Reusable Components for Middleware Agents

Alun Preece

University of Aberdeen, Computing Science Department
Aberdeen AB9 2UE, Scotland
Phone: +44 1224 272296; FAX: +44 1224 273422
Email: apreece@csd.abdn.ac.uk

Overview of Current Projects

There is currently considerable interest in using agent-based middleware technology to integrate sources of data and knowledge. Some of these sources are legacy systems (pre-existing databases and expert systems), while others are custom-built services specifically designed to operate as components within distributed information architectures. Some of the common scenarios served by such architectures are:

- extracting data from databases and providing it as the input to knowledge-based systems, which in turn derive new information;
- extracting data from databases and knowledge from knowledge bases, and combining both to compose a new information source;
- extracting and transforming data and knowledge into constraint programs, the solution of which yields new information.

Some example instantiations of these scenarios in actual application domains are:

- distributed engineering design, where data on components is combined with knowledge of how designs are composed, and constraints given by the customer's requirements (for example, configuring a modular computer system);
- medical informatics systems, where patient data is fed into expert systems for therapy recommendation or critiquing (for example, recommending appropriate drugs based on the patient's needs and current drug usage);
- University admissions systems, where appropriate programmes of study can be offered, based on the student's needs and consistent with data on their academic history (for example, knowledge of prerequisites can be checked against the student's database entry from their previous institution).

At Aberdeen, we have a number of current projects developing technology to support such scenarios. We favour an agent-based architecture for these projects, in which there are three types of agent, but any agent is potentially able to communicate with any other:

User Agents These are components that interact directly with human users; typically they offer graphical user interfaces; sometimes they are embedded in Web pages in the form of Java applets.

Resource Agents These are resources — databases or knowledge-based systems — which have been "agentified" by providing them with an agent front-end.

Middleware Agents These are the agents that actually provide the database and knowledge base integration services; typically, each performs some useful task, on request from some other agent or on its own "initiative", and calling upon the services of other agents.

Our agents communicate using an agent communication language (generally a variant of KQML), implemented on top of a distributed object messaging mechanism (currently either CORBA IIOP or Java RMI). Various directory services allow agents to discover one another dynamically at run-time; these range from the simple directory services offered by CORBA and Java RMI, to richer agent-oriented advertisement and brokerage services.

A selection of our current projects are briefly described below.

KRAFT: Knowledge Reuse and Fusion/Transformation KRAFT is a three year, multiple-site research project involving the Universities of Aberdeen, Cardiff and Liverpool, and British Telecom. Funded by the EPSRC and BT, KRAFT aims to create a generic architecture for sharing knowledge in the form of constraints. Constraint knowledge is extracted on demand from databases and knowledge-bases, transformed to a shared ontology, and delivered to an appropriate solver. Middleware agents include *mediators*, which locate and transform the constraints, and *facilitators*, which provide sophisticated directory and brokerage services. KRAFT is specifically intended to support distributed engineering design applications, and a prototype application is currently under construction. An overview of the project has been published in (Gray *et al.* 1997).

Knowledge Reuse and Validation Following the author's earlier work in the verification and validation of standalone knowledge-based systems (Preece, Shinghal, & Batarekh 1992), tools are under development to assist in the design and validation of distributed knowledge-based systems. DISCOVER and COVERAGE are tools that support the construction of multiple-agent systems. DISCOVER verifies that "agentified" knowledge bases and databases conform to a shared ontology; this is a pre-requisite for sharing knowledge in an integrated system. DISCOVER is described in (Waterson & Preece 1997). COVERAGE verifies the well-formedness of "teams" of agents: it establishes the closure of task interdependencies, by checking that agents are able to meet their advertised commitments. COVERAGE is described in (Lamb & Preece 1996).

Reusable Middleware Components In parallel with the above projects, we have been attempting to create a library of reusable middleware components to ease the construction of knowledge base and database integration applications. We currently have sets of Java classes for:

- agent communication using a subset of KQML, running over Java RMI or CORBA IIOP;
- "agentifying" SQL databases (using JDBC) and knowledge bases implemented in CLIPS and Prolog;
- monitoring and visualising the interaction of agents, for debugging and demonstration purposes.

As we gain more understanding of building these kinds of applications, we expect to identify design patterns which will make it easier to reuse the above classes.

Acknowledgements The author would like to acknowledge current and recent collaborators on the above projects at Aberdeen: Alastair Borrowman, Suzanne Embury, Jutta Eusterbrock, Ted Francis, Peter Gray, Kit Hui, Graham Kemp, Neil Lamb, Brian Rogers, and Andrew Waterson.

References

Gray, P.; Preece, A.; Fiddian, N.; Gray, W.; Bench-Capon, T.; Shave, M.; Azarmi, N.; and Wiegand, M. 1997. KRAFT: Knowledge fusion from distributed databases and know ledge bases. In Wagner, R., ed., *Eighth International Workshop on Database and Expert System Applications (DEXA-97)*, 682–691. IEEE Press.

Lamb, N., and Preece, A. 1996. Verification of multi-agent knowledge-based systems. In *ECAI'96 Workshop on Validation, Verification and Refinement of Knowledge-Based Systems*, 114–119. Budapest, Hungary: ECCAI/NJSZT.

Preece, A. D.; Shinghal, R.; and Batarekh, A. 1992. Principles and practice in verifying rule-based systems. *Knowledge Engineering Review* 7(2):115–141.

Waterson, A., and Preece, A. 1997. Knowledge reuse and knowledge validation. In *Verification and Validation of Knowledge-Based Systems: Papers from the 1997 AAAI Workshop (Technical Report WS-97-01)*. Menlo Park, CA: AAAI Press.