International Journal of Scientific Research in Multidisciplinary Studies
Vol.5, Issue.10, pp.39-44, October (2019)
Research Paper
E-ISSN: 2454-9312
P-ISSN: 2454-6143

A Goal Oriented Requirement Engineering Research Structure

Shaik Subhani[1*], K.P. Suryateja[2]

[1, 2]Dept. of Information Technology, Sreenidhi Institute of Science and Technology (Autonomous), JNTUH, Hyderabad, India

**Corresponding Author: shaiksubhani@sreenidhi.edu.in*

Available online at: www.isroset.org

Received: 10/Oct/2019, Accepted: 18/Oct/2019, Online: 31/Oct/2019

Abstract-Requirements engineering is a sub regulation of software engineering and critical phase in the overall software engineering life cycle. A number of improvements have been made to process the requirements engineering process. Most of them attempted to bond these requirements to goals. In this paper, we present major research works done in Goal-Oriented Requirement Engineering. It offers an incremental approach for elicitation, analysis, elaboration & refinement, specification and modeling of requirements. The assets of GORE claimed in the literature are presented. Several goal-oriented methods have been proposed and a comparative study is made, which handle as a pilot for readers to choose a suitable goal-oriented technique to accomplish the requirements engineering needs.

***Keywords**:* Requirements, GORE, Elicitation, Software Product, Goals, Refinement.

I. INTRODUCTION

The quality of a software system critically depends on the degree to which fulfills its necessities. Such requirements can be bring out, modeled, and analyzed as stakeholder goals. The domain of goal-oriented requirements engineering has emerged in order to create and study different methods which RE from a goal-oriented perception. Goals are intended output to be achieved by the system under discussion. The concept of Goals is gradually more being used in requirement engineering process. Over the last two decades, much concentration has been paid to the field of goal-oriented requirements engineering, where goals are used as a useful conceptualization to model, analyze requirements, conflicts, elicit, and capturing alternatives. Goal models have been used as an efficient for capturing the communications and tradeoffs between necessities, but they have been practically more broadly to move forward the state of software adaption, legal compliance, business intelligence and security among other domains. In this paper, we discuss important research works done in Goal-Oriented Requirement Engineering.

GORE is treated as main attainment that the essential of the Requirement Engineering area has generated since its inception. Several GORE methods were designed in the past twenty years in both research and industry. GORE methods seem to have emerged out of now here in the early 1990s, the idea of Goal appearing as natural elements in human and organizational behavior. A variety of method have been proposed e.g. Deriving Operational Software Specifications, Deriving Tabular Event-Based Specifications from goal oriented requirement model, Goal-Based Requirements Analysis Method, AGORA, Visual Variability Analysis for goal models, goal-oriented requirement elicitation based on General System Thinking Heuristics, Goal-Oriented Idea Generation Method and Agent-Based Tactics for goal-oriented requirements elaboration. These methods define software requirements from organizational objectives, and provide a framework for relating organizational goals and problems in the project formulation for making decision. The following figure 1 shows overview of GORE frame work

Analyzing the interactions among nonfunctional requirements: four types of communications among nonfunctional requirements are recognized, it may be irrelevant, conflicting, cooperative, or counterbalanced. Structuring fuzzy object-oriented models based on the interactions analyzed: goals are arranged into different alternatives based on the interaction analyzed to form a goals ladder and a steady kernel is constructed to handle as a basis for further modification in an incremental fashion. Various methods are also projected for resolving conflicts between goals into numerous alternatives based on the interactions analyzed to form goals levels. A stable kernel is constructed to use as a basis for further refinement in an incremental fashion. Different techniques are also proposed for solving conflicts between goals.

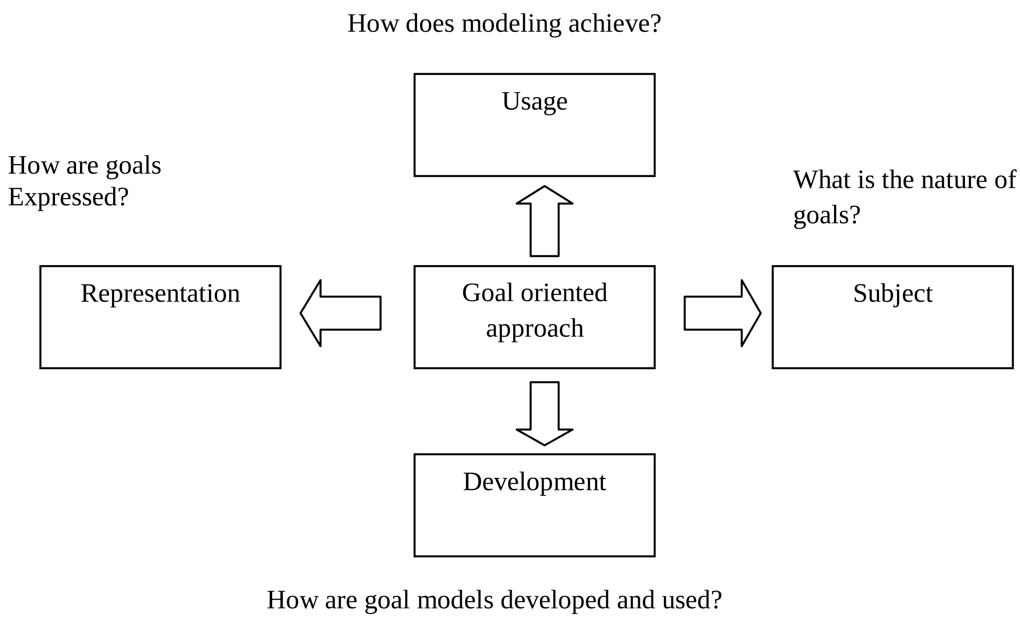

Figure 1. A Framework for understanding goal-oriented approach

The rest part of this paper is organized as follows. Section 2 states background view Goal concepts. In section, 3 we define GORE Framework. Gore methods for soft goal elicitation are discussed in Section 4. We also present a conclusion and future scope of extension of our work in Section 5.

II. RELATED WORK

Much more work is done for requirement engineering. Duboc et.al discusses application of GORE for eliciting the scalability requirements of a huge, real-world financial fraud finding system. Duboc states a case study that acknowledge both the appropriateness and the drawbacks of GORE as a technique for eliciting the data needed by stakeholders to indicate scalability goals of a system. Later, a number of researchers have reported the progress toward the improvement of goal-based methods. Dardenne et al. have suggested a goal-directed procedure to models acquisition. Mylopoulos et al. have projected a framework for presenting nonfunctional requirements in terms of goals, which can be evaluated in order to decide the degree to which a nonfunctional requirement is backed by a specific design. Moreover, they suggested that object-oriented modeling process can be used to model functional requirements to reimburse the goal-oriented procedure. Meanwhile, Anton has proposed a goal-based requirement analysis method to recognize, elaborate and filter goals for requirements specifications. GREMSoC et.al methodology is to encourage a process to enhance reusability, maintainability and comprehensibility of requirements specification by means of separation of specific principle. But the authentication of catalogs to discover and specify concerns is time taking process. AGORA et al. goal graph technique offered requirement with the goal graph but it does not give the methodology for decomposing the goals into sub goals. So the clarification of the goals and the detailed requirement gathering of each goal are limited. They didn't consider that goal elicitation should be the collaborative job done by a team of stakeholders who have knowledge of dissimilar fields.

III.THE GOAL MODEL

A goal defines an objective the composite system should meet usually through the cooperation of multiple agents. For example, a goal in a meeting scheduling problem would be that each requested meeting is eventually held with the presence of all intended participants. This ideal goal might be captured by the following specification fragment.

GoaAchieve [ConvenientMeetingHeld]

Definition each requested meeting is eventually being held with the presence of all intended participants.

FormalDef ∀ m: Meeting: m.Requeste ⇒ ◊ m.Holds∧ (∀ p: Participant): Intended (p, m) → Participates (p, m)

Each goal has a name, a natural language definition, and an optional formal definition. The above goal is named Achieve [ConvenientMeetingHeld]. A goal defines a set of admissible histories in the composite system. Intuitively, an history is a temporal sequence of states of the system. Each goal is satisfied by some histories and falsified by some other histories. The notation h |= G is used to express that history h satisfies the goal G. The definition of a goal is a natural language statement describing the set of histories satisfying the goal. The formal definition of a goal is a temporal logic formula describing the same set of histories. Goal taxonomy is used to guide the acquisition and definition of goals. Goals are classified according to their pattern and category. The pattern of a goal is based on the temporal behaviour required by the goal. The KAOS language distinguishes the following four goal patterns: Achieve goals- goals need that a little property eventually holds. Cease goals- goals needs that a few property eventually stops to hold. Maintain goals- goals needs that a number of properties always hold. Avoid goals- goals needs that a little property never holds goal patterns offer a lightweight way of declaring the temporal behaviour of a goal without writing formal goal definitions. Goal patterns and equivalent temporal formula templates include the following:

Achieve: $P \Rightarrow \Diamond Q$, $P \Rightarrow \Diamond_{\leq d} Q$, $P \Rightarrow \bigcirc Q$ Cease: $P \Rightarrow \Diamond \neg Q$, $P \Rightarrow \Diamond_{\leq d} \neg Q$, $P \Rightarrow \bigcirc \neg Q$

Maintain: $P \Rightarrow Q$, $P \Rightarrow \Box Q$, $P \Rightarrow Q \; W \; R$

Avoid: $P \Rightarrow \neg Q$, $P \Rightarrow \Box \neg Q$, $P \Rightarrow \neg Q \; W \; R$

IV.GORE FRAMEWORK

Requirements engineering is the part of software engineering with the real-world goals for, functions and constraints on software systems. It is also link with these factors to accurate specifications of software behavior, and to their development over time and across software families. RE is now defined by the RE community as goal-driven. The following tangled activities (as shown in Figure 2) that are covered by requirements engineering:

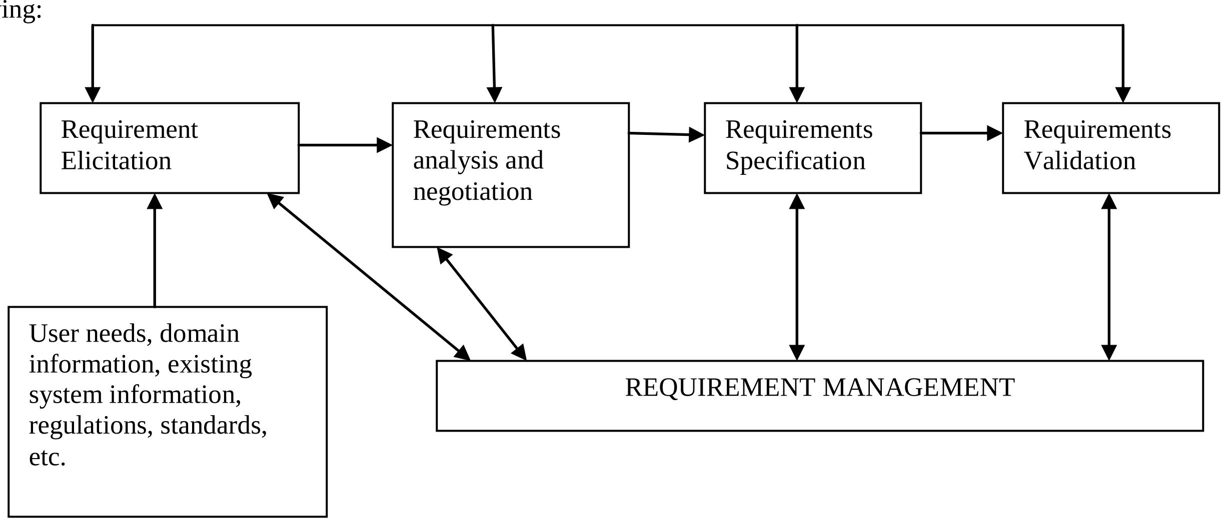

Figure 2. Requirements engineering activities

Domain analysis: the environment for the system to be is deliberate. The relevant stakeholders are identified and interviewed. Problems with the present system are discovered and opportunities for improvement are investigated. Objectives for the goal system are recognized.

Elicitation: alternative models for the goal system are analyzed to reach the recognized objectives. Requirements and assumptions on elements of such models are recognized. Scenarios could be involved to help in the elicitation process.

Negotiation and agreement: substitute requirements and assumptions are evaluated, risks are analyzed by the stakeholders and the best alternatives are chosen.

Specification: necessities and assumptions are formulated precisely.

Specification analysis: the specifications are checked for issues such as incompleteness, inconsistency, etc. for feasibility.

Documentation: different decisions made during the requirements engineering process are documented together with the underlying rationale and assumptions.

Evolution: requirements are changed to accommodate corrections, environmental changes and novel objectives. In the requirements engineering process, the following three major types of requirements are:

Functional Requirements (FR): Functional requirements specify the functionality the system shall provide to its users. It describes inputs, outputs and the function it provides.
Nonfunctional requirements: Nonfunctional requirements are used to express the attributes of the system to be

developed. They represent software system qualities (e.g., security, ease of use, maintainability, performance of the system, reliability, etc.)

Constraints: A Constraint is an organizational or technological requirement that restricts the way in which the system shall be deployed.

Traditional approaches for requirements engineering is generally focused on identifying the functional requirements. Nonfunctional requirements and constraints though identified are given less importance at this level. A priority functional requirement over nonfunctional requirements often compromises the quality of the system. Nonfunctional requirements are factored in last few levels of the software development life cycle which may ensure that a small amount of of the preferred quality attributes are not met to the satisfaction of the stakeholders. NFRs are generally tough to express in a measurable way and their analysis also more tricky. Goal- Oriented Requirements Engineering makes fine attempts to solve and other issues. Two major clarifications used during goal decomposition, they are AND and OR. AND-refinement links a goal to a set of sub goals. That means satisfying all sub goals in the refinement is sufficient for satisfying the parent goal. The following figure 3 illustrates AND refinement.

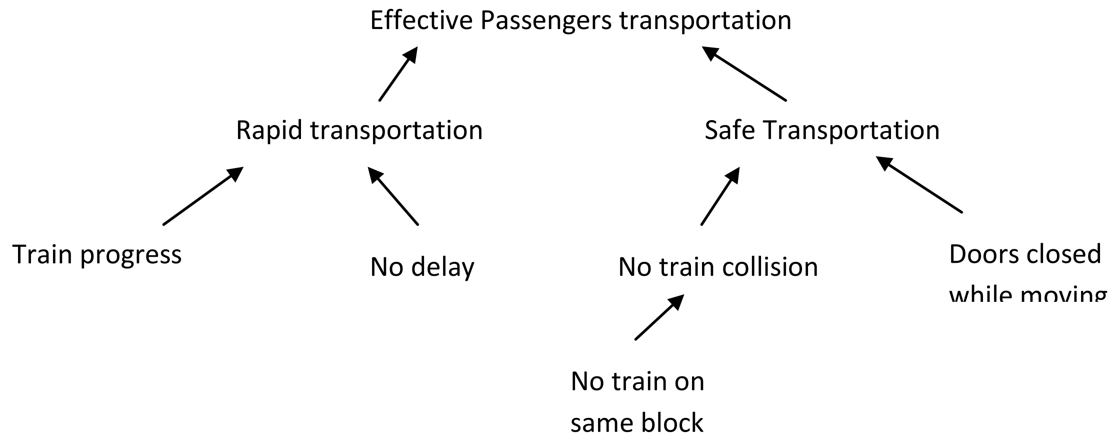

Figure 3. AND refinement

OR-refinement related to goal, an alternative set of refinements. This process satisfying one of the refinements is sufficient for satisfying the parent goal. Figure 4 shows OR refinement.

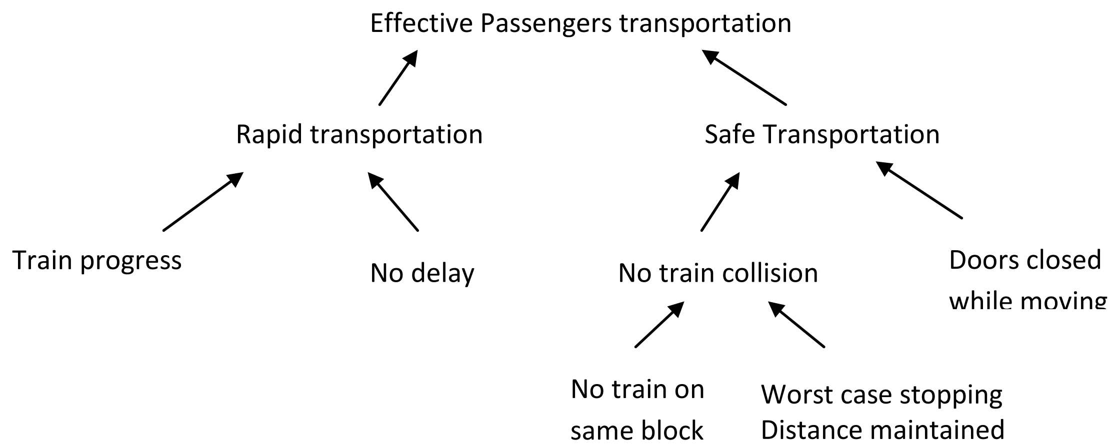

Figure 4. OR refinement

- Goal G is AND-refined into sub goals G1 , ..., G_n iff achieving G_1 , ..., G_n offer to achieving G The set {G_1, ..., G_n } is called refinement of G G_i is said to contribute positively to G_u

- The set $\{G_1, ..., G_n\}$ is a complete AND-refinement of G iff $G_1, ..., G_n$ are sufficient for achieving G in view of known domain properties $\{G_1, ..., G_n, Dom\} \models G_u$
- Goal G is OR-refined into refinements $R_1, ..., G_m$ iff achieving the sub goals of R_i is one alternative to achieving G (1 £i £m) R_i is called substitute for G

Unlike traditional approaches, Goal-oriented approaches lays significance on analysis of non-functional requirements. These NFRs are frequently represented in requirements engineering models by soft goals. There is no proper satisfaction state for a soft goal. Unlike regular goals, soft goals very infrequently be said to be satisfied. For soft goals require to discover solutions that are good, where soft goals are satisfied to a sufficient degree. In many situations the victory of systems based on satisfying of advanced level non-functional requirements.

V.GORE TECHNIQUES

This section presents a survey of GORE techniques and their process and methodology. A diversity of techniques has been projected. GSTH deals with requirement elicitation and defines the uppermost level of goals and proposes a set of heuristics on General System Thinking and Cybernetics. DOSS deals with the Requirement Specification activity and 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 uses the similar models for deriving tabular event-based specifications. GBRAM deals with requirement analysis activity. Set of heuristics (25 totals) are planned in GBRAM, 6 are related to classification, and 8 related to Refinement, 12 heuristics helps in Elaboration. In addition elaboration is supported by different scenarios. GOIG is concerned with requirement elicitation. A process is defined for requirement elicitation based on idea-generation. And they are clustered into goals, and it mostly uses heuristics for idea-generation based elicitation. A-BT generally proposes plans for resolving issues of un-realization of goals by agents. These goals are assigned to agents and agents realize the goals. A goal is unrealized by an agent when agent cannot monitor variables. AGORA strengthens to back for selecting goals to quality estimation, prioritizing, conflicts resolution, and decomposed. It works by attaching attribute values (-10 to 10) to nodes and edges in the AND-OR goal graph at the time of working. The values express how many degrees the sub-goal offers to the achievement of its parent goal. Dissimilar values is given in every edge in OR and similar value is assigned to all the edges in AND decomposition. It uses preference matrix to discover conflicts and gaps of understanding amongst different stakeholders. VVA deals with analysis and offer complete reports for variability of requirements in order to meet the satisfaction of stakeholders.

Algorithm: Requirements selection algorithm

```
Procedure Requirements-selection (p, c, C)
{
    density Di = Pi/Ci
    SortDecreasing (density)
    while i <= n do
    {
        if ci + TotalCost <= C then
        {
            RequirementIsSelected
            TotalCost = ci + TotalCost
            i = i+1
        else
            RequirementIsNotSelected
            i = i+1
        }
    }
    while n > i do{
    if cn + TotalCost <= C then {
        RequirementIsSelected
        TotalCost = ci + TotalCost
        n = n-1
    else
        RequirementIsNotSelected }}
    return TotalCost
}
```

Table 1. Comparative study of different GORE techniques

RE / Methods	Elicitation	Analysis	Specification	Management
DTEBS		yes	yes	yes
GBRAM		yes		yes
AGORA		yes		yes
VVA		yes		yes
GSTH	yes			
KAOS	yes			
GSP		yes	yes	
A-BT		yes		yes

VI.CONCLUSION

The RE methodologies underlying concepts of Goal Oriented Requirements Engineering (GORE) concentrated on recognition, modeling and specifying of different types of goals. A number of contributions have been made to process the requirements engineering process. Most of them seek to link requirements to targets. The Goal taxonomy describes first and foremost achievement, maintenance and soft goals in engineering process. In this paper we present a unifying framework of the Goal Oriented Requirements Engineering concepts through analysis of definition, Modeling and Specifying of Goals. GORE provides an additional approach for elicitation, analysis, elaboration, refinement, specification and modeling of requirements. Several goal-oriented methods have been proposed and a comparative study is made, which serve as a guide for readers to select an appropriate goal-oriented technique to fulfill the requirements engineering needs. The main conclusions are: (i) there is a variety of purposes and uses of goal models in RE, (ii) goal models deserve to be treated as important design artefacts, and (iii) further research is necessary in order to be able to understand the role of goal analysis across different RE activities and offer better methodological support for performing goal-driven processes.

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", Vol. 6 No. 4. pp. 237-251, 2002.

[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"2003.

[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, , 219-225, 1989.

[10] P. Zave,"Classification of Research Efforts in Requirements Engineering", ACM Computing Surveys, 29(4), 1997.

[11] Subhani Shaik et al.,"A Comparative Study Based on Optimization Techniques for Software Cost Estimation", International Journal of Innovative Research in Computer and Communication Engineering, Vol. 4, Issue 10, 2016.

[12] Megha Jain , Abhikrati Shukla , Swati Gupta," A Comparative Study of Financial Performance of Maruti Suzuki India Ltd & Tata Motors Ltd", International Journal of Scientific Research in Multidisciplinary Studies, Vol.5, Issue.9, pp.75-85, 2019.

