Project Defect Prediction

Lipika Goel, Mayank Sharma, Sunil Kumar Khatri and D. Damodaran (2021). *International Journal of Software Innovation* (pp. 52-68).

www.irma-international.org/article/a-framework-for-homogeneous-cross-project-defect-prediction/266282

Retrofitting Existing Web Applications with Effective Dynamic Protection Against SQL Injection Attacks

San-Tsai Sun and Konstantin Beznosov (2010). *International Journal of Secure Software Engineering* (pp. 20-40).

www.irma-international.org/article/retrofitting-existing-web-applications-effective/39007

Systems Engineering Modeling and Design

Kumar Saurabh (2009). *Systems Analysis and Design for Advanced Modeling Methods: Best Practices* (pp. 96-114).

www.irma-international.org/chapter/systems-engineering-modeling-design/30017

New Software Methodologies and Techniques for Business Models with Evolutionary Aspects

Hamido Fujita (2008). *Information Systems Engineering: From Data Analysis to Process Networks* (pp. 252-291).

www.irma-international.org/chapter/new-software-methodologies-techniques-business/23419