Guiding agent-oriented requirements elicitation: HOMER

Damien Wilmann and Leon Sterling
University of Melbourne
Department of Computer Science and Software Engineering
Email: [dmwilm, leon]@cs.mu.oz.au

Abstract

There has been a surge of interest in agent-oriented software engineering in recent years. The key area of requirements engineering for agent-based systems has received considerable attention. However, while notations and models for requirements have been discussed, there has been little attention focused on techniques for elicitation. This paper introduces HOMER, an approach for requirements elicitation that is explicitly agent-oriented. HOMER can be used with existing approaches for agent-oriented software engineering. In this paper, we show how HOMER may be integrated with a specific agent-oriented software engineering approach, ROADMAP. HOMER is aimed to increase the ease with which people may readily design and develop agent-based systems.

1 Introduction

There has been a considerable amount of research in expressing requirements for agent-oriented systems. Agent-oriented requirements analysis and specification has much support in the literature. Agents provide a more intuitive, human way to look at software systems [14]. Rolland et al [11] have proposed goal and scenario modeling, using EKD, to completely discover and specify requirements. Yu [18] argues that the concept of agent goals is a useful construct in RE. Yu points to a number of agent-oriented RE frameworks, such as KAOS [3], Albert II [6] and F3 [1]; all of which are used to model requirements using agent concepts. The CommonKADS methodology [13], used for knowledge based systems, also has support for agent modeling at the requirements specification level.

Each of these techniques, however, falls short of providing support for the actual elicitation of the requirements. The approach suggested by Rolland et al [11] comes closest, describing the activities that must be performed by a “Requirements Chunk Author” (RCA). The RCA could, conceivably, be the client, though this would assume that the client was involved with writing the requirements specification. While [11] states that requirements elicitation must be systematic, they do not say how this is to be achieved. The support that their processes provide is for analysis and specification only.

In this paper, we present HOMER (Human Oriented Method for Eliciting Requirements). HOMER is an elicitation technique that aims to provide elicitation support to existing agent-oriented requirements engineering frameworks and AOSE (agent-oriented software engineering) models with requirements engineering support. This paper shows how HOMER may be used to support ROADMAP, an AOSE methodology that includes support for the specification and analysis (but not elicitation) of agent requirements [2], [7]. HOMER uses organisational metaphors, such as the hiring of new staff, to elicit requirements from clients. This elicitation style discovers requirements that are easily translated into the goal and role models used by many AOSE frameworks, including ROADMAP.

This paper first looks at what HOMER is and how it is conducted in section 2. Section 3 then shows how HOMER may be integrated with an AOSE framework, using the specification models in ROADMAP as an example. Section 4 concludes the paper and discusses the work that is still to be done in the area of agent-oriented requirements elicitation.

2 Agent-oriented elicitation

The organisational metaphor has been widely used to analyse and specify requirements [17] [16] [13]. Dignum et al have proposed an approach for modeling organisations with agents in the field of knowledge management [4]. In [5] Donzelli and Bresciani use goal modeling to develop, during the analysis phase, an organisational view of agent oriented systems. How-
ever, no elicitation technique yet exists to explicitly elicit requirements for agents using organisations as the guiding metaphor. HOMER aims to address this lack. While HOMER is presented in this paper as an extension to ROADMAP, there is no reason that it could not be used to elicit requirements for any of the other existing agent-oriented requirements specification and analysis techniques.

2.1 Background

HOMER was developed in response to problems encountered during two projects at the University of Melbourne: Project ARMS and Daedalus. The first, ARMS (Agentlab Research Management System), was developed by the Agentlab to automate the management of their research students and associates and their agent-related activities. The second project, Daedalus, was a 4th year software engineering project\(^1\) to develop software for an automated home security system.

During the project concept phase in both projects, they were identified as projects where an agent-based architecture would be an appropriate choice. Both projects found, however, that during the requirements specification phase the elicited requirements were not of a form that was easy to specify using ROADMAP (our chosen AOSE framework for both projects). The elicitation had proceeded using the standard interview technique\([12]\), and did not capture completely the clients' desired solution. There were aspects of both systems, relating their open-ended and autonomous nature, that were hard for the clients to articulate. Furthermore, the requirements were elicited based on desired features. Specification into an agent framework required breaking down those features into goals and roles. This seemed an unnecessary step; since the clients felt comfortable reading specifications that use goals and roles, they should also be comfortable talking about their requirements in that manner.

To address these issues, we looked at interview questions that would more easily draw out the nature of these agent-oriented systems and lend themselves better to agent-oriented frameworks. The solution was an agent-oriented elicitation technique, HOMER.

\(^1\)The project-based subject, 433-440 Advanced Software Engineering Project, runs over an entire academic year. The subject is taught in the final year of a Bachelor of Engineering (Software) degree offered within the Department of Computer Science and Software Engineering. Students work in teams of 12-15 on an industry scale software project.

2.2 Aims

This section describes the aims that motivated us to develop the agent-oriented elicitation technique, HOMER. The following were our main goals in developing HOMER:

** Appropriateness:** HOMER’s main aim is to gather requirements for agent-friendly systems that are appropriate for use (specification) in an agent-oriented software engineering (AOSE) model. In order to be appropriate for non-technical clients, we aim to specify our requirements in a non-technical manner.

**Completeness:** We also aim to elicit a full set of requirements from the client. These requirements need to describe all the attributes of the desired system to solve the client’s problem. Even with supporting agent models for specification, it can be very hard to conceptualise and describe requirements - with the elicitation process typically involving many iterations between the agent model and system requirements\([16]\). Both Team Daedalus and Project ARMS have encountered similar difficulty in getting a complete set of requirements from clients.

**Comprehensibility:** Our final major aim is to gather requirements in a way that is easily understood by the client, no matter what their technical background. This also applies to the way in which we specify those requirements - though this is handled by ROADMAP (as well as the other agent-oriented requirements frameworks).

2.3 Elicitation technique

This section describes, using examples, how to elicit requirements using HOMER. The way in which HOMER integrates with AOSE frameworks is left to section 3. The discussion here includes how HOMER addresses the aims stated in section 2.2.

As stated above, HOMER uses the organisational metaphor to elicit requirements. Specifically, we use the metaphor hiring new staff which is described below. It is assumed, before using HOMER, that the project concept phase is complete and that the project is agent-friendly (see figure 1 in section 3). The client has already described, in general terms, their problem that requires an agent-oriented software solution.
Hire new staff

The elicitation technique that forms the basis for HOMER is based around the metaphor of hiring staff in an organisation. The idea is for the client to think about the problem in terms of which new staff would need to be hired in order to solve, or partially solve, their problem. For nontechnical clients, we believe this perspective will be a more intuitive way to conceptualise the system. For technically minded clients, this will help to elicit their requirements in a more abstract manner than they are perhaps used to. In the former case, this will help to achieve the comprehensibility aim stated in section 2.2 and in the latter case it will help to achieve the appropriateness aim - the requirements specified will be both agent-friendly and non-technical.

To discover what the client needs, ask the following questions:

1. If you were to hire more staff to handle your current problem, which positions do you need to fill?
   eg. Secretary.

2. For each position, we need to collect a “job description”:
   (a) What is the purpose of this position? What aspects of the problem will this person solve, or partially solve?
   eg. Keep the project manager’s schedule efficiently arranged and respond to all meeting requests on behalf of the project manager. This allows the project manager to attend as many meetings as possible. Generally speaking, this helps to achieve the overall objective of saving time and/or spending time more effectively.
   (b) What tasks will commonly be required?
   eg. Meeting scheduling.
   (c) For each task above:
      i. What subtasks make up this task?
         eg. For meeting scheduling: Reading emails regarding meetings, knowing meeting times and rearranging meeting times to allow for new meetings.
      ii. What constraints are there for this task?
         eg. Management meetings take priority, meetings must have at least 30 minutes slack time following the meeting.
   (d) Which other people in the company does this person rely upon?
      eg. Building supervisor (for room bookings).
   (e) Which other people in the company rely upon this person?
      eg. Project manager (for schedule information).
   (f) What knowledge of your company will this person require to correctly perform their tasks?
      eg. Room locations, room availability, project manager’s private schedule.
   (g) What resources, existing and new, are required by this person in fulfilling their position?
      eg. Room booking system (existing), filtered access to project manager’s emails (those relating to meeting bookings only).

3. What codes of behaviour must be observed by all of your employees? (This will get rather detailed, but it should apply to all “employees” and will only need to be asked once)
   eg. Do not overwrite one another’s room bookings.
   (a) Are there specific codes of behaviour for certain positions, and what are they?
      eg. The secretary must not divulge the project manager’s schedule.

4. What other rules and regulations must your company adhere to?
   eg. No spam or personal mail may originate from a company email account.

3 HOMER and AOSE

HOMER is used in the requirements phase of a software project, regardless of the software development lifecycle (SDLC) you are using. HOMER can be used whenever elicitation is required. ROADMAP is presented in this section as the specification technique of choice, and is described briefly in section 3.1.

For illustrative purposes only, figure 1 shows where in the waterfall model of the software development lifecycle (SDLC) [10] HOMER would be used in relation to an AOSE framework (shown here as ROADMAP)³.

3.1 ROADMAP

ROADMAP is an AOSE methodology being developed at the University of Melbourne. ROADMAP has

³Please note that the authors do not recommend the use of the waterfall model. It is used here as an illustration because of its simplicity. It should be easy to see where HOMER would fit in with other SDLC models, such as the spiral or incremental model.
been chosen by the authors as the illustrative example for using HOMER because of their familiarity with this approach.

ROADMAP views agent-oriented systems as a collection of models at each stage of the SDLC. The models are divided into three areas: Domain specific models, application specific models and reusable service models. Each type of model has support in the requirements analysis and architectural design phases. Figure 2 shows these models conceptually. The horizontal dashed lines show the separation of the phases in the SDLC. As shown in figure 2, HOMER provides support for ROADMAP’s goal and role models. These models are the application specific components of ROADMAP’s requirements analysis and specification layer [8].

The goal model in ROADMAP is a high level overview of the system requirements. The model includes both system and quality goals (orthogonal to functional and non-functional requirements). The role model (or role schema) in ROADMAP is analogous to a position description [8]. It includes all the goals that are associated with that position as well as the constraints involved in fulfilling those goals. A sample goal model and role schema are included in section 3.2.

For more details on ROADMAP, readers are encouraged to view the works of Juan [7], Chan [2] and Kuan [8].

3.2 Integrating HOMER

This section shows how HOMER may be integrated with an AOSE framework, using ROADMAP as the example. Once HOMER has been used to drive the elicitation, ROADMAP’s role and goal models can be used for specification and analysis of requirements during the requirements phase. The questions described in section 2.3 have a direct realisation in the ROADMAP goal model and role schema.

Answers to question 2(a) are realised by the goal model. Figure 3 shows an example goal model using the secretary elicited in section 2.3.

Similarly, all other questions presented in section 2.3 have a realisation in ROADMAP’s role schema (also called the role model). The way in which these ques-
tions are realised is shown in table 3.2, again using the secretary example.

<table>
<thead>
<tr>
<th>Role Name</th>
<th>Secretary</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Manages the project manager’s schedule</td>
</tr>
<tr>
<td>Responsibilities</td>
<td>Scheduling meetings:</td>
</tr>
<tr>
<td></td>
<td>• Responding to emails requesting meetings;</td>
</tr>
<tr>
<td></td>
<td>• Booking rooms for meetings;</td>
</tr>
<tr>
<td></td>
<td>• Knowing the project manager’s current schedule; and</td>
</tr>
<tr>
<td></td>
<td>• Rearranging the project manager’s schedule to allow for new meetings.</td>
</tr>
<tr>
<td>Constraints</td>
<td>Management meetings take priority over all other meetings (i.e. may always be scheduled in place of normal meetings). When entering meetings into the project manager’s schedule, the secretary must also schedule a 30 minute block of “slack” time. Room bookings must be made through the building supervisor. The project manager’s schedule may only be revealed to the project manager.</td>
</tr>
</tbody>
</table>

Table 1. Example role schema for the secretary

4 Conclusion

Agent-oriented requirements engineering is a useful abstraction that allows clients to view their software solution on familiar, “human” terms. Much work has been done on for agent-oriented requirements specification and analysis. It is the focus of requirements engineering frameworks such as EKD [11], KAOS [3], Albert II [6] and F3 [1]. Agent specification and/or analysis also has support in agent methodologies, such as ROADMAP [2], Gaia [15] [19] and Prometheus [9].

Despite the recognised need for agent-oriented requirements engineering, however, no support currently exists for a technique for eliciting these requirements. Analysts are assumed to know the best way to elicit requirements within their chosen agent-oriented requirements engineering framework. HOMER addresses this lack, providing a technique for analysts to use when eliciting requirements for specification using an AOSE methodology.

Future work

Future work should follow in extending HOMER to support other agent-oriented requirements frameworks. Likely candidates include Gaia [15], upon which ROADMAP was based [7] and EKD [11], which uses goal modeling and task analysis to analyse and specify requirements. HOMER should also be extended to include other specification aspects of ROADMAP, such as the social model in figure 2. It would be a natural extension to ask the client in what way they expect the employees elicited with HOMER to interact (both with other software agents and with real people).

Acknowledgments

Support for this paper came from a number of sources. These include: the contributors to Project ARMS, notably Shanika Karunasekera and Pei Pei Kuan; our ROADMAP guru, Thomas Juan; Liz Sonnenberg for her invaluable advice; the Agentlab group at the University of Melbourne; and finally, our industry partner, Adacel Technologies in financially supporting our project, “Towards Invisibly Intelligent Appliances via Experience-Based Computing”, ARC Reference Number LP0348797.

References


