A Customer Notification Agent for Financial Overdrawn Using Semantic Web Services

José Manuel López-Cobo, Silvestre Losada, Oscar Corcho, Richard Benjamins, and Marcos Niño

Intelligent Software Components, S.A. (iSOCO),
C/ Francisca Delgado, 11 – 2. 28108 Alcobendas,
Madrid, Spain
{ozelin,slosada,ocorcho,rbenjamins,marcosn}@isoco.com

Abstract. In this paper, we present a Notification Agent designed and implemented using Semantic Web Services. The Notification Agent manages alerts when critical financial situations arise discovering and selecting multichannel notification services. This agent applies open research results on the Semantic Web Services technologies including on-the-fly composition based on a finite state machine and automatic discovery of semantic services. Financial Domain ontologies, based on IFX financial standard, have been constructed and extended for building agent systems using OWL and OWL-S standard (as well as other approaches like DL or f-Logic). This agent is going to be offered through integrated Online Aggregation systems in commercial financial organizations.

Keywords: Semantic Web Services, Ontologies, Composition, Intelligent Agent

1 Introduction

The objective of the distributed system described in this paper (the Customer Notification Agent) is to provide added value to customers of financial services. This added value consists in a fully customizable and configurable set of aggregations and estimation functionalities on account balance evolution, as well as SMS and email alerts (among others), which will allow customers to have more efficient information about his financial position.

This system reuses existing technology for aggregation available at our company (iSOCO GETsee®), and migrates it to Semantic Web Services technology. The integrated use of Semantic Web technologies and Web Services allows us to describe and reason with pieces of code understandable for machines, discharging the sometimes tedious task of checking the online accounts to a software system. This system is able to engage with other commercial solutions for aggregation and to detect at run-time and raise alerts if some conditions are detected (for example, a possible overdrawn of a customer saving account, due to the future payment of an invoice).
We have developed different ontologies to express the needed knowledge for this application. These ontologies are divided into three groups: general ontologies, which represent common sense knowledge reusable across domains; domain ontologies, which represent reusable knowledge in a specific domain; and application-dependent ontologies, which represent the application-dependent knowledge needed.

We have defined three high-level services for performing the task of the Customer Notification Agent. The **GETseeSWS Service** accesses the online accounts of the customer and the invoices associated with them, and calculates the balance for these accounts. The **NotificationService** notifies customers with different types of messages (discharging in 3rd party providers the execution of the actual notification) and finally, the **EstimationService** estimates, using different kinds of arithmetical functions, the expectable amount of an invoice for preventing an overdrawn situation.

One of the main innovations of our systems is the proposal of a finite state diagram to represent the composition of atomic processes into composite ones using conditions as a way to choose between different choices. Such an approach allows at run-time the discovery and invocation of services which comply with the conditions defined for the transition from one state to another. This allows describing a composite process at design-time by defining its behavior and leaving the selection of the specific service to the execution time. This is an innovation with respect to other approaches where the selection of the specific services is done also during the design time.

The paper is organized as follows. Section 2 describes a sample scenario where the Notification Agent can be used, showing the main actors and agents involved in the overall process and the steps usually followed by them. Section 3 describes the ontologies that we have developed, either from scratch or by reusing other ontologies or vocabularies already available elsewhere. Section 4 describes the Semantic Web services created for the system, which have been implemented using OWL-S, DL and f-Logic. Section 5 describes one of the main contributions of this paper, namely the proposal for service composition using finite state diagrams. Finally, section 6 provides some conclusions of our work and future lines of research.

### 2 Scenario Description

Let us suppose that we are working on the scenario presented in figure 1. In this scenario we have a customer with several banking accounts where he/she has different amounts of money. This customer has also contracts with some consumer goods companies such as a telephone company, and gas and electricity providers, among others.

Everyday, the Customer Notification Agent will detect whether any of the customer accounts is going to be in an overdrawn situation. Bank accounts may have different invoices associated from different consumer good companies. If the amount of the invoice is bigger than the amount of money of the account, there could be an overdrawn situation. To help the customer, the system calculates an estimation of the amount of every invoice expected for that account before its value date and notifies the customer if the balance of the saving account is less than the expected invoice amount. The system will choose any of the notification channels available for the customer and will notify him/her about the overdraw possibility.