An Agent Architecture for Dynamic Re-design of Agents

Frances M.T. Brazier, Catholijn M. Jonker, Jan Treur, and Niek J.E. Wijngaards

Vrije Universiteit Amsterdam,
Department of Mathematics and Computer Science, Artificial Intelligence Group
De Boelelaan 1081a, 1081 HV Amsterdam, The Netherlands
Email: {frances, jonker, treur, niek}@cs.vu.nl URL: http://www.cs.vu.nl/~{frances, jonker, treur, niek}

Abstract

This paper presents a generic architecture for an agent capable of designing and creating new agents. The design agent is based on an existing generic agent model, and includes a model for design, in which strategic reasoning and dynamic management of requirements are explicitly modelled. The model includes an explicit formal representation of (1) requirements that can be formulated for agents and multi-agent systems, and (2) design object descriptions of a (part of a) multi-agent system.

1. Introduction

Agents that are able to dynamically design and create new agents, or to dynamically modify existing agents can be very useful. For example, Internet agents that are capable of dynamically creating new agents to retrieve specific information, or agents that are capable of creating new interface agents tuned to specific users, are agents of this type. To design an agent capable of (re)designing and creating agents, the following aspects must be addressed:

- an *agent model* as a basis for the design agent,
- a *model* of the *design task* used by the design agent,
- explicit representation (within the design agent) of *requirements* on agents to be (re)designed and *knowledge to derive refinements* of these requirements,
- explicit representation (within the design agent) of agent *design object descriptions*, and *knowledge to derive properties* of design object descriptions,
- a *model and implementation* of the execution of the *creation action* that actually creates (while the multi-agent system is running) the new or modified agent on the basis of the design.

A generic architecture for a design agent capable of designing and creating new agents, has been modelled, specified and implemented using the compositional development method for multi-agent systems DESIRE, cf. [1]. After a new agent has been designed by the design agent, this design is effectuated by execution of a creation action initiated by the design agent in the external world. After this creation action the multi-agent system functions with the additional agent.

The redesign process is illustrated in Figure 1. The left box contains the multi-agent system (consisting of the two agents Personal Assistant and Client, and External World) before modification, the right box depicts the multi-agent system after modification (with an additional agent D). The Personal Assistant (PA) is the design agent. It first reasons to obtain a plan for the desired modification (the larger grey box within the left box depicts the reasoning process), and executes this plan in interaction with the External World.

2. A generic model of a design agent

The generic model of a design agent proposed in this paper is a refinement of a generic agent model and includes a refinement of a generic model of design.

The *generic agent model* (cf. [2]), distinguishes a number of generic agent tasks, in addition to specific tasks for which an agent is designed (for example, information retrieval). Instead of designing each and every new agent individually from scratch, this generic agent model can be used to structure the design process: the acquisition of a specific agent model is based on the generic structures in the model.

The *generic model of design* (cf. [4]) distinguishes reasoning about requirements and their qualifications, reasoning about design object descriptions, and reasoning about the design process. An initial design problem statement is expressed as a set of initial requirements and requirement qualifications. *Requirements* impose conditions and restrictions on the structure, functionality, and behaviour of the *design object* for which a structural description is to be generated during design.

3. Representation of requirements and design object descriptions

For the (re)design of a multi-agent system, requirements are formulated in terms of abilities and properties. Abilities and properties can apply to individual agents, the external world, an individual agent in relation to the agents and the world with which it interacts, the world in relation to the agents with which it interacts, or a multi-agent system as a whole. Abilities of agents such as co-operation, bi-directional communication, and world interaction are often needed for agents to jointly be able to perform a certain task. In addition to knowledge of abilities and properties, a design agent needs knowledge of refinements of abilities. This knowledge is formally represented within the design agent; cf. [3].

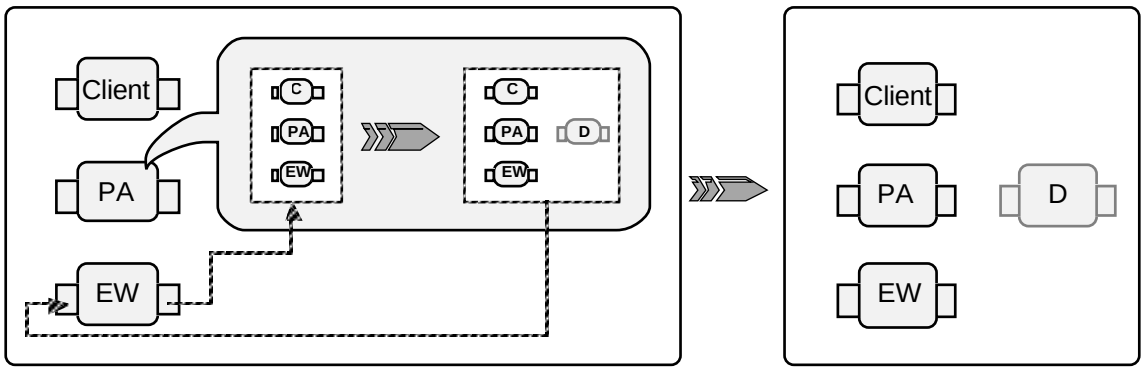

Figure 1 Redesign of a multi-agent system: agent PA modifies the structure of the multi-agent system

The implication of designing (parts of) a multi-agent system, is that the multi-agent system itself is the object of design. In this paper the design object description is a compositional object description, formally represented within the design agent. This compositional structure guides the re-design process.

4. Creation of a designed agent

After the design process within the design agent has been completed, the agent decides to effectuate the modifications. As discussed in [6], effectuation of the modification of the design can be modelled by changing the material representation of the multi-agent system within the external world. To change the number of agents and their characteristics, the external world has to adapt the executable specification of that system while the system is running. This implies that the parts of the system that are affected by the modifications need to be interrupted, their information states stored, after which the executable specification of those parts need to be modified, and the modified system need to be reactivated with the correct information states.

5. Discussion

Most of the research in the area of dynamic agent creation is based on a genetic programming approach; e.g., [5], [8]. The approach taken in this paper is that to create new agents, an existing agent must be capable of designing a new agent on the basis of a model for design and then be capable of bringing this agent to life. The integration of re-design on a conceptual and logical level and run-time modification of the system at the implementation level is an important distinguishing aspect of the approach presented in this paper. This is in contrast to, on the one hand, conceptual and logical approaches for which no direct connection to executable code exists, and, on the other hand, to approaches that address agent creation at an implementation level.

Acknowledgements

The authors are grateful to Pieter van Langen for his contribution to the generic model of design, and to Lourens van der Meij and Frank Cornelissen for their support of the DESIRE software environment. This research has been (partially) supported by NWO-SION within the project 612-322-316 REVISE.

References

[1] Brazier, F.M.T., Dunin-Keplicz, B.M., Jennings, N.R., and Treur, J., Formal Specification of Multi-Agent Systems: a Real World Case, In: Lesser, V., (ed.), Proceedings of the First International Conference on Multi-Agent Systems, ICMAS'95, MIT Press, 1995, pp. 25-32. Extended version in: Huhns, M., and Singh, M., (eds.), International Journal of Co-operative Information Systems, IJCIS, vol. 6 (1), special issue on Formal Methods in Co-operative Information Systems: Multi-Agent Systems, 1997, pp. 67-94.

[2] Brazier, F.M.T., Jonker, C.M., and Treur, J., Formalisation of a cooperation model based on joint intentions. In: [7], 1997, pp. 141-155.

[3] Brazier, F.M.T., Jonker, C.M., Treur, J. and Wijngaards, N.J.E., The Role of Abilities of Agents in Re-design. In: Gaines, B. and Musen, M., (eds), Proceedings of the 11th Knowledge Acquisition Workshop, KAW'98, Banff, 1998.

[4] Brazier, F.M.T., Langen, P.H.G. van, Ruttkay Zs., and Treur, J., On formal specification of design tasks. In Gero, J.S., and Sudweeks, F., (eds.), Artificial Intelligence in Design, Proceedings of AID'94. Kluwer Academic Publishers, 1994, pp. 535-552.

[5] Cetnarowicz, K., Kisiel-Dorohinicki, M., and Nawarecki, E., The Application of Evolution Process in Multi-Agent World to the Prediction System. In: Tokoro, M., (ed.), Proceedings of the Second International Conference on Multi-Agent Systems, ICMAS'96, MIT/AAAI Press, Menlo Park CA, 1996, pp 26-32.

[6] Jonker, C.M., and Treur, J., Modelling an Agent's Mind and Matter. In: Boman, M., Velde, W. van de, (eds.), Proceedings of the 8th European Workshop on Modelling Autonomous Agents in a Multi-Agent World, MAAMAW'97, Lecture Notes in AI, vol. 1237, Springer Verlag, 1997, pp. 210-233.

[7] Müller, J.P., Wooldridge, M.J., and Jennings, N.R., (eds.), Intelligent Agents III (Proceedings of the Third International Workshop on Agent Theories, Architectures and Languages, ATAL'96), Lecture Notes in AI, vol. 1193, Springer Verlag, 1997

[8] Numaoka, C., Bacterial Evolution Algorithm for Rapid Adaptation. In: Van de Velde, W. and Perram, J.W., (eds.), Proceedings of the 7th European Workshop on Modelling Autonomous Agents in a Multi-Agent World, MAAMAW'96, Lecture Notes in Artificial Intelligence, vol. 1038, Springer Verlag, 1996, pp. 139-148.