A Case Study in Business Application Development Using Open Source and Semantic Web Technologies

Ivana Novicic[1], Zoran Kokovic[1], Niksa Jakovljevic[1], Vuk Ljubicic[1], Marko Bacetic[1], Nenad Anicic[1], Zoran Marjanovic[1], Nenad Ivezic[2]

[1] Faculty of Organizational Sciences, Jove Ilica 154, Belgrade, Serbia
{novicic.ivana|kokovic.zoran|jakovljevic.niksa|ljubicic.vuk|bacetic.marko|anicic.nenad|marjanovic.zoran@fon.bg.ac.yu}

[2] National Institute of Standards and Technology, 100 Bureau Drive, Gaithersburg, MD, 20899, USA
nivezic@nist.gov

Keywords: business application development, Semantic Web, open source, interoperability

1 Introduction

Apollo is an open source, experimental Inventory Visibility (IV) application. The IV applications enable both suppliers and customers to have a Web view and to manage activities that affect the customer inventory based on a certain inventory replenishment process. The type of the replenishment process that is supported by the Apollo application is electronic Kanban (eKanban).

The motivation to explore new development approaches follows from the need to address high costs in cross-enterprise business application integration efforts, particularly in complex industrial supply chains [2].

The primary direction of exploration in developing the Apollo IV application was to assess the new Semantic Web-based technology for data messaging interface in the context of the newly proposed semantic mediation architecture within the ATHENA project [1]. In addition, we are interested in how the Open Source technology can support our development.

2 Requirements

The primary requirements for the Apollo IV application follow from the Inventory Visibility and Interoperability (IV&I) eKanban Business Process Model specification defined by the Automotive Industrion Action Group (AIAG) [3].

The eKanban protocol specifies four messages that are exchanged between the customer and supplier to indicate and exchange data about the corresponding four important inventory replenishment events. A Business Object Document (BOD) specification corresponds to each of the eKanban messages and defines data types and structure of the document-based message exchange [9]. The BOD specifications were built by adapting and further building on the Open Applications Group (OAG) XML standard specifications [12].

The final requirement comes from the ATHENA research project in which Apollo is intended to be used as an experimental application. The Apollo IV application was required to utilize both Semantic Mediation and Web Services execution support of the relevant ATHENA tools and eKanban-specific artefacts (e.g., IV&I eKanban Ontology) to become IV&I conformant.

3 Development Method and Architecture

The following were the five major steps in developing the Apollo application

- Identify the IV Functionality
- Define IV&I eKanban Protocol-to-process messages
- Define IV&I eKanban BOD Processing Functionality
- Determine message exchange framework
- Perform semantic reconciliation of messages

Fig. 1 shows the adopted Apollo architecture, as a consequence of the first two steps. The following are the main components: IV&I Web Client, Business Component, Web Service Interface, and XML/RDF Adapter.

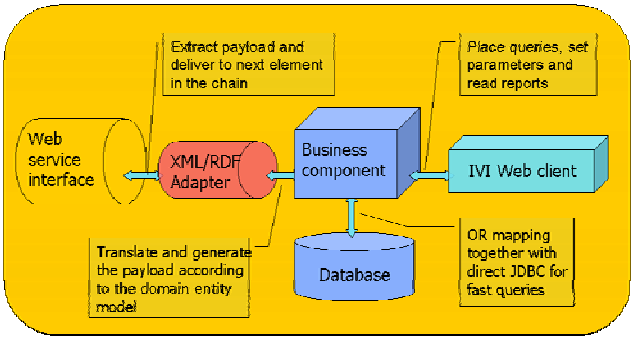

Fig.2. The Apollo IV Application Architecture

4 Semantic Web-based Application Interface

4.1 ATHENA Semantic Mediation Approach

The ATHENA semantic mediation capability uses an ontology-based integration approach to address the issues of semantic interoperability. Using the ATHENA approach, the reconciliation rules between each application logical interface schema and the IV&I eKanban Reference Ontology (RO) may be defined to map the logic of that IV application data interface using the RO elements. Once these reconciliation rules are available, IV&I eKanban messages may be transformed from one form to another and interoperable data exchange may be achieved using a semantic mediator to run the reconciliation rules.

4.2 RDFS-based Data Exchange Interface

We defined data exchange interface using Resource Description Framework Schema (RDFS) [10]. RDF Schema provides modeling primitives for organizing Web objects into hierarchies[13].

RDFS was selected for several reasons. The first reason is its simplicity because RDFS does not contain any structure at all [15]. Classes and properties are connected to each other by simple referencing statements. The second reason was the RDFS' ability to support inheritance between classes [11].

Developing the RDF interface consisted of the following activities:

- Defining the Kanban model for understanding received messages.
- Testing the information contained in the message payload.
- Enabling translation of domain entities into RDF instances.

5 Open Source Implementation

Development of the IV&I tool started with the selection of existing open source software (OSS) products, that represent a good starting point for the development process. Using existing OSS products gave us the ability to modify an existing solution instead of creating a new one from scratch.

For the business model and process solutions we used Glassfish Enterpise Java Beans [7] (EJB) 3.0. The majority of the GlassFish code is available under the Common Development and Distribution License [14] (CDDL) v1.0. Java Server Faces [4] (JSF) 1.2 was our choice for the presentation tier. Sun's implementation of JSF has been released under the OSI-approved CDDL. SOAP [5] with Attachments API for Java (SAAJ [6] Standard Implementation) was used as a Web Service technology. A standard implementation has been released under the OSI-approved CDDL. For developing the RDF interface, we used Sesame, Elmo, and related OSS technologies [8].

6 Conclusions

This paper presented an overview of one applied approach, based on an open source development approach and Semantic Web technologies, to achieve interoperability between business applications. The approach was tested in practice and it showed in experimental conditions that it meets our expectations: Apollo successfully exchanged an eKanban message instance with another independently developed application using the ATHENA Semantic Mediation and Web Services Execution approach. Hence, the Apollo application provides an evidence that the novel approach may be put into practice using today's open source solutions.

Disclaimer

Certain commercial software products are identified in this paper. These products were used only for demonstration purposes. This use does not imply approval or endorsement by NIST, nor does it imply these products are necessarily the best available for the purpose.

References

[1] ATHENA, Advanced Technologies for interoperability of Heterogeneous Enterprise Networks and their Applications Project, (2006), http://www.athena-ip.org.
[2] QAD inc. Establishing standards for automotive IV&I, 2004, available at www.qad.com.
[3] AIAG, Automotive Industry Action Group, IBP-2 Inventory Visibility & Interoperability Electronic Kanban Business Process,Version 1, Mar. 2006, Available at www.aiag.org.
[4] JavaServer Faces Technology 1.2 Reference Documentation, Available at: http://java.sun.com/javaee/javaserverfaces/reference/api/index.html
[5] SOAP Version 1.2 Part 1: Messaging Framework, Available at: http://www.w3.org/TR/2003/REC-soap12-part0-20030624/
[6] Java Web Services - SOAP with Attachments API for Java (SAAJ), Available at: http://java.sun.com/webservices/saaj/index.jsp
[7] Java Platform, Enterprise Edition (Java EE), Enterprise JavaBeans Technology, Available at: http://java.sun.com/products/ejb/
[8] Documentation for Sesame, Rio and Elmo, Available at: http://www.openrdf.org/
[9] Open Applications Group (OAG), Available at: http://www.openapplications.org/
[10] Resource Description Framework (RDF) Schema Specification 1.0 W3C 2000, Available at: http://www.w3.org/TR/2000/CR-rdf-schema-20000327
[11] RDF Primer, W3C Recommendation, Available at: http://www.w3.org/TR/rdf-primer/
[12] XML 1.0 3.ed *DTD*, World Wide Web Consortium Recommendation, Available at: http://www.w3.org/TR/REC-xml/
[13] Grigoris Antoniou, Frank van Harmelen., A Semantic Web Primer. The MIT Press, 2004.
[14] Common Development and Distribution License (CDDL) , Available at: http://www.sun.com/cddl
[15] Shelley Powers. Practical RDF. O'Reilly, 2003.