Load-balancing mechanism for the RFID middleware applications over grid networking

Yi-Wei Ma, Han-Chieh Chao, Jiann-Liang Chen, Cheng-Yen Wu

Department of Engineering Science, National Cheng Kung University, Tainan, Taiwan
Institute of Computer Science & Information Engineering, National Ilan University, Ilan, Taiwan
Department of Electrical Engineering, National Taiwan University of Science and Technology, Taipei, Taiwan
Department of Electronics Engineering, National Dong-Hwa University, Hualien, Taiwan

Abstract

RFID middleware often has to process enormous amounts of data, possibly causes overloading and leading to data errors. Therefore, RFID middleware applications require load-balancing mechanisms. This work proposes a grid-based load-balancing mechanism for RFID middleware applications. The proposed mechanism incorporates functional modules Buffer Management and Load Balancing Management over a grid networking platform, to buffer the read data and share the middleware loading, thereby solving the overloading issues in RFID applications. Performance analysis results show that the proposed mechanism is more efficient than the existing mechanisms. The mechanism has an average improvement of 29.15% for the processing time and improvement of 88.75% for the packet lost ratio, while the number of middleware hosts is fixed. The proposed RFID load-balancing mechanism is indicated to be very reliable.

1. Introduction

Despite its development over the past half century, Radio Frequency Identification (RFID) technology has recently attracted renewed interest from many industries including retail, pharmaceutical and health care (Auto-ID Labs, 2009). Industry observers have recognized RFID as a very important technology. RFID is a non-contact automatic identification technology that identifies specific targets and retrieves information of interest using radio signals (EPCglobal Inc., 2009). An RFID system comprises three major parts, namely a tag, a reader and a unique code called an electronic product code (EPC). The RFID tag is associated with the object to be identified. The RFID reader extracts the unique identifier of the object from the tag. And the unique code EPC is provided by the electronic product code global (EPCglobalTM).

Although small, RFID systems have significantly changed in information tracking applications, and are adopted to trace objects and assets worldwide (Weinstein, 2005; Nath et al., 2006). Information applications industries can reduce the investment in management, and can enhance high-quality services by attaching smart tags to objects. An RFID network can track and identify every unique object in a network. The physical environment may have many objects with unique codes that need to be identified. Therefore, the RFID middleware may need to process huge amounts of data, causing it to become overloaded. Hence, RFID middleware applications need to consider the load-balancing problem. This study proposes a load-balancing mechanism based on grid networking for RFID middleware applications.

This investigation described EPC network and RFID middleware architecture, and presented relevant research on methods for load balancing using connection pool in the RFID middleware, the dynamic load-balancing approach on RFID middleware and grid-based job scheduling for RFID middleware applications. Specifically, this study presented a load-balancing mechanism designed to solve the middleware overload. The proposed mechanism incorporates the functional modules, buffer management and load balancing management, over a grid networking platform. The buffer management component is the first to connect with the reader in the middleware host. Meanwhile, load balancing management is the core component of the proposed load-balancing mechanism.

The rest of this work is organized as follows: Section 2 describes the related works on RFID architecture and load-balancing issues. Section 3 draws up a grid networking that could combine with the RFID middleware applications to solve the load-balancing problem. Section 4 presents the performance analysis of the proposed mechanism and compared with the other existing architectures described in Section 2. Conclusions and future works are finally drawn in Section 5.