

ENHANCED GOAL BASED MODELING TECHNIQUE

Kavita¹, Shikha² & Sunila³

Software engineering is an engineering discipline that is concern with all aspect of software production. Requirement engineering is a branch of software engineering that concerned with real world goals and also deals with discovery and specification of the objectives for the system under development.

In this paper we present the overview of requirement engineering framework with tropos methodology that support tool for the tropos methodology itself. This methodology is based on the agent based and goal based requirement engineering framework that improve the quality of the software product. This paper presents a framework that guides the requirements engineer to model the requirements engineering process in an effective manner.

Keywords: Software Development, Agent-oriented Systems, Requirements Engineering Methodology and Process, Goal Oriented Requirement Engineering, Soft and Hard Goals

1. INTRODUCTION

Requirement engineering is concerned with the identification of goal to be achieved by system. Currently in requirement engineering the attention is being more and more on the understanding of the problem by studying in existing organizational setting in which the system will operate. Requirement engineering started with the study of what the system should do.

Requirement engineering(RE) is a discipline that provide a set of process to identify and engineer requirement, not only manage but also analyze, validate and traceable. The main strategy of requirement engineering on how to engineer a requirement rather than how to manage the requirements.[1][2] The process involve in requirement engineering include domain analysis, elicitation, specification and documentation. Requirement engineering is the key bridge between the user, system, analyst, developer or other stakeholders of software engineering. [3]

This paper presents the brief of main concept of RE and tropos methodology developed to support the RE task. This methodology is based on agent and goal based framework that improve the quality of the system. This framework is simple, effective and represents the graphical notation. Tropos deals with all the phases of system requirement analysis and all the phases of system design and implementation in a uniform and homogeneous way, In order to ensure the quality of a software requirements specification, there needs to be emphasis on implementing engineering disciplines into the RE process by using various techniques and methodologies.

2. RELATED WORK

A requirement is a condition or capability that must be met or possessed by a system or system component to satisfy a contract, standard, specification, or other formally imposed documents (IEEE Std 610.12 – 1990). A well-formed requirement is a statement of system functionality (a capability) that must be met or possessed by a system to satisfy a customer's need or to achieve a customer's objective, and that is qualified by measurable conditions and bounded by constraints (IEEE Std 1233-1998).

Requirements validation consists of activities that try to confirm that the behavior of a developed system meets its user needs. Requirements verification consists of those activities that try to confirm that the product of a system development process meets its technical specifications (Stevens et al., 2003).

Requirements engineering (RE) is concerned with the identification of the goals to be achieved by the envisioned system, the operationalization of such goals into services and constraints, and the assignment of responsibilities for the resulting requirements to agents such as humans, devices, and software. [4] The processes involved in RE include domain analysis, elicitation, specification, assessment, negotiation, documentation, and evolution. Getting high quality requirements is difficult and critical. Recent surveys have confirmed the growing recognition of RE as an area of utmost importance in software engineering research and practice.

Requirements engineering technique that does not fit the current project is selected, the project is doomed to fail. [5]In this paper, we propose a framework to characterize typical requirements engineering techniques and use it as a base for selecting appropriate techniques at the time of starting a project as well as at the time of recognizing a

^{1,2,3}UIET Deptt. Kurukshetra, Computer Deptt. Guru Jambheshwar University Hissar

Email: kavitadawra285@gmail.com¹, shikha.mtech28@gmail.com², sunilagodara@gmail.com³

change in the project nature or encountering an obstacle in defining a suitable set of requirements.

Requirements engineering is arguably the most important activity in the development of complex, software-intensive systems. Generally, the higher the complexity of the system under development, the more exacerbated the importance of good requirements engineering becomes. [17]

Tropos, a novel agent-oriented software engineering methodology, is heavily characterized, among other features, by the fact that it pays great attention to the activities that precede the specification of the prescriptive requirements, such as understanding how the intended system would meet the organizational goals. [3][6]The focus here is on the early requirements and on how to manage the transition from them to the late requirement analysis. But there are some risks in requirement engineering which was overcome In requirement engineering, a goal-oriented requirements engineering methodology aiming at modeling not only what and how aspect of requirements but also why, who, and when. [7]

3. COMPARE THE METHODOLOGIES

Case 1 Tropos is a software development methodology founded on the key concepts of agent-oriented software development. Specifically, Tropos emphasizes concepts for modeling and analysis during the early requirements phase. The TROPOS methodology addresses agent-oriented development and it is intended to support all analysis and design activities in the software development process.

One of the main advantages of the Tropos methodology is that it allows to capture not only what or the how, but also the why a piece of software is developed. The 'why' question helps to discover the objectives and rationale behind the goals which in fact identify the higher goals. It also helps to improve the quality. Tropos supports five phases of software development. [6] The early requirements analysis is concerned with the understanding of a problem by studying an existing organizational setting. During the late requirements analysis, the system-to-be is described within its operational environment, along with relevant functions and qualities. The architectural design phase deals with the definition of the system global architecture in terms of subsystems that are represented as actors, and their data dependencies, that are represented as actor dependencies. The detailed design phase aims at specifying each architectural component in further detail in terms of inputs, outputs, control and other relevant information. Finally, during the implementation phase, the actual implementation of the system is carried out, consistently with the detailed design. Tropos methodology itself, called, since now on, the Tropos tool.

Case 2 I* The i* framework [Yu95, Yu97] was developed for modeling and analyzing organizations to help support business process reengineering and requirements engineering. The framework focuses on modeling intentional and strategic relationships between actors.[12] The i* framework is a goal-oriented and agent-oriented language defined with the aim of modeling and reasoning about organizational environments and their Information Systems. The i* framework does not address the construction of the i* models as a main issue, but it provides some guidance on how to develop them based in constructing the SR(Strategic Rationale) models by using strategic reasoning in order to obtain also the SD (Strategic dependencies).

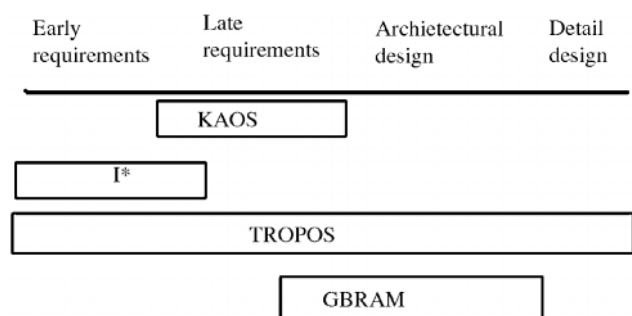


Fig. 1: Comparison of Tropos with other Software Development Methodologies

In the i* framework [Yu93, Yu97], various types of agent dependency links are defined to model situations where an agent depends on another for a goal to be achieved, a task to be achieved, or a resource to become available.

Case 3 Other Approaches

KAOS stand for Knowledge Acquisition in autOMated Specification. It is a focus on a goal-directed requirements acquisition task and is composed by three components: a conceptual model for acquiring and structuring requirements models, with an associated acquisition language; a set of acquisition strategies for elaborating requirements models in this framework; and an acquisition assistant to provide automated guidance in the acquisition process according to such strategies. [13] The categories established by KAOS to classify goals are: System Goals and Private Goals.

System Goals are application-specific goals that must be achieved by the composite system, and On the other hand, Private Goals are agent specific goals that might be achieved by the composite system.

GBRAM (Goal-Based Requirements Analysis Method) defines a top-down analysis method that refines goals and attributes them to agents starting from inputs such as corporate mission statements, policy statements, interview transcripts.[5][14] For doing so, it focus on the

transformation of enterprise and system goals into requirements, more specifically to assist analysts in gathering software and enterprise goals from many sources. The method's chief contribution was the provision of heuristics and procedural guidance for identifying and constructing goals.

So Tropos covers the software development process as a whole, from the early steps, in which the software engineer picks up and models requirements of the organizational setting (early requirements) and of the system-to-be (late requirements), up to the detailed design

4. NEW APPROACH

Research methodology provide the scientific way of solution for the given problem and analysis of the problem.

Tropos [Trop] is an agent-oriented software development methodology which is founded on the concepts of goal-based requirements adopted from the i* [I Star] and GRL (Goal-oriented Requirements Language) [GRL] [6]. Tropos-specific modeling deals primary with modeling of needs and intentional aspects of the agent system, from the early requirements analysis to the late design.

Requirements engineering started with the study of what the system should do, i.e. late-phase requirements analysis, which focuses on the specification of requirements, their completeness, consistency, automated verification, etc. Early-phase requirements analysis focuses on why the system must be developed, how the desired system will meet its goals

The framework tackles the modeling effort by breaking the activity down into more intellectually manageable components, and by adopting a combination of different approaches, on the basis of a common conceptual notation.

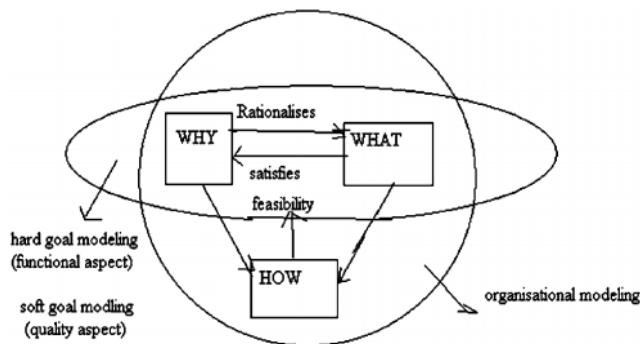


Fig. 2: Proposed Requirement Engineering Framework (REF)

According to the nature of a goal, a distinction is made between hard-goals and soft-goals. Soft-goals are used to specify, at a qualitative level, not sharply-cut objectives, the precise definition of which require to develop further details, while hard-goals clearly define a state/target, an actor desires

to reach. [10] For example, .having a passport delivered is clearly a hard-goal, while .having it delivered quickly is a soft-goal, being the notion of .quickly, highly subjective.

Soft goals play a central role, providing a systematic and organized way of handling non-functional requirements (or quality attributes, qualities, or more colloquially “-abilities”) à constraints in operational terms, or fit criteria for assessment purposes.

Organization Modeling during which the organizational context is analyzed and the agents and their goals identified. Any agent may generate its own goals, may operate to achieve goals on the behalf of some other agents, may decide to collaborate with or delegate to other agents for a specific goal, and might clash on some other ones. The resulting goals will then be refined, through interaction with the involved agents, by hard- and soft-goal modeling.

Hard-Goal Modeling seeks to determine how an agent can achieve a received hard goal, by decomposing it into more elementary subordinate hard-goals, tasks³, and resources⁴. Supported by the REF graphical notation, the analyst and the agent will work together to understand and formalize how the agent thinks to achieve the goal, in terms of subordinate hard-goals and tasks that he or she will have to achieve and perform directly, or indirectly, by passing them to other agents.

Soft-Goal Modeling aims at producing the operational definitions of the soft-goals, sufficient to capture and make explicit the semantics that are usually assigned implicitly by the involved agents. [2][11]

Improve the Quality

Requirements engineering (RE) is concerned with the identification of the goals to be achieved by the envisioned system, the operationalization of such goals into services and constraints, and the assignment of responsibilities for the resulting requirements to agents such as humans, devices, and software.

In software-intensive systems, the achievement of qualities—such as performance, availability, security, and modifiability—is dependent on the software architecture. Quality should be measured from the early stages of software building or else the development can end up with software that fulfills the requirements but fails to satisfy the customer. [15]

High Quality Requirement documents are a must for successful software projects. Requirements are the initial step in every project, and therefore it is necessary to collect good requirement as early as possible in the project life cycle. [7][16] All process phases directly and indirectly depend on the requirement document. As Tropos methodology is that it allows to capture not only what or

the how, but also the why a piece of software is developed. The 'why' question helps to discover the objectives and rationale behind the goals which in fact identify the higher goals. It also helps to improve the quality.

5. CONCLUSION

In this paper, we presented the overview of requirement engineering and introduced an agent-oriented Requirements Engineering Framework (REF), explicitly designed to support the analysts in transforming high-level organizational needs into system requirements, while redesigning the organizational structure itself. Agent-based Requirements Engineering Framework designed around the adoption of a simple, but effective, representational graphical notation. The purposed framework based on the methodology named Tropos, seems to complement nicely current proposals for agent-oriented programming platforms. This new approach helps to analyze the system that gives a good impact on quality. Tropos methodology focused on its application in developing specific classes of applications, as for instance distributed knowledge management systems.

REFERENCES

- [1] D. Bertolini, A. Novikau, A. Susi, and A. Perini, "TAOM4E: an Eclipse Ready Tool for Agent-Oriented Modeling", Issue on the Development Process. Technical Report, Fondazione Bruno Kessler – iris pp.45-48, 2006.
- [2] M.-P. G. Federico Bergenti and F. Zambonelli, Editors, "Methodologies and Software Engineering for Agent Systems: The Agent-Oriented Software Engineering Handbook", Springer, 2004, pg 121-130.
- [3] A. I. Anton, "Goal-based Requirements Analysis", In Proceedings of the IEEE International Conference on Requirements Engineering (ICRE '96), Colorado Springs, USA, Apr. 2006.
- [4] Axel van Lamsweerde, "Requirements Engineering in the Year 00: A Research Perspective", 0-7695-2579-2/00 IEEE.
- [5] Hofmann, H. F. and Lehner, F. : "Requirements Engineering as a Success Factor in Software Projects", 18, No. 4, pp.58-66/01 IEEE.
- [6] P. Bresciani, P. Giorgini, F. Giunchiglia, J. Mylopoulos, and A. Perini, "Tropos: An Agent-Oriented Software Development Methodology", 8(3), 203–236, July 2004.
- [7] Paolo Bresciani¹ and Paolo Donzelli², "REF: a Practical Agent-Based Requirement Engineering Framework", 978-0-7695-3316-2/05 IEEE.
- [8] P. Bresciani and F. Sannicol_o, "Applying Tropos Requirements Analysis for defining a Tropos Tool", In P. Giorgini, Y. Lesperance, G.Wagner, and E. Yu, Editors, Agent-Oriented Information System. Proceedings of AOIS-2002: Fourth International Bi-Conference Workshop, Pages 135.138, Toronto, Canada, May 2002.
- [9] P. Bresciani, P. Giorgini, F. Giunchiglia, J. Mylopoulos, and A. Perini, "TROPOS: An Agent-Oriented Software Development Methodology", Submitted to the Journal of Autonomous Agents and Multi-Agent Systems. Kluwer Academic Publishers, Page 23-29 March 2002.
- [10] A. van Lamsweerde, "Goal-oriented Requirements Engineering: A Guided Tour", In Proceedings of RE'01 - International Joint Conference on Requirements Engineering, pages 249–263, Toronto, Aug 2001, IEEE.
- [11] [GJKL01] Gans, G., Jarke, M., Kethers, S., and Lakemeyer, G., "Modeling the Impact of Agent Networks", to Appear in Proc. of the 3rd International Bi-Conference Workshop on Agent-Oriented Information Systems (AOIS-2001), Interlaken, Switzerland, June, 2001.
- [12] Anwer, S., Jinnah, M.A., Ikram, N.: "Goal Oriented Requirement Engineering: A Critical Study of Techniques". In Proceedings of the 13th Asia Pacific Software Engineering Conference, APSEC 2006.
- [13] Decker, B. et al. "Wiki-based Stakeholder Participation in Requirements Engineering", IEEE Software, March/April 2007.
- [14] [FPMT01] Fuxman, A., Pistore, A., Mylopoulos, J., Traverso, P., "Model Checking Early Requirements Specifications in Tropos", Submitted to the Fifth IEEE International Symposium on Requirements Engineering (RE01), August, 2001, Toronto, Canada, available at <http://www.cs.toronto.edu/km/tropos/>.
- [15] Barbacci, M.; Klein, M.; & Weinstock, C., "Principles for Evaluating the Quality Attributes of a Software Architecture (CMU/SEI- 96-TR-36, ADA324233)", Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2006.
- [16] Boehm, B. W., "Characteristics of Software Quality", New York, NY: American Elsevier, 2008.
- [17] Shahzad Anwer, Naveed Ikram, "Goal Oriented Requirement Engineering: A Critical Study of Techniques", 0-8186-4300 5/06 IEEE.