UNIVERSITY OF SOUTHAMPTON

DEPARTMENT OF ELECTRONICS AND COMPUTER SCIENCE

DISTRIBUTED INFORMATION MANAGEMENT WITH MOBILE AGENTS

Jonathan Dale and David C. DeRoure

University of Southampton

ECS Research Journal April 1997

Distributed Information Management with Mobile Agents

Jonathan Dale and David C. DeRoure

Multimedia Research Group

Department of Electronics and Computer Science

University of Southampton

UK

{jd94r, dder}@ecs.soton.ac.uk

With more users taking advantage of publicly accessible networks, such as corporate intranets and the Internet, larger amounts of information is becoming electronically distributed and disseminated. Distributed information management is an emerging technology for dealing with the problems of managing information that is spread across networks, users and applications. We present four categories that we consider being necessary to developing tools to undertake distributed information management tasks.

To help model the dynamic and heterogeneous nature of a user's distributed information, we advocate the use of agents and agent technologies when building distributed information management applications. We present an agent-oriented architecture which is based around a concept of mobile agents, since they provide a convenient abstraction for modelling distributed applications.

1. Introduction

Through the increasing use of large-scale networked environments, such as the Internet, users are becoming more aware of the advantages presented through the distribution of information. More electronic information is available now than ever before and this is set to grow as information networks achieve greater penetration into corporations and consumer households. However, any form of distribution brings with it a management overhead and in systems such as the World Wide Web (WWW), the burden of dealing with this lies firmly with the user.

Distributed Information Management (DIM) [8,4] is a term that is being used to describe the set of activities that are necessary to allow a user to manage the entire life-cycle of their distributed information. The set of individual or composite actions required to move distributed information between the various states within the life-cycle are termed *DIM tasks*. Considering the WWW as an exemplar, very few tools exist to allow users to create, publish, manage and navigate information on a global scale; most WWW tools lie either very much in the management of generally small-scale information sets or in the discovery and subsequent retrieval of information.

Mobile agents are particularly suited to developing tools to deal with DIM tasks [3], since they offer a level of abstraction that can be modular and tailored to a particular aspect of the system; they can represent tasks, users, information, information resources, services, security management and so on. Their dynamic and extensible nature can help them deal with heterogeneous networks, data formats and communication protocols, and their mobility can help them address distribution and scalability issues.

In this paper, we present an agent-oriented architecture that supports both static and mobile agents which can support or carry out DIM tasks. Section 2 provides a description of DIM and the types of DIM agents that we are developing and in section 3 we detail the mobile agent architecture that we are using to build our exemplar DIM applications.

2. Distributed Information Management

DIM is an initiative that is being promoted by the UK Technology Foresight panel on IT and Electronics to help promote research into the management of change and evolution in distributed systems, particularly in multimedia applications [8]. In summary, this programme of work deals with the management of multimedia information across distributed and federated systems and applications.

2.1. DIM Activities

We have identified four activities that we believe comprises DIM [2]:

- *Resource Discovery*. The discovery, retrieval and monitoring of information that is of relevance to the user. It is important that any resource discovery is tailored to the context of a user's query and also helps to qualitatively refine any search results to ensure that only appropriate information is presented.
- *Information Integrity*. The creation, modification and maintenance of information that exists across distributed information servers. This also includes hypermedia link management, collaborative working, versioning and service provision during intermittent network connectivity.
- *Navigation Assistance*. The process of assisting the user in navigating some information resource or set of information resources. This might involve determining the information that is of most relevance to the user and expanding on the amount of information when not enough is presented or reducing the amount when too much is offered.
- *System Integration*. The integration of information from heterogeneous networks, communication protocols and data formats. This also includes integration with the user's favourite desktop applications and the presentation of any information in an application most convenient to the user.

2.2. DIM Agents

The agents that we envisage for representing our DIM tasks are based around a fundamental belief that these agents should help users with tasks that they encounter and perform on an everyday basis. We believe that the intelligence attributed to an agent should be directly related to the tasks that the agent is performing and how it should be *expected* to perform them. This theory is supported by Brooks [1] when he states that '...intelligence is in the eye of the beholder' and that the intelligence of an agent arises from its interaction with its environment.

Therefore, our agents conform to the weak agent classification identified by Wooldridge and Jennings [9]. That is, our agents posses the ability to operate independently from their user (*autonomy*), to communicate with the outside world (*social ability*), to perceive and react to changes in their environment (*reactivity*) and to be adaptive to new situations and to take the initiative when necessary (*proactivity*).

Jennings [6] uses the term *ubiquitous computing* to describe the shift from the computer and its software being just a user-directed processor to more of a delegating partnership with the user. In this way, we see DIM agents expressing and performing DIM tasks on behalf of the user to help them manage their distributed information resources; both as directed by the user, but also in the background and automatically as changes occur in the environment that either impinge upon the user's information or are of relevance to the user.

3. An Architecture for Mobile DIM Agents

The architecture that we have developed [2,3] is written in the Agent Process Interaction Language (APRIL) [7] and is serving as an environment in which we can test our DIM agents. Every object within the architecture is either an agent or is abstracted through an agent; this provides for a flexible and dynamic model, since new components can be added to the system by adding a new agent type.

Figure 1 shows the interaction of the core agents within the architecture:

- *Domain Agent*. A static agent that controls access to and from a *domain* (a logical collection of machines, information resources, agents and users), as well as initially mediating access to agents within the domain. The domain agent is also responsible for enforcing security policies, providing an advertising and brokering service for agents and for providing a migration policy into and out of the domain.

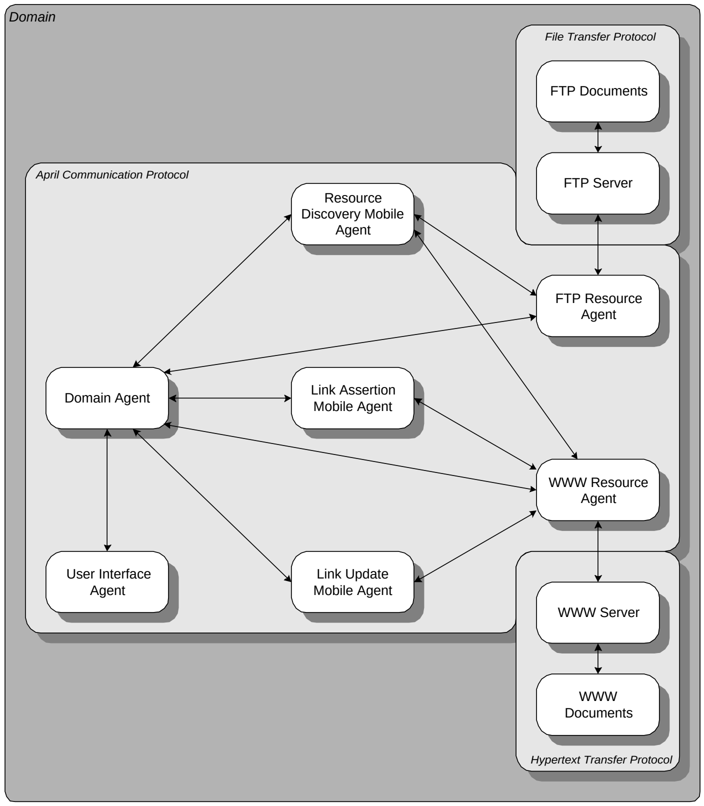

Figure 1 : Core Architecture Components and Example Agents

- *Resource Agent*. A static agent that abstracts and controls access to a given information resource or set of information resources. The resource agent is responsible for advertising (through the local domain agent) the services offered by the resource, for providing an ontological description of those services and for providing complete access to the resource.
- *Mobile Agent*. An agent that abstracts the DIM tasks of a user from the agent interactions between components of the architecture. Mobile agents possess the ability to migrate to the most suitable location to reduce network traffic, to undertake the tasks that they have been given by the user interface agent and to transmit their findings or results to the user interface agent.
- *User Interface Agent*. A semi-mobile agent that abstracts the user from the details of the mobile agent architecture. The user interface agent is responsible for launching, monitoring and controlling a user's mobile agents, for pre-processing any results from a mobile agent into a form appropriate to the user and for being available at all times so that a user's mobile agent can always get in contact.

As can be seen in figure 1, there are a number of agents executing within the domain. The user interface agent provides the user with a window onto the status and activities of their agents and distributed information. Here, three mobile agents are in operation, whether launched directly by the user or proactively by the user interface agent. Two resources are evident, a collection of FTP documents which are abstracted through an FTP resource agent that interacts with the FTP server and mobile agents—the WWW resource agent interacts in a similar manner.

The three mobile agents are performing DIM-related tasks; the Resource Discovery mobile agent is querying the FTP and WWW resource agent for a particular search term, the Link Assertion mobile agent is traversing the user's document set from a given root hypermedia link and determining which links are valid or invalid, and the Link Update mobile agent is updating links with a user's FTP and WWW documents that have changed recently.

The given example is restricted to a simplistic case within a single domain. An extra agent, the *gateway agent*, manages a group of related domains (called a *gateway domain*) and provides for addressing and management of agents across domains. The gateway agent can, for example, protect a corporation where there are a number of domains which represent the individual departments within the organisation; the gateway agent could restrict the types of mobile agents wishing to enter or leave any domain it was managing.

In the context of multiple domains and gateway domains, the concept of mobility becomes more appropriate since not all of the resources that a given agent needs to access will be within the local domain. Client-server technologies have some inherent problems when considering programming agents [2], primarily that since servers cannot provide the exact service that a client may require, a client must make multiple calls to the server to elicit the required end service. This results in intermediate data being transferred across the network, which is inefficient. If the client can move to the local area where the server resides, then the distance that the intermediate data has to traverse can be considerably reduced. Harrison [5] highlights these advantages as; *efficiency* through reduced network traffic, *persistence* through longevity, *peer-to-peer communication* in which agents are clients and/or servers as appropriate, and increased *fault tolerance* since mobile agents do not need to maintain synchronous connections. These details make mobile agents particularly suited to operating in large-scale networks.

4. Conclusion

In this paper we have highlighted the need for and the characteristics of Distributed Information Management. We have developed and are currently experimenting with an architecture that supports both static and mobile agents to support user-related DIM tasks. The abstraction that this architecture affords allows agents to be dynamic, modular and flexible in nature, which allows then to react and be proactive in how they interact with their environment.

The future work of this research lies in extending the example DIM agents described in section 3 into more complex agents which work together to discovery, manage, navigate and integrate the distributed information of a user in a useful and intelligent manner.

References

[1] BROOKS, R. A., Intelligence without Reason. *In:* Proceedings of the Twelfth International Joint Conference on Artificial Intelligence, Sydney, Australia, pages 569-595, 1991.

[2] DALE, J., A Mobile Agent Architecture to Support Distributed Resource Information Management, PhD Mini-Thesis, University of Southampton, UK, May 1996.

[3] DALE, J. and DeROURE, D. C., A Mobile Agent Architecture for Distributed Information Management. *Submitted to:* IEEE Internet Computing Special Edition on Internet Agents, March 1997.

[4] DeROURE, D. C., HALL, W., DAVIS, H. C. and DALE, J., Agents for Distributed Multimedia Information Management. *In:* Proceedings of the First International Conference on the Practical Application of Agents and Multi-Agent Technology, London, UK, pages 91-102, 1996.

[5] HARRISON, C. G., CHESS, D. M. and KERSHENBAUM, A., Mobile Agents: Are they a Good Idea?, IBM Research Report, IBM Research Division, 1995.

[6] JENNINGS, N. R., Agent Software, Keynote Speech. *Presented at:* The First International Conference on the Practical Application of Agents and Multi-Agent Technology, London, UK, 1996.

[7] McCABE, F. G. and CLARK, K. L., April—Agent Process Interaction Language. *In:* Intelligent Agents, Lecture Notes in Artificial Intelligence, 890, Wooldridge, M. J. and Jennings, N. R., *Eds.*, Springer-Verlag, pages 324-340, 1995.

[8] Office of Science and Technology, Technology Foresight Panel on IT and Electronics, Volume 8, HMSO Publications, 1996.

[9] WOOLDRIDGE, M. J. and JENNINGS, N. R., Intelligent Agents: Theory and Practice. *In:* Knowledge Engineering Review, 10(2), June 1995.