Realm: The Resource Management System for the Intelligent Room

Krzysztof Gajos, Luke Weisman, Stephen Peters & Michael Coen

Artificial Intelligence Laboratory
Massachusetts Institue Of Technology
Cambridge, Massachusetts 02139

http://www.ai.mit.edu

The Problem: The problem is that of assigning abstract services provided by various devices (physical and computational) to requestors. Each device can provide a number of services. Each kind of abstract service can potentially be provided by a number of different devices with some variation of quality. The problem is to find the right devices for the requests while keeping conflicts to a minimum.

Motivation: As computers become cheaper, smaller and more powerful, they begin to appear in places where until recently we did not expect to find them. The idea of ubiquitous computing and smart environments is no longer a dream and has long become a serious area of research and soon this technology will start entering our every day lives. One of the major obstacles preventing this technology from spreading is the fact that different smart spaces will have very different kinds of devices available. This lack of consistency will make it very difficult to write portable applications for intelligent spaces.

Previous Work: There are several other systems that deal with Resource Management issues in smart environments. Examples of such systems include Jini[1]. Open Agent Architecture (OAA)[7], Hive[8], or Resource Description Framework (RDF)[6]. None of them, however, address all of the issues identified by us as crucial in designing a resource management system for intelligent spaces. Jini is only a communication, description and discovery framework with no management capabilities of its own. OAA manages tasks and not services. Hive does not have any explicit mechanism for automatic conflict avoidance and RDF only provides a description framework without any explicit support for reasoning.

Approach: Realm[5] is composed of three major components: the knowledge part, the constraint-satisfaction engine and infrastructure for integrating with the Metaglue system[4].

The knowledge part is mostly composed of templates for describing agents, services and requests. It also contains a number of rules and functions for matching services to requests and a set of meta control rules. The knowledge part stores information about the environment and is responsible for suggesting candidate services for all requests.

The constraint satisfaction engine uses information from the knowledge part about the current state of the room and about requests and the corresponding candidate services. On the basis of this information, it produces a model of the world with as many requests satisfied as possible and hands it back to the knowledge part. The engine uses depth-first search method with local constraint propagation.

The Metaglue interface is responsible for mediating service and information requests between Metaglue agents and Realm.

Difficulty: Some of the problems in deploying Realm are with the real world complexities that Realm has to deal with. Those complexities include name collisions, similar agents working on behalf of multiple people, and control issues associated with resource swapping.

Impact: The main contributions of the research on Realm were the following:

1. Realm provided the Room's software infrastructure with a powerful layer of abstraction. With Realm in place, the Room's applications became portable and plans have been made to install Room-like infrastructure in several other spaces ranging from individual faculty and student offices with very minimal equipment to a medium size conference room filled with a variety of high-tech devices. Also, Realm simplifies enormously the process of creating higher-level applications for the Room by relieving the software designers from having to explicitly choose or control devices needed by their applications.

2. Through the research on Realm we identified a number of general principles and design requirements for resource management systems for intelligent spaces.
3. Realm enables introduction of a new paradigm within the intelligent Room research: one in which the user is viewed as one of the resources in the system. This is important because human attention is limited, much more than our ability to present information, and thus it has to be used efficiently.

Research Support: This material is based upon work supported by the Advanced Research Projects Agency of the Department of Defense under contract number DAAD17-99-C-0060, monitored through Rome Laboratory. Additional support was provided by Microsoft Corporation.

References:

[1] Arnold, K.; O'Sullivan, B.; Schifler, R.; Waldo, J.; and Wollrath, A. The Jini Specification. Addison-Wesley, Reading, MA, 1999.

[2] Coen, M. Building Brains for Rooms: Designing Distributed Software Agents. Proceedings of the Ninth Conference on Innovative Applications of Artificial Intelligence. IAA/97. Providence, RI. 1997.

[3] Coen, M. Design Principles for Intelligent Environments. Proceedings of the Fifteenth National Conference on Artificial Intelligence. AAAI-98. Madison, Wisconsin. 1998.

[4] Coen, M., Phillips, B., Warshawsky, N., Weisman, L., Peters, S., and Finin, P. Meeting the Computational Needs of Intelligent Environments: The Metaglue System. Proceedings of the 1st International Workshop on Managing Interactions in Smart Environments. MANSE99. Dublin, Ireland. 1999.

[5] Gajos, K. A Knowledge-Based Resource Management System for the Intelligent Room. Master's Thesis. Massachusetts Institute of Technology. Cambridge, MA. 2000.

[6] Lassila, O., and Swick, R. (eds.) Resource Description Framework (RDF) Model and Syntax Specification. W3C Recommendation, 22 February 1999. `http://www.w3.org/TR/REC-rdf-syntax/`

[7] Martin, D.; Cheyer, A.; Moran, D. The open agent architecture: A framework for building distributed software systems. Applied Artificial Intelligence. vol. 13, pp. 91–128, January-March 1999. http://www.ai.sri.com/pubs/papers/Mart:Open/document.ps.gz.

[8] Minar, N.; Gray, M.; Roup, O.; Krikorian, R.; and Maes, P. Hive: Distributed Agents for Networking Things. Proceedings of ASA/MA'99, the First International Symposium on Agent Systems and Applications and Third International Symposium on Mobile Agents. August 1999.