Pushing Business Data Processing Towards the Periphery

Pablo Guerrero*, Kai Sachs, Mariano Cilia, Christof Bornhövd**, Alejandro Buchmann
Databases and Distributed Systems Group, TU Darmstadt, Germany
{sachs, cilia, buchmann}@dvs1.informatik.tu-darmstadt.de
*GK Enabling Technologies for Electronic Commerce
pguerrer@gkec.informatik.tu-darmstadt.de
**SAP Research Center Palo Alto, SAP Labs, LLC, Palo Alto, USA
Christof.Bornhoevd@sap.com

Abstract

The usage of RFID and sensing technologies in Supply Chain Management applications requires the automatic conversion of large amounts of raw data into manageable business process information. This has led to many performance and scalability issues in existing RFID infrastructures. We present an approach to alleviate these shortcomings based on a flexible system architecture that partially migrates business data processing towards the periphery.

1. Introduction

In the recent past, Radio Frequency IDentification (RFID) and a wide range of wireless sensing technologies have received much attention, as they can be used to provide companies with accurate information about their business operations in a more timely fashion. In the Supply Chain Management context, *smart items* are objects like containers, pallets, cases or even individual end-consumer items tagged with RFID chips or further equipped with sensors [4]. This allows objects to be identified, e.g., with an Electronic Product Code (EPC), or to communicate data about themselves or their environment. These technologies can lead to cost reduction and additional business benefits like increased asset visibility, improved responsiveness and even extended business opportunities.

RFID middleware infrastructures have been proposed to integrate the automatic data acquisition with existing business processes, converting large amounts of raw data into manageable business process information. Such systems are typically composed of four layers [1]. The *Device Layer* transforms the data obtained by interacting with the smart items in the physical world, into observation data usable in the digital world and vice versa. The *Device Operation Layer* coordinates multiple devices and allows to filter, condense, aggregate, and adjust the data received from the Device Layer. This preprocessed data is associated with business processes at the *Business Process Bridging Layer*. The observation data can be enriched with business data coming from the *Enterprise Application Layer*. The latter represents backend (e.g., ERP) systems, which consume the results of the business bridging processes (executed at the Business Process Bridging Layer). Various RFID infrastructures have been built, such as SAP's Auto-ID Infrastructure (AII) [1], Oracle's sensor-based services [3] or IBM's Websphere RFID Premises, among others. Although in this paper we focus on SAP's infrastructure, the problem of coping with large amounts of data is common to all of them.

2. Proposed Approach

The existing AII follows a centralized approach, i.e., business logic expressed in the form of business rules is executed in a monolithic component called Auto-ID Node (AIN). With increasing load, such a centralized system faces performance and scalability issues. Therefore, we propose a more distributed and flexible architecture aligned with the principles of the Smart Items Infrastructure [4]. This involves moving parts of the business logic closer to the point of observation, reacting locally to incoming data. We take advantage of the processing capabilities that peripheral layers offer, expecting to achieve better system performance and scalability.

In particular, at the Device Operation Layer we make use of *smart gateways*, devices with processing capabilities like those of a PDA. These devices are the gateway between the smart items placed on the Device Layer (dealing with the physical world) and the Business Process Bridging Layer (handling objects in the digital world). These smart gateways are responsible for business data processing tasks that go beyond the typical filtering and aggregation activities, for instance, by triggering an alarm when an item's cold chain is interrupted.

To achieve a flexible and configurable system architecture, we define business rules in the form of Event-Condition-Action rules [2]. These rules describe the reaction of the system to incoming data (e.g., RFID observations and sensor readings) in a simple and platform-independent way. They allow an easy deployment on rule engines located at different layers, making it possible to migrate business data processing across the layers. Based on the content of the deployed rules, different actions can be triggered by the rule engine to process incoming data.

Smart gateways, on the other hand, have limited memory and computation power. The processing of complex business rules in the form they are currently expressed and executed on the AIN is not always possible on those devices. These limitations require a lightweight, small footprint rule engine. To accomplish this, we have implemented a component-based rule engine, consisting of elementary building blocks (known as elementary services), e.g. basic event detection, event composition, condition evaluation, and action execution. Additionally, the design is based on a service platform that controls the life-cycle of the services and allows their remote deployment and management.

By implementing the rule engine using this approach, we are able to define a minimalist rule processor that is specialized for processing the rules needed on the Device Operation Layer. Therefore, these rules will have a limited set of actions to be invoked in response to detected events. More complex activities, on the other hand, will be delegated to (or remain at) the Business Process Bridging Layer. The advantage of decomposing some of the complex business rules into simpler rules is that they can be evaluated on the Device Operation Layer. In this way, some of the business logic from the AIN is offloaded to the smart gateways, thus distributing the processing of business data, exploiting the processing capabilities available at the periphery.

3. Overview of the Demonstration

The supply chain scenario we selected for this demonstration deals with shipment handling. A supplier ships goods stacked on pallets to a retail distribution center. Each pallet has an RFID tag attached and a wireless sensor node. As the pallet leaves the supplier's loading dock, the tag is automatically scanned and the sensor node is instructed to start recording temperature and light readings. When the truck loading process is completed, the supplier's internal system sends an Advance Shipment Notice (ASN) to the retailer.

The demonstration begins by showing the use cases that cover different shipping situations ranging from correct arrival of expected goods, through missing pallets, sensor readings infringing agreed physical conditions, and finally sensor nodes not responding (due to dead batteries or comm. errors). Particularly, in this ASN scenario, four business rules suffice and are presented. The deployment of these rules (i.e., registration and activation) is shown. Moreover, exceptions, like unexpected items, are also thrown and shown with the appropriate context information.

3.1. Implementation Details

In our implementation we fetch ASNs on-demand from the repository to reduce the traffic between the AIN and the smart gateways and memory space, maintaining only ASNs in memory that are currently being processed. This also enables the distribution of the ASN processing by having multiple smart gateways processing individual cases of incoming goods while the AIN maintains a central/consolidated repository and is responsible for complex and exceptional situations. We also incorporate the possibility to consider environmental data (i.e., sensor readings) during the ASN verification process. This means that the business rules on the smart gateways can interrogate the sensors attached to the delivered goods. Moreover, these sensor nodes verify ASN's physical conditions by reading their history data, further offloading business data processing.

4. Conclusions

The obvious advantage of decomposing complex business rules into simpler rules that can be executed at the smart gateways is the ability to offload some of the business logic from the Business Process Bridging Layer to the Device Operation Layer. The approach presented here can be used to implement a more flexible and scalable Auto-ID solution which allows to execute business logic directly at the location of the observation of RFID and sensor reading events. A detailed performance analysis is currently being performed.

References

[1] C. Bornhövd, T. Lin, S. Haller, and J. Schaper. Integrating Automatic Data Acquisition with Business Processes - Experiences with SAP's Auto-ID Infrastructure. In *Procs of VLDB'04*, Aug. 2004.

[2] M. Cilia. An Active Functionality Service for Open Distributed Heterogeneous Environments. Ph.D. Thesis, Technische Universität Darmstadt, Germany, Aug. 2002.

[3] E. Kalischnig. RFID: Making Sense of Sensor-based Technology, July 2005. Oracle WP.

[4] P. Spiess, C. Bornhövd, T. Lin, S. Haller, and J. Schaper. Going Beyond Auto-ID: A Service-oriented Smart Items Infrastructure. *JEIM Journal*, Mar. 2007. (to appear).