

Structural Analysis of Agent Oriented Methodologies

M. Elammari, R. Elsaeti

Information Technology University of Benghazi, Benghazi, Libya

Abstract

Agent-oriented software engineering (AOSE) systems are often distinguished with intelligence, autonomy, and reasoning. The main purposes of development AOSE methodologies are to facilitate the process of developing software lifecycle, and increase the quality of software products. Now, there are no agreements on constructing agents or modeling them in the lifecycle development process. Large number of AOSE methodologies with various backgrounds, was available in the last years. However, assessment of these methodologies faced several difficulties; the completeness of various methodologies varies explicitly, for this, comparing agent methodologies is more difficult; as they cover different aspects or differ in their terminology. The main objective of this study is to analyze several agent-based methodologies in an attempt to assess and understand the relationship among these agent-oriented methodologies by performing structural analysis.

1. Introduction

In the last few years, agent researchers have made clear progress in proving a theoretical and practical understanding of many aspects of software agents and multi-agent systems (MASs). Also they attempted introduce many diverse methodologies to change software development from an ad-hoc practice to high-quality software.

Software developers faced numerous problems regarding AOSE such as specifying which methodology is the best approach to take to develop a solution. As well as, there is a lack of consistency between the different analysis and design methods that are currently available.

In this paper we select four agent-based methodologies ROADMAP methodologies [1] ROADMAP (Role-Oriented Analysis and Design for Multi-Agent Programming) is emerged in 2002 as a methodology for agent oriented analysis and design, HILM methodologies [2] An agent-oriented methodology: High-Level and Intermediate Models (HLIM) is an AOSE methodology, which is design specifically for agent- based system development, Styx methodologies [3], is a detailed AOSE methodology, and SONIA [4] (Set of models for a Natural Identification of Agents)

methodologies. Each of the selected methodologies has different strengths and weaknesses, and different specialized features to support different aspects of their intended application domains. We performed structural analysis for selected methodologies based on [5] and investigated their relevancy to agent-based systems to identify the similarities and differences among selected agent-oriented methodologies.

2. Structural Analysis

As a result of this study, clearly no single methodology is suitable to all applications. However, with application complexity increasing, we expect future applications to have an increasingly large number of aspects which cannot be addressed by a single methodology alone. To introduce engineering support for such applications, specialized features to address different aspects should be collected together from various methodologies.

In this section we will focus attention on this issue. It is important to identify common core of models and processes and different elements of the participant's agent oriented methodologies. This enables us to determine the similarity among these methodologies and features of each methodology.

2.1 Structural Analysis - The Commonalities

● Visual Representation

When system requirements have been completed, the next step introduced by methodologies, a model that will translate the system's outline from an external point of view without entry to more detail.

The representation will clearly facilitate communication and exchanging views among developers on one hand, and developers and traditional users on the other hand.

Of the four selected agent oriented methodologies, HLIM and Styx make use of this technique; both use UCMs, which are suitable for high level visual representations, as a starting point for generating more detailed visual descriptions. In HLIM Use-case maps are used to model the high-level activities because of their ability to easily and successfully depict the design of complex systems, and provide a powerful visual notation for a review and detailed critique of the design.

HLIM exploits the mapping UCMS to agent concepts, where each component in UCM meaning in agent systems, where exist strong relation among concepts which became in agent technology and UCM, the UCM components can be translated to agent system, each component in UCM represent agent responsibly and static stub represent tasks, dynamic stubs represent sub-goals, the path segment which exist between agents means that it found coordination's among these agents, plan concept can be represented by plug-ins, also the path segments which are crossing component represent plans, precondition and post-condition considered beliefs.

Also Styx is using the UCM to the same purpose in HLIM; it uses UCM to give a high-level review for the system. As it exists in standard OO approach, components are drawn as boxes, while paths are drawn as lines crossing various components. The start of a path is indicated by a solid circle, while the end point is indicated by a

strong line. When a path crosses a component that component is assigned one or more responsibilities associated with the path. After identifying the roles which agent may play in the requirements specification, these roles can represent as UCM components, while interactions between roles that are defined in the requirements specifications represent paths in the UCM.

● **Social System Structure**

The increased interest in using MASs in all kinds of computer system applications requires a stronger foundation in MASs software engineering. One of the challenges of adjunction the agent-oriented paradigm with software engineering field represents in its ability in designing and implementing complex open systems.

So, it is very necessary for an agent oriented methodology to introduce a clear abstraction for the comprehension of the whole structure of the system.

Indeed most of the four methodologies address this respect to some extent. ROADMAP methodology provides strong support for engineering complex open systems it emphasize the social aspects of an agent system.

HLIM on the other hand, has the internal agent diagram, which describes agents in terms of their goals, beliefs, plans and tasks. Similarly, ROADMAP services diagram, which describes the services that agents introduce to each other. The agent relationship diagram in HLIM depicts inter-agent dependencies and jurisdictional between agents, similarly, Styx introduces the role relationship diagram to describe the dependency among the roles that exist in the system and agents that may play it. More importantly, HLIM has conversational diagram depicting the social structure of the system as a collections of agents and messages among them.

On interaction terms, SONIA provides interactions among agents and agents and objects on other term. ROADMAP provides interactions diagram, based on AUML interaction diagrams. The conversational model in HLIM defines the sequence of the messages exchanged between agents.

● **Knowledge Representation**

To support complex open systems agent methodologies need explicit models to describe the domain knowledge and the execution environment.

Of the four selected agent oriented methodologies, ROADMAP and SONIA specified models to represent knowledge; they cover those features in different ways. The knowledge model in ROADMAP formalizes the knowledge aspect of the system. It will also guide the system to be designed and implemented at a higher level. The knowledge model consists of a knowledge component hierarchy, and a set of schema for each knowledge component. It can be added into the use-case model, where required knowledge components to achieve the scenarios are identified. Knowledge components can also be assigned to roles, to model knowledge distribution in the organization.

The knowledge components in SONIA came as a result of grouping concepts and associations in the structural model. The knowledge components will be used internally or shared by the agents.

● **Environment Concept**

The concept of environment is essential for MAS, agents are developing in an environment, complex open systems usually have highly dynamic and heterogeneous environments. By formally describing the environment, we create a knowledge foundation on which environment changes are handled consistently. So an agent systems need to have models to represent the environment in which it operates.

Despite the importance of developing an environment model, only ROADMAP and SONIA introduce models to describe environment.

ROADMAP integrates it into the methodology's environment process; it is designed for open systems, so it presents a holistic model of the execution environment to the developers. The purpose of ROADMAP environment model is providing a holistic description of the system environment.

It handles this issue in a different way; it defines a tree hierarchy of zones in the environment, and a set of zone schema to describe each zone in the hierarchy. A zone schema includes a text description of the zone, and the following attributes: static objects, objects, constraints, sources of uncertainty and assumptions made about the zone.

Static objects are entities in the environment whose existences are known to the agents, with which the agents do not interact explicitly. Objects are similar entities with which agents interact. Sources of uncertainty in the environment are identified and analyzed. The zone hierarchy uses object oriented-like inheritance and aggregation to connect zones and different objects inside zones. For each use-case scenario, there exists a new zone and update new information into the existing zones.

SONIA defines the system external entities and their interactions with the system via the environment model.

2.2 Structural Analysis - The Differences

● **Use Cases Model in ROADMAP**

Creating use-cases have been proven to be an effective technique in OO methodologies. It supports the requirement gathering. Of the four selected methodologies, ROADMAP makes use of this technique.

According to object oriented methodologies, the use-case model involves generalized graphical diagrams and specialized text scenarios.

The scenarios illustrate the system response given a series of user's actions. ROADMAP introduces new concepts such as zones and roles that can be used in the use-cases. It aims to obtain the richer agent behaviors and provides sufficient support, allowing the complexity and knowledge in the MAS to scale modularly to any level required.

For this, the using of use-cases is different from the traditional OO method. There the use-case dose not depicts the user interacting with the software system to do work. It depicts the user interacting with a team of abstract ideal agents.

● **Conversational & Contract Model in HLIM**

In the MASs, agents need to communicate with each others to work together towards a common goal and organizing problem solving activity for beneficial interactions are

exploited. So its important for agent methodologies to address this issue. In contrast to other selected methodologies, HLIM covers explicitly the conversational process. Additionally, it introduces another model (contract model) that specifies, in a high-level language, obligations and authorizations between the different agents, and the services provide to each other.

The purpose of the conversational model is to identify what messages are exchanged in order for the agents to cooperate and negotiate with each other. More importantly, it has a detail depicting the social structure of the system as a collection of agents, messages between them. The conversational model basically identifies what messages are exchanged to achieve the dependencies and the jurisdictional relationships among agents; it specifies messages to each relationship. The content of the conversational messages is determined by the plans that satisfy the dependencies that are obtained in the internal agent model.

The contract model aims to specify the expectations of how agents can achieve the dependencies defined by the dependency model additionally the expectations of agents when they play the roles defined by the jurisdictional model.

● **Deployment Model in Styx**

In the multi-agent systems, agent may play one or more roles to accomplish their tasks; so agents need to know what roles they will play? Deployment model in Styx connects among agents and their roles by specifying a many-to-many mapping between them. This model specifies what agents will exist in the system, and what roles they will play.

● **Hierarchy in ROADMAP**

To provide an integral description for the target system, facilitating iterative refinement of the system through different levels of abstraction, and present a holistic model of the execution environment to the developers, ROADMAP used the hierarchy concept in three models environment knowledge, and role models.

3. Conclusions

In our view all of the four agent oriented methodologies are clearly agent-based, the design is widely regarded as an agent-oriented methodology and most of them consider the social aspects.

In general, all four methodologies provide a reasonable support for basic agent-oriented attributes. Social features addressed in all four agent oriented methodologies provide strong support for most social criteria as communication, communication language and relationship.

The four methodologies also share several common features; for example HLIM and Styx share the use of a visual representation technique. In addition, all of the four methodologies provide models to cover the social system structure.

Distinguishing features also exist such as, ROADMAP which is unique among the four methodologies in having roles and environmental zones. It used these concepts to model trust in open organizations of agents. For the other three methodologies, there

is still more room for improvement with respect to these issues. Also Styx offers a deployment diagram to specify which agents will exist in the system, and what roles they will play.

Additionally HLIM is unique among the four methodologies, by providing mechanisms to cover conversational process. Additionally it introduces a contract model to specify, in a high-level language, obligations and authorizations between the different agents about the services provided to each other.

4. References

- [1] Juan, T., Pearce, A. and Sterling, L. Extending the Gaia Methodology for Complex Open Systems, Proceedings of the 2002 Autonomous Agents and Multi-Agent Systems, Bologna, Italy, July 2002.
- [2] Elammari, M. and Lalonde, W. An agent-oriented methodology: High-level and intermediate models. In Wagner, G. and Yu, E. editors, Proc. of the 1st Int. Workshop. on Agent-Oriented Information Systems., 1999.
- [3] Bush, G., Cranefied, S. and Purvis, M. The Styx Agent Methodologies, university Otago ISSN 1172-6042 January 2001
- [4] Frutos, S. Modelo de Diseno duna Arquitectura Multi-Agente Basado en un Modelo de Sociedad de agents. PhD Thesis. Universidad Politecnica de Madrid, Spain, 2003.
- [5] Dam, K.H. Evaluating and Comparing Agent-Oriented Software Engineering Methodologies. MSC Thesis, RMIT University, Australia, 2003.
- [6] Berard, E.V. A comparison of object-oriented methodologies. Technical report, Object Agency Inc., 1995.
- [7] Shen, Weiming and Douglas Norrie. Agent-Based Systems for Intelligent Manufacturing: A State-of the-Art Survey, Knowledge and Information Systems, an International Journal. Vol 1, Num 2. p 129-156, 1999.
- [8] Wand, Y., Weber, R. An Ontological Model of an Information System, IEEE Trans. on Software Engineering, Vol. 16, No. 11, pp. 1282-1292, November 1990.
- [9] Juan, T., Pearce, A. and Sterling, L. Extending the Gaia Methodology for Complex Open Systems, Proceedings of the 2002 Autonomous Agents and Multi-Agent Systems, Bologna, Italy, July 2002.
- [10] Bush, G., Cranefied, S. and Purvis, M. The Styx Agent Methodologies, university Otago ISSN 1172-6042 January 2001
- [11] Frutos, S. Modelo de Diseno duna Arquitectura Multi-Agente Basado en un Modelo de Sociedad de agents. PhD Thesis. Universidad Politecnica de Madrid, Spain, 2003.