Goal Oriented Requirements Engineering - A Review

Sultan Aljahdali, Jameela Bano, Nisar Hundewale

College of Computers and Information Technology
Taif University, Taif, Saudi Arabia
aljahdali@tu.edu.sa, jbano2007@gmail.com, nisar@computer.org

Abstract

Requirements engineering is concerned with producing a set of specifications for software systems that satisfy their stakeholders; and can be implemented, deployed and maintained using these alternatives. The last fifteen years have seen the rise of a new phase in software development which is concerned with the acquisition, modeling and analysis of stakeholder purposes ("goals") in order to derive functional and non-functional requirements. Goals play a prominent role in the requirement engineering process; they drive the elaboration of requirements to support them. Goals are intended outcomes to be achieved by the system under consideration. The concept of "Goals" is increasingly being used in requirement engineering. Goal oriented requirements engineering refers to the use of goals for requirements elicitation, elaboration, organization, specification, analysis, negotiation, documentation and evolution.

The objective of this paper is to provide a brief and comprehensive review of the major efforts undertaken along this line of research. In this paper we have done literature review of the concepts, terminology, significance and techniques of Goal oriented requirements engineering.

Keywords: Requirements engineering, Goals, Goal oriented requirements engineering,Security requirements,Scalability requirements,Risk management.

1. INTRODUCTION

Requirement engineering (RE) is the very first step of the system development process. It is concerned with the identification of stakeholders' goals about the intended system; the specification of services and constraints that operationalize those goals; and the assignment of responsibilities for the resulting requirements to agents such as humans, devices and software [1]. Requirement engineering is concerned with requirement elicitation, analysis, specification, validation and requirement management [2], [3], [4], [5], [6], [7]. Requirements engineering research has increasingly recognized the leading role played by goals in the RE process [8], [9], [10], [11], [12], [13], [14].

RE has been developed as a discipline to identify and then translate stakeholder needs into system requirements [14]. Traditional systems analysis concentrates on what features a system will support, but the Goal Oriented Requirements Engineering (GORE) approach, is based on the identification of system goals and the transformation of those goals into requirements; it addresses concerns about why a certain goal is required, how it can be achieved and who is responsible for it in the system and/or the environment [4], [15], [1]. The notion of concentrating on the why is not new; organizing requirements around goals is relatively new [16]. During goal-oriented analysis, we start with initial stakeholder goals such as "Fulfill every book request", or "Schedule meeting" and keep refining them until we have reduced them to alternative collections of functional requirements each of which can satisfy the initial goals [17].

Goals play a prominent role in the RE process; they drive the elaboration of requirements to support them [18], [11], [19]; they provide a completeness criterion for the requirements specification—the specification is complete if all stated goals are met by the specification [8]. Goals are intended outcomes to be achieved by the system under consideration [22]. The concept of "Goals" is increasingly being used in requirement engineering [19]. They provide a rationale for requirements—a requirement exists because of some underlying goal which provides a base for it [20], [21]. Goal oriented requirements engineering refers to the use of goals for requirements elicitation, elaboration, organization, specification, analysis, negotiation, documentation; and evolution [23].

The objective of this paper is to provide a brief and comprehensive review of the major efforts undertaken along this line of research. Section 2 defines the basic concepts and terminology of GORE. Section 3 differentiates traditional versus goal based approach. Section 4 explains about the benefits of GORE, its advantages and disadvantages. Section 5 specifies different GORE techniques and summarizes them briefly. Section 6 specifies future scope and the paper concludes with a brief summary.

2. CONCEPTS AND TERMINOLOGY

Dardenne [22] defines requirements in terms of goals and propose a goal directed acquisition technique for requirements elicitation. Lamsweerde [23] presents significance of goals and classifies the goals into three types; the goals have been defined in terms of GORE concepts. Anton [24] gives a technique for goal identification and refinement for a software based information systems based on GORE concepts. GORE techniques [25], [26] for identification, elicitation, and idea generation of goals have been proposed based on GORE concepts. Regev and Wegman [27] define the underlying principles of GORE from the principles of General System Thinking and Cybernetics. Naveed and Anawar [35] give a critical study of the GORE techniques for evaluation and common unifying framework. Lapouchnian [29] presents concepts of goals, beliefs, constraints, taxonomies, agents, requirements and assumptions.

2.1 Requirements Engineering

Requirements engineering is the branch of software engineering concerned with the real-world goals for, functions of, and constraints on software systems. It is also concerned with the relationship of these factors to precise specifications of software behavior, and to their evolution over time and across software families [1, 40]. RE is now defined by the RE community as goal-driven.

2.1.1 Requirements Engineering Activities

The following intertwined activities that are covered by requirements engineering [1], [30].

- **Domain analysis**: the environment for the system-to-be is studied. The relevant stakeholders are identified and interviewed. Problems with the current system are discovered and opportunities for improvement are investigated. Objectives for the target system are identified.
- **Elicitation:** alternative models for the target system are analyzed to meet the identified objectives. Requirements and assumptions on components of such models are identified. Scenarios could be involved to help in the elicitation process.
- **Negotiation and agreement**: alternative requirements and assumptions are evaluated; risks are analyzed by the stakeholders; the best alternatives are selected.
- **Specification:** requirements and assumptions are formulated precisely.
- **Specification analysis:** the specifications are checked for problems such as incompleteness, inconsistency, etc. and for feasibility.
- **Documentation:** various decisions made during the requirements engineering process are documented together with the underlying rationale and assumptions.
- **Evolution:** requirements are modified to accommodate corrections, environmental changes, or new objectives.

A considerable knowledge is built up during early-phase RE. This knowledge is used to support reasoning about organizational objectives, alternatives and implications, etc. It is important to retain and maintain this knowledge in order to guide system development, evolution to have a deep understanding about a domain, one need to understand the interests, priorities and abilities of various stakeholders. Hence early phase RE models that deal with organizational goals and stakeholder interests cut across multiple systems. These can reason about of the cooperation among systems. Hence the emphasis in Goal-Oriented RE is on understanding the "whys".

2.2 Limitations of Traditional RE approaches

Traditional requirements engineering research takes start from the initial requirements statements, which express customer's wishes about "**what**" the system should do. The objective, of traditional requirements engineering tasks, is to produce a requirements document to pass on to the developers. The purpose is that the resulting system should be adequately specified and constrained, often in a contractual setting. It ignores to focus on "**why**" the system should do, which is the focus of Goal-Oriented RE.

2.3 Definition of Goals

Van Lamsweerde [1] defines a goal as an objective that the system should achieve through cooperation of agents in the software-to-be and in the environment.
Anton [31] states that goals are high-level objectives of the business, organization or system; they capture the reasons why a system is needed and guide decisions at various levels within the enterprise.

Dardenne [25]) defines the concept of goal as "A non-operational objective to be achieved by the composite system." Oshiro [32] suggests a technique for generation of sub goals from goals through a refinement process of meeting involving all the stakeholders. The process continues until it becomes an objective that can be satisfied through the process of programming.

The focus on goals can be understood by examining the following quote from the seminal KAOS paper [25] Goals are important in several respects. They lead to the incorporation of requirements components which should support them. They justify and explain the presence of requirements components which are not necessarily comprehensible to clients. They may be used to assign the respective responsibilities of agents in the system; more precisely, they may provide the basis for defining which agents should best perform which actions to fit prescribed constraints (according to their capabilities, reliability, cost, load, motivation, and so forth). Finally, they provide basic information for detecting and resolving conflicts that arise from multiple viewpoints among human agents [9].

2.4 Definition of Agents

Agents are active components of the system such as human, devices, and software [22], [43]. Agents are responsible for the fulfillment of goals. Agents are active components in system or its environment. [42], [37].

Goals are to be achieved by the cooperation of various agents. Such agents may include software components that exist or are to be developed, external devices, and humans in the environment. The system being considered in the requirements engineering process is thus composite [32];

it includes both the software-to-be and its environment.

3. TRADITIONAL ANALYSIS VERSUS GORE

Goal-driven requirements engineering takes the view that requirements should initially focus on the why and how questions rather than on the question of what needs to be implemented. "Traditional" analysis and design methods focused on the the functionality of the system to be built and its interactions with users. Instead of asking what the system needs to do, goal-driven methods ask why a certain functionality is need and how it can be implemented. Thus goal-driven methods give a rationale for system functionality by answering why a certain functionality is needed while also tracking different implementation alternatives and the criteria for the selection among these alternatives [33].

Goal-Oriented RE is “Early-phase RE” activities vis-à-vis traditional RE that is late phase RE. Hence Goal-Oriented RE supports the activities performed before the formulation of the initial requirements.

4. BENEFITS OF GOAL MODELING

There are a number of important benefits associated with explicit modeling, refinement, and analysis of goals.

- It is important to note that GORE takes a wider system engineering perspective compared to the traditional RE methods: goals are prescriptive assertions that should hold in the system made of the software-to-be *and its environment*; domain properties and expectations about the environment are explicitly captured during the requirements elaboration process, in addition to the usual software requirements specifications. Also, goals provide rationale for requirements that operationalize them. Thus, one of the main benefits of goal-oriented requirements engineering is the added support for the early requirements analysis [23].
- Goals provide a precise criterion for *sufficient completeness* of a requirements specification. The specification is complete with respect to a set of goals if all the goals can be proven to be achieved from the specification and the properties known about the domain [8].
- Goals provide a precise criterion for requirements *pertinence*. A requirement is pertinent with respect to a set of goals in the domain if its specification is used in the proof of one goal at least [8]. Even without the use of formal analysis methods, one can easily see with the help of goal models whether a particular goal in fact contributes to some high-level stakeholder goal[23].
- A goal refinement tree provides traceability links from high-level strategic objectives to low-level technical requirements [23].
- Goal modeling provides a natural mechanism for structuring complex requirements documents [23].
- One of the concerns of RE is the management of conflicts among multiple viewpoints [4]. Goals can be used to provide the basis for the detection and management of conflicts among requirements [34] [35].
- A single goal model can capture variability in the problem domain through the use of alternative goal refinements and alternative assignment of responsibilities. Quantitative and qualitative analysis of these alternatives is possible [23].
- Goal models provide an excellent way to communicate requirements to customers. Goal refinements offer the right level of abstraction to involve decision makers for validating choices being made among alternatives and for suggesting other alternatives [23].
- Separating stable from volatile information is also important in requirements engineering. A number of researches point out that goals are much more stable than lower-level concepts like requirements or operations [31], [23]. A requirement represents one particular way of achieving some goal. Thus, the requirement is more likely to evolve towards a different way of achieving that same goal than the goal itself. In general, the higher level the goal is the more stable it is.

4.1 Advantages of GORE

The main advantages of goal-oriented RE [23], [36] are as follows:

a. From goals one can systematically derive requirement and object models.
b. Goals give the rationale for requirements.
c. A goal graph can provide traceability from strategic concerns to technical details.
d. Goal formalization can prove if the refinements are correct and complete.
e. The goal refinement structure can indicate a comprehensible structure that is helpful in the requirements document.
f. Alternative system proposals also could be explored with the help of alternative goal refinements.

4.2 Disadvantages of GORE

The main disadvantages of goal-oriented RE [36] are as follows:

a. Unless a rigorous automated reason is used with formal methods, an abstract model may go unquestioned.
b. Records contain vague intentions without thinking properly about the practical applications.

5. GORE TECHNIQUES

A variety of techniques have been proposed e.g. Deriving Tabular Event-Based Specifications from goal oriented requirement model (DTEBS) [37] GBRAM [38], AGORA [39], Visual Variability Analysis for goal models

(VVA) [40], Goal-Oriented Idea Generation Method (GOIG) [41], Deriving Operational Software Specifications (DOSS)[42], Agent-Based Tactics for goal-oriented requirements elaboration (A-BT) [22], and goal oriented requirement elicitation based on General System Thinking Heuristics (GSTH) [43]. Duboc [44] describes application of GORE for eliciting the scalability requirements of a large, real-world financial fraud detection system. Duboc [44] presents a case study that reveals both the suitability and the limitations of GORE as a technique for eliciting the information needed by stakeholders to specify scalability goals of a system.Requirements engineering community has accepted the need and importance of security from the early stages of the software systems development which has been presented in the publications[45,46,47,48] to focus on that area.The most advanced ideas in the information system security risk management (ISSRM) are reported in the new ISO/IEC 2700× series [49].

This section presents a survey of GORE techniques, their process and methodology. Table 1 summarizes the GORE techniques with respect to the coverage of different requirement engineering activities [16].

The following presents a brief overview of GORE Techniques [16].

GSTH [43] deals with requirement elicitation. It defines the highest level goals (a new level of abstraction) and proposes a set of heuristics based on General System Thinking (GST) and Cybernetics.

DOSS [42] deals with the Requirement Specification activity. It defines formal semantics for goals operationalization based on pre, post and trigger conditions; agents and their realization of goals and goal operations performed by agents; also defines taxonomy of goal patterns.

DTEBS [37] uses the same models proposed by [42] for deriving tabular event-based specifications.

GBRAM [38] deals with Requirement Analysis activity. Set of heuristics (25 total) are proposed in GBRAM, 6 are related to Classification, 8 related to Refinement, 12 heuristics helps in Elaboration. In addition elaboration is supported by scenarios.

GOIG [41] is concerned with requirement elicitation. A process is defined for requirement elicitation based on idea-generation. Ideas are grouped into goals, and it mainly uses heuristics for idea-generation based elicitation.

A-BT [22] mainly proposes tactics for resolving problems of un-realization of goals by agents. Goals are assigned to agents and agents realize goals. A goal is unrealized by an agent when agent cannot observer monitored variables or cannot control controlled variables. No formal process is as such defined.

AGORA [39] Strengthens to support selecting goals to decomposed, prioritizing, conflicts resolution, and quality estimation. It works by attaching attribute values (-10 to 10) to nodes and edges in the AND-OR goal graph. The values express how many degrees the sub-goal contributes to the achievement of its parent goal. Different score is given in each edge in OR and same value is assigned to all the edges in AND decomposition. It uses preference matrix to find conflicts and gaps of understanding amongst different stakeholders (Customer, Developer, and Administrator).

VVA [40] deals with analysis and provide comprehensive reports for variability of requirements in order to achieve the satisfaction of stakeholders.

Table 1: GORE Techniques w.r.t. RE Coverage [16]

Elicitation	Domain Analysis	GSTH
	Requirement and Assumptions identification	GSTH GOIG
Analysis	Classifying	GBRAM A-BT AGORA VVA
	Modeling	GBRAM A-BT AGORA VVA
	Elaboration	GBRAM A-BT AGORA VVA
	Conflict Identification and Resolution	A-BT AGORA VVA
	Prioritization	AGORA VVA
Specification		DOSS DTEBS
Requirement Management	Requirement Change/Evolution Management	AGORA
	Traceability	AGORA GBRAM A-BT DTEBS DOSS VVA
	Conflict Management	AGORA
	Measurement	AGORA

6. FUTURE SCOPE

We hope this paper motivates the reader to do research in GORE. The various emerging research directions based upon GORE are the security requirements, scalability requirements (software quality) and risk management.

7. CONCLUSIONS

In this paper, the various efforts of GORE research has been summarized briefly by specifying the importance of requirements engineering, Goal oriented requirements engineering and GORE methods.

REFERENCES

[1] A. van Lamsweerde, "Requirements Engineering in the Year 00: A Research Perspective", Keynote Paper for ICSE’2000 - 22nd International Conference on Software Engineering, Limerick, ACM Press, 2000.

[2] Axel van Lamsweerde, “Goal-Oriented Requirements Engineering: A Roundtrip from Research to Practice”. 12th IEEE International Requirements Engineering Conference (RE’04), Kyoto, Japan, 2004.

[3] Alexei Lapouchnian “Goal-Oriented Requirement Engineering: An Overview of the Current Research”, University of Toronto, 2005.

[4] Kavakli, E. “Goal-Oriented Requirements Engineering: A Unifying Framework.” Requirements Engineering, January 2002, Vol. 6 No. 4. pp. 237-251.

[5] I. Sommerville, *Software Engineering 7th Edition,* Addison-Wesley, 2004.

[6] Bashar Nuseibeh & Steve Easterbrook "Requirements Engineering: A Roadmap", 2000

[7] Klaus Pohl "The Three Dimensions of Requirements Engineering"

[8] K. Yue, “What Does It Mean to Say that a Specification is Complete?”, *Proc. IWSSD-4, Fourth International Workshop on Software Specification and Design*, Monterey, 1987.

[9] Robinson, W.N., “Integrating Multiple Specifications Using Domain Goals”, *Proc. IWSSD-5 - 5th Intl. Workshop on Software Specification and Design,* IEEE, 1989, 219-225.

[10] V. Berzins and Luqi, *Software Engineering with Abstractions*. Addison-Wesley, 1991.

[11] A. Dardenne, S. Fickas and A. van Lamsweerde, “Goal-Directed Concept Acquisition in Requirements Elicitation”, *Proc. IWSSD-6 - 6thIntl. Workshop on Software Specification and Design*, Como, 1991, 14-21.

[12] Mylopoulos, J., Chung, L., Nixon, B., “Representing and Using Nonfunctional Requirements: A Process-Oriented Approach”, *IEEE Trans. on Software. Engineering*, Vol. 18 No. 6, June 1992, pp. 483-497.

[13] M. Jarke and K. Pohl, “Vision-Driven Requirements Engineering”, *Proc. IFIP WG8.1 Working Conference on Information System Development Process*, North Holland, 1993, 3-22

[14] P. Zave, “Classification of Research Efforts in Requirements Engineering”, *ACM Computing Surveys*, Vol. 29 No. 4, 1997, 315-321.

[15] J. Mylopoulos, "Information Modeling in the Time of the Revolution".

[16] Shahzad Anwer, Naveed Ikram, “Goal Oriented Requirement Engineering: A Critical Study of Techniques” XIII Asia pacific software engineering conference (APSEC'06) IEEE.

[17] John Mylopoulos, key note talk on “Goal-Oriented Requirements Engineering”, 14th IEEE Requirements Engineering Conference Minneapolis, September 15, 2006

[18] D.T. Ross and K.E. Schoman, “Structured Analysis for Requirements Definition,” *IEEE Trans. Software Eng.*, vol. 3, no. 1, pp. 6-15,1977.

[19] K.S. Rubin and A. Goldberg, “Object Behavior Analysis,” *Comm.ACM*, vol. 35, no. 9, pp. 48-62, Sept. 1992.

[20] Emmanuel Letier and Axel van Lamsweerde “Agent-Based Tactics for Goal-Oriented Requirements Elaboration”, 24th International Conference on Sofware Engineering, ACM Press, May 2002

[21] I. Sommerville and P. Sawyer, *Requirements Engineering: A GoodPractice Guide*. Wiley, 1997

[22] Dardenne, A.; Lamsweerde, A.V.; Ficas, S. (1993). Goal Directed Acquisition. Science of Computer Programming. 20:(1-2)

[23] A. van Lamsweerde. Goal-Oriented Requirements Engineering: A Guided Tour. Proc.5th IEEE International Symposium on Requirements Engineering (RE'01), Toronto, Canada, August 2001.

[24] Anton, A. I. (1997). Goal Identification and Refinement in the Specification of Software based Information Systems. PhD Dissertation. Georgia Institute of Technology. Atlanta GA.

[25] Kaiya, H., Horai, H. and Saeki, M. (2002). AGORA: Attribute Goal-Oriented Requirements Analysis Method. Proceedings. IEEE Joint International Conference on Requirements Engineering (RE’02), University of Essen, Germany.

[26] Oshiro, K., Watahiki, K.; Saeki, M. (2003).Goal-Oriented Idea generation Method for requirement elicitation.Proceedings. 11th IEEE International requirements Engineering Conference, California.

[27] Regev, G.; Wegmann, A. (2005). Where do Goals Come from: the Underlying Principles of Goal-Oriented Requirements Engineering. Proceedings. 13th IEEE International Requirements Engineering Conference,Paris, France, August.

[28] Naveed, S.; Anawar, I. (2006).Goal Oriented Requirement Engineering: A Critical Study of Techniques. Proceedings. 13th Asia Pacific Software Engineering Conference, Bangalore, India.

[29] Lapouchnian, A. (2005). Goal-Oriented Requirements Engineering : An Overview of the current Research, University of Toronto. USA, 1987.

[30] B. Nuseibeh, S. Easterbrook. Requirements Engineering: A Roadmap. Proc. Conference on the Future of Software Engineering, Limerick, Ireland, June 2000.

[31] A. Anton, W. McCracken, C. Potts. Goal Decomposition and Scenario Analysis in Business Process Reengineering. Proc. 6th Conference On Advanced Information Systems Engineering (CAiSE’94), Utrecht, Holland, June 1994.

[32] M. Feather, “Language Support for the Specification and Development of Composite Systems”, *ACM Trans. on Programming Languages and Systems*9(2), Apr. 87, 198-234.

[33] http://lams.epfl.ch/reference/goal, : LAMS

[34] A. van Lamsweerde. Divergent Views in Goal-Driven Requirements Engineering. Proc.Workshop on Viewpoints in Software Development, San Francisco, USA, October,1996

[35] W. Robinson. Integrating Multiple Specifications Using Domain Goals. Proc. 5th International Workshop on Software Specification and Design (IWSSD-5), Pittsburgh,USA, May 1989.

[36] Naeem Ur Rehman,Sarfraz Bibi,Sohail Asghar,Simon Fong “Comparative Study of Goal-Oriented Requirements Engineering”,

[37] De Landtsheer R., Letier E. & van Laamsweerde A., “Deriving Tabular Event-Based Specifications from Goal-

Oriented Requirements Models", Proceedings 11th IEEE International Conference on Requirements Engineering, California, IEEE Computer Society Press, 200-210, 2003.
[38] A.I. Anton and C. Potts, "The Use of Goals to Surface Requirements for Evolving Systems", 20th International Conference on Software Engineering, Kyoto, April 1998.
[39] Haruhiko Kaiya, Hisayuki Horai, Motoshi Saeki, "AGORA: Attributed Goal-Oriented Requirements Analysis Method". Proceedings of the IEEE Joint International Conference on Requirements Engineering (RE'02), University of Essen, Germany, 2002.
[40] Bruno Gonzalez-Baixauli, Julio Cesar Sampaio do Prado Leite, John Mylopoulos, "Visual Variability Analysis for Goal Models", 12th IEEE International Requirements Engineering Conference (RE'04), Kyoto, Japan, 2004, *re*, pp. 198-207.
[41] Kazuya Oshiro, Kenji Watahiki, Motoshi Saeki, "Goal-Oriented Idea Generation Method for Requirements Elicitation". Proceedings. 11th IEEE International Requirements Engineering Conference, California, 2003.
[42] Emmanuel Letier and Axel van Lamsweerde, "Deriving Operational Software Specifications from System Goals" SIGSOFT 2002/FSE-10, ACM Press, Charleston, SC, USA, 2002.
[43] Regev, G. and Wegmann, A., "Where do Goals Come from: the Underlying Principles of Goal-Oriented Requirements Engineering", 13th IEEE International Requirements Engineering Conference, Paris, France, August, 2005.
[44] Leticia Duboc, Emmanuel Letier, David S. Rosenblum, Tony Wicks "A Case Study in Eliciting Scalability Requirements", 16th IEEE International Requirements Engineering Conference, 2008 pp 247-252.
[45] Anton AI (2002) Requirements engineering for information security, guest editorial. Requir Eng J 7(4):177–178
[46] Devanbu P, Stubblebine S (2000) Software engineering for security: a roadmap, In: Proceedings of the 22nd International Conference on Software Engineering (ICSE '00), Future of Software Engineering Track, pp 227–239, ACM
[47] McDaniel P, Nuseibeh B (2008) Guest editors' introduction: special section on software engineering for secure systems. IEEE Trans Softw Eng 34(1): 3–4
[48] Mouratidis H, Giorgini P (eds) (2006) Integrating security and software engineering: advances and future vision. Idea group, IGI Publishing Groups.
[49] Eric dubois,Haralambos Mouratidis,"Guest editorial: security requirements engineering:past,present and future",Springer Journal: Requirements Eng (2010)15:1-5.