An Agent-Oriented Requirement Modeling Method based on GRL and UML

Niu Xi  
Software Department  
Tsinghua University  
Beijing, China  
nx03@mails.tsinghua.edu.cn

Liu Qiang  
Software Department  
Tsinghua University  
Beijing, China  
lquiang@mail.tsinghua.edu.cn

Abstract — As a background of the research on complex information systems, this paper presents an agent-oriented requirements modeling method based on GRL and UML. This method divides requirements engineering into two stages: early requirements capture and late requirements analysis. During early requirements, GRL is employed to identify the roles and relationships between them, capturing the business goals and generating design alternatives. During late requirements specification, extended UML is employed to model the requirements at agent level of abstraction. During the transition from early requirements to late requirements, unified model mapping rules are set so as to keep the model consistent. To illustrate the detailed modeling process, a case study of 'network-based project manager' is presented.

Keywords: agent-oriented modeling, requirements engineering, GRL, extended UML.

1 Introduction

Software requirements have been repeatedly recognized during the past years as a real problem. Yet it remains very difficult to get high-quality requirements quickly and to keep them updated to reflect changing needs. Agent-oriented software engineering is a new pattern of developing complex software system. Agent-oriented modeling treats software at a higher level of abstraction compared to current models and languages [1](e.g., UML or architectural description languages). 'Agents' are more coarse-grained than 'objects'. Software agents have autonomy; they communicate, coordinate, and cooperate with each other to achieve goals. It is recognized that during software engineering, requirement analysis is rather critical. And so it is for agent-based software development. How do we quickly obtain and precisely express the requirements of users is a problem that is difficult to software designers and nowadays it is a hot issue in this field. In some recent researches, several agent-based frameworks has been proposed, such as KAOS, i* framework, Gaia and Tropos[2].

Recently, a new kind of idea about requirements engineering is proposed, i.e. requirements engineering can be separated into two phases: early requirements capture and late requirements specification. Early requirements include non-formal and non-functional requirements, focusing on the domain background information and the rationale and objectives of system;

Late requirement specification aims to provide formal requirement description, to elicit requirement specification and to check the completeness and consistency of the specification. However, most of current modeling methods emphasize on late requirement analysis while ignoring early requirements. Accordingly, several frameworks, such as i* and GRL, early requirements, is employed. The GRL (goal-oriented requirements language)[3] is a language for supporting goal- and agent-oriented modeling and reasoning about requirements, with an emphasis on dealing with non-functional requirements (NFRs). GRL describes the intention and interests of stake-holders according to the environments, thus helping software designers to learn the motives of system roles and the rationale between roles ('Why' question).

GRL is proposed for identifying and refining goals in early requirements. In this paper, it is employed as a framework for capturing the early-phase requirements of agent-based systems. On the other hand, our view of agents as the next step beyond objects leads us to explore extensions to UML. [4] and idioms within UML to accommodate the distinctive requirements of agents. The UML unifies and formalizes the methods of many approaches to the object-oriented software lifecycle and provide completeness and consistency for model. UML is a common tool for late requirements and suitable for eliciting requirements specification.

![Diagram](image_url)

Figure 1. Requirement modeling process.

To combine the early and late requirements analysis in software design, this paper presents a new agent-oriented modeling method based on GRL and extended