Automating Business Processes with Software Agents

David L. Bailey
Express
1 Limited Parkway
Columbus, Ohio 43230
(614) 479-4479

Premise

Agents provide the artificial intelligence community with a significant opportunity to participate in broad business application architectures, rather than focusing on niches. While agents enhance the usability of commercial software, such as email and personal information systems, their greater potential may lie in automating business processes and mission critical applications.

Recent interest in distributed computing, business process modeling and re-engineering, and enterprise architecture planning are catalysts for revitalizing the existing business information infrastructure. As organizations look for new ways to increase productivity and capture competitive advantage, agent technology and other advanced computing techniques become increasingly important and attractive.

The Promise of Agents

Agent-based approaches differ from traditional business applications in many ways. Traditional applications typically focus on data storage and manipulation, without much support for decision making. In addition, these applications tend to be rigid, procedural, and have character-based interfaces. The user is often required to work around outdated application logic due to development backlog and complexity in maintenance.

Agents, on the other hand, promise flexibility, ease of use, and a full range of services. They can be more autonomous and intelligent, independently exploring the environment for new information and discovering important trends. Presumably, an agent would have access to other agents, applications, and data sources. They can cooperate with other agents or spawn subtasks to perform tasks. Agents would collaborate with users, and adapt to their needs and to the environment. Agents should have the potential to be non-intrusive and to not interfere with other user processes.

An Agent-Based Information Architecture

One approach to an agent-based information architecture is to use agents to model business processes, important business concepts or data entities, system services, and users. An agent in each of these areas is responsible for providing current information and analysis, suggesting courses of action, critiquing user work, diagnosing problems, monitoring workflow, and tutoring novices. These roles require the agent to have

knowledge of the business, current status information, historical information, and context from other agents.

A business process agent supports the users, applications, and information required to perform a business process such as product development, order management, and inventory management and logistics. In order to perform these roles, the business process agent would require business models and information, including a:

- Glossary of salient terms
- Models of business concepts (static / structural, dynamic / behavioral, state)
- Model of the business process (tasks, resource requirements, work activities and products, a dynamic simulation)
- Knowledge of information resources.

For example, an agent responsible for product development would have knowledge of the process (trend identification, creating prototypes, testing market response, and so on), a glossary of product, manufacturing, and testing terms, knowledge of the organization's policies and procedures, and a map of information resources.

Agents for business concepts provide a level of abstraction above data entities, and are responsible for providing salient information to business process agents and other business concept agents. In addition to extracting concept information on request, these agents perform knowledge discovery tasks, and monitor data quality and availability. More abstract business concept agents interact with data entity level agents to monitor and explore increasingly complex information.

User agents are responsible for modeling user preferences and priorities, and communicating with both the user and other agents. User agents perform several tasks, including:

- Monitoring user activity, performing user modeling and plan recognition functions
- Accepting and translating requests for information from the user
- Communicating priority, resource allowance, and preference information to other agents
- Ensuring that results are properly formatted and presented to the user.

System agents include control the use of system resources, provide special modeling and formatting services, and answer questions about the environment. There are several classifications of systems agent, the most common modeling the operating system functionality.

Agent Behavior

Agents in this architecture should conform to several social and environmental laws. Social laws ensure cooperation and efficiency, while environmental laws prevent interference with the user. Proposed social laws include:

- Take turns using resources
- Share information with all that ask
- Perform higher priority tasks before those with lower priority
- Do own work before helping others.

Environmental laws govern resource use, security, and user "avoidance". Proposed environmental laws include:

- Conserve resources, especially on low priority tasks
- Clean up after finishing (delete, unlock, restore)
- Stay out of the user's way (if user is heavily accessing disk, avoid disk-intensive processing)
- Abide by the user's security (data and application) when performing tasks.

Agent Composition

To perform the range of tasks described above, individual agents require several software components. Several components are common across agents, though the emphasis on each component will vary by the type of agent. Components include:

- Domain model (business process, business concept, user)
- Environment model (resource availability, information access)
- Self model (performance, activities, requirements)
- Motivator (actions are based on motivations, requests, and environmental events)
- Planner/Estimator (task activities, probability of success, time to achieve, resource usage)
- Communicator (inter-agent, data, user, application)
- Analyst (current data, trends, events)
- Knowledge discoverer (data, events, own behavior)
- Inferencer/Diagnostician (business concept problem, user questions)
- Other effectors (peripherals, applications, data).

Agents communicate via a logic-based command language within packets of information. Communication can be centrally coordinated, occur through blackboards, or through direct communications based on an agent-specific model of the environment.

Adaptation within agents can occur within the business, user, environment, and self models. Methods of adaptation should include both inductive and explanation-based techniques.

Issues for Business Information Agents

Business information agents have incredible potential, but will require answers to several questions. Some of the research questions to answer include:

- How can agents be believable, without causing the user to rely too heavily on them?
- Can agents estimate the accuracy of their responses?
- Can agents effectively model their environment and provide input without being obtrusive?
- How can agents be developed and maintained effectively (tools, shells, generic business components)
- What are the tradeoffs between effectiveness, performance, and resource usage?
- Which current learning techniques are sufficient to adapt an agent to its environment?
- Can agents be sufficiently controlled and protected?

Current Status

We are pursuing an information architecture project that includes identifying business processes and a number of the key concepts supporting business information agents. We are investigating a number of advanced computing techniques, including software agents. My current work with agents includes information gathering, feasibility and applicability determination, and conceptual design.