AgFlow: Agent-based Cross-Enterprise Workflow Management System

Liangzhao Zeng, Boualem Benatallah, Phuong Nguyen
University of New South Wales, Australia
{zlzhao,boualem,ntp}@cse.unsw.edu.au

Anne H.H. Ngu
Telcordia Technologies, U.S.A
angu@research.telcordia.com

1 Introduction

The ability to efficiently and effectively share business processes on the Web is a critical step towards the development of the new on-line economy driven by the B2B E-commerce. Existing enterprises would form alliances and integrate their processes to share costs, skills and resources in offering *Virtual Enterprises* [Geo99].

With respect to meeting the requirements of virtual enterprises, we identified two relevant emerging technologies: *workflows* and *agent* technologies. Despite the success of using workflow technology in automating internal business processes, there has been little success to achieve integration of cross-enterprise workflows. The support of flexible and efficient integration of workflows requires the ability to efficiently discover and exploit business services in a dynamic and constantly growing environment. It also requires the capacity to dynamically establish relationships among business processes.

Existing agent-based systems [HS97] do not have the concept of a business process model. At best, a business process can be worked out and agreed to offline. After that, the different activities of the business process are implemented and embedded in different agents that collaborate with their peers in order to achieve the global business process. These solutions are mostly appropriate if the business processes to be integrated belong to organizations with *long-term* and *static* trading relationships. However, B2B E-commerce requires more flexible integration techniques. More importantly, the fast and dynamic integration of business processes is an essential requirement for organizations to adapt their business practices to the dynamic nature of the Web.

In order to address the problems of sharing workflow processes over the Web, the *AgFlow* prototype has been developed. The philosophy behind our agent-based approach is that current workflow technology is unable to cope with dynamic interactions which is what agent-based systems can provide. In the following sections, we briefly overview the design and implementation of the AgFlow system, and sketch the proposed demo.

Proceedings of the 27th VLDB Conference, Roma, Italy, 2001

2 Design Overview

Our approach combines agents with workflows to effectively *integrate* cross-enterprise workflows. In AgFlow, a workflow schema is defined as a statechart diagram. A cross-enterprise workflow instance is a collection of tasks combined in certain ways by a process agent. The tasks can be assigned to different service providers and executed in a distributed environment. In our approach, among other things, agents are used to wrap services which are able to execute workflow tasks. Based on the requirements of tasks, our system searches for agents with matching capabilities.

Four types of agents are used to provide the core functionalities of the cross-enterprise workflow engine, namely *user agent*, *process agent*, *discovery agent* and *service agents*. Users can define workflow schemas, create workflow instances, control and monitor execution instances. In order to manipulate an actual service and abstract it from its physical organization, we wrap it by a *service agent*. Typical examples of services include procurement, customer relationship management (CRM), finance, billing, accounting, supply chain and manufacturing. The discovery agent allows each service agent to advertise its location and properties in a meta-data repository. It is also used to locate service agents based on constraints over their meta-data.

In AgFlow, a workflow agent community is defined as a set of service agents which are involved in executing a workflow instance. We use the *process agent* to dynamically construct the workflow agent community. For each task in a workflow instance, the process agent queries the discovery agent to locate the service agents that can provide the task. In the case where several service agents are found, the process agent uses the following formula to choose one service agent:

$$Q = W_c * \left(\frac{C_{actual}}{C_{planned}}\right) + W_t * \left(\frac{T_{actual}}{T_{planned}}\right)$$

where C_{actual} (respectively, T_{actual}) represents the provider estimated fee (respectively, response time). $C_{planned}$ (respectively, $T_{planned}$) represents the user expected fee (respectively, execution time). W_c, $W_t \in$

$[0,1]$, are the weights of service fee and response time factors. The workflow definition provides values of $C_{planned}$, $T_{planned}$, W_c, W_t for each task. The values of the C_{actual}, T_{actual} are provided by service agents. The process agent assigns a task to a service agent which has the minimal value of Q. Note that user defined selection strategies are also supported. We use statechart diagrams to specify these strategies.

3 Implementation

The AgFlow architecture (see Figure 1) is deployed using *Enterprise JavaBeans* (EJB). Agents communicate through a communication bus which is implemented using Java Shared Data Toolkit (JSDT). The user agent has two components: *Workflow Specification Editor* and *Workflow Execution Monitor*. The Workflow Specification Editor allows workflow designer to define workflow schemas and end users to create and execute workflow instances. The Workflow Execution Monitor allows users to monitor and control the execution of workflow instances. Currently, three types of services are considered in AgFlow: Domino-based workflows, EJB applications, and Java applications that allow access to relational databases via JDBC. We are currently working on wrapping other types of services including Oracle Workflow, BEA Weblogic Process Integrator and SAP/R3.

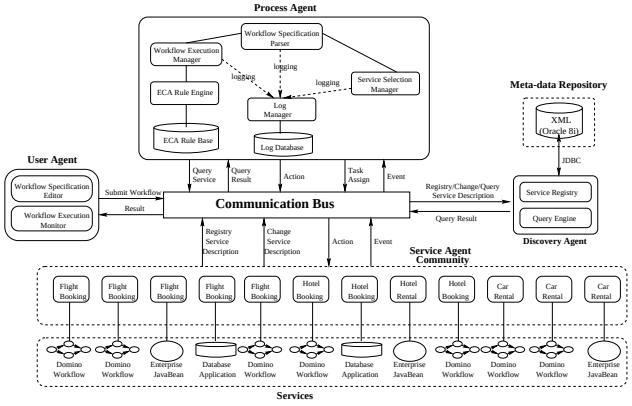

Figure 1: AgFlow Architecture

The discovery agent consists of two components: *Service Registry* and *Query Engine*. The Service Registry is used to advertise agent services in an XML-based meta-data repository. We adopt the Oracle8i database server as meta-data repository. The Query Engine takes a service query message from the process agent as input and generates service agent descriptions (XML document) as a result.

The process agent consists of several components implemented using EJB: *Specification Parser*, *Service Selection Manager*, *Workflow Execution Manager*, *ECA Rule Engine* and *Log Manager*. The Specification Parser parses the workflow specification into a set of tasks and a set of ECA rules. For each task, an instance of the Service Selection Manager is created. This instance queries the discovery agent to locate service agents that can be used to execute the task. The Workflow Execution Manager coordinates the execution of the workflow by sending action messages to service agents.

4 Demo Description

We consider a business trip planning application to illustrate the use of AgFlow. The prototype that we developed uses 12 service agents (flight ticket booking services, hotel room booking services and car rental services).

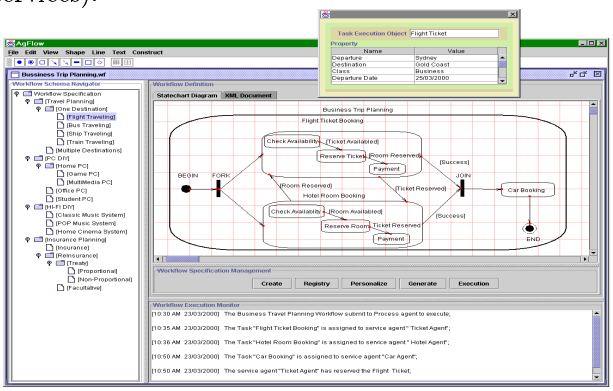

Figure 2: User Agent Interface

A user can browse workflow schemas using the Workflow Schema Navigator panel (see Figure 2). When she/he clicks on a leaf node, the corresponding statechart diagram is displayed in Workflow Definition panel. An instance of the workflow is created by clicking on the `Instance` button in the Workflow Management panel. The user can then supply values of the parameters that are needed to execute the workflow (e.g., the destination and departure date). After that, the user can click on the `Generate` button to generate the XML document that describes the tasks and rules of the workflow. After clicking on the `Execution` button, the user agent creates a workflow instance for the selected workflow schema. The XML document is sent to the process agent. The process agent queries the discovery agent to locate the relevant service agents. It also coordinates the execution of the tasks of the `Business Trip` workflow.

References

[Geo99] D. Georgakopoulos, editor. *Information Technology for Virtual Enterprises*, Proc. of the 9th Int. Workshop on Research Issues on Data Engineering. IEEE Computer Society, March 1999.

[HS97] M. Huhns and M. Singh, editors. *Internet-Based Agents*, IEEE Internet Computing. IEEE, July 1997. Special Issue on Agents.