

Engineering the Requirements of Data Warehouses: A Comparative Study of Goal-Oriented Approaches

Waffa Setra, Rachid Chalal, Mohamed Chouder

► **To cite this version:**

Waffa Setra, Rachid Chalal, Mohamed Chouder. Engineering the Requirements of Data Warehouses: A Comparative Study of Goal-Oriented Approaches. 5th International Conference on Computer Science and Its Applications (CIIA), May 2015, Saida, Algeria. pp.254-265, 10.1007/978-3-319-19578-0_21 . hal-01789961

HAL Id: hal-01789961

<https://hal.inria.fr/hal-01789961>

Submitted on 11 May 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Engineering the Requirements of Data Warehouses: A Comparative Study of Goal-oriented Approaches

Waffa Setra¹, Rachid Chalal¹, and Mohamed Lamine Chouder¹

¹LMCS (Laboratoire de Methodes de Conception des Systemes)
ESI (Ecole nationale Supérieure d'Informatique), Algiers, Algeria
{w_setra,r_chalal,m_chouder}@esi.dz

Abstract. There is a consensus that the requirements analysis phase in the development project of a data warehouse (DW) is of critical importance. It is equivalent to application of requirements engineering (RE) activities, to identify the useful information for decision-making, to be met by the DW. Many approaches has been proposed in this field. Our focus is on goal-oriented approaches which are requirement-driven DW design approaches. We are interested in investigating to what extent these approaches went well with respect to the RE process. Thus, theoretical foundations about RE are presented, including the classical RE process. After that, goal-oriented DW design approaches are described briefly; and evaluation criteria, supporting a comparative study of these approaches, are provided.

Keywords: Data warehouse; goal-oriented approach; requirements engineering process; comparative study; evaluation criteria.

1 Introduction

In the last years, great interest has been shown in the field of Data warehouse (DW) design [1]. Indeed, many design approaches has been proposed in this field. These approaches are usually classified into two categories: data-driven and requirement-driven. The former also called supply-driven designs the DW starting from a detailed analysis of the data sources [1,2,3,4]. The user is not much involved in this category of approaches [5]. The latter also called demand-driven, attempts to identify the information requirements from business users [6,7,8,9]. We focus on the requirement-driven approaches.

Requirements analysis is the initial phase of DW design cycle [10]. It is equivalent to application of requirements engineering (RE) activities, to identify the useful information for decision-making, to be met by the DW. Requirement-driven DW design approaches define requirements through different orientations: process, user and goal. Process-oriented approaches [5], [11,12,13] analyze requirements by identifying the business processes of the organization. User-oriented approaches identify the target users and specify their individual needs to integrate them into a unified requirement model [14,13]. Goal-oriented approaches [8], [15,16,17,18,19] identify goals and

objectives of users that guide decisions at various levels of the organization. Most of requirement-driven DW design approaches are goal-oriented. Many authors recognize that these approaches provide a better definition of user requirements [15], [20], for two reasons: (i) the gathered requirements are validated by identifying conflicting goals; (ii) the different modelling alternatives to achieve a goal are provided [21]. However, in the beginning of a DW development project, identifying users' objectives and goals is a crucial step, where achieving the goals is an important indicator of the organization's activity [22].

RE is an important field dedicated to requirements definition. It is concerned by transforming users' expectations into agreed requirements through a well-defined process, called RE process. RE applied in the field of DW allows determining users' requirements. In this paper, our interest is on goal-oriented DW design approaches. Many approaches have been proposed in this field. As yet, there is no common strategy for these approaches [23]. Furthermore, we argue that the process of RE is not completely applied in this field. Besides, there is no common RE process for DWs. Indeed, if we consider that a structure of an approach is the set of its activities; the proposed approaches does not share the same structure. The purpose of this paper, is to extract the invariant steps from the classical RE process, in order to identify a set of criteria (see section 3.1), to allow evaluating the goal-oriented DW design approaches; in other words, to see what does each approach provide to support those criteria.

In this work, a general overview is shown, as well as a comparative study, of six famous goal-oriented approaches for DW design. The comparison highlights the evaluation criteria based on the classical RE process. The remainder of this paper is structured as follows: section 2 gives theoretical foundation of RE. In section 3, a brief description of goal-oriented approaches is given as well as the criteria to evaluate these approaches are defined in order to make a comparative analysis. Finally, Section 4 summarizes our work and presents our conclusions.

2 RE: Theoretical Foundation

2.1 What Is RE

The first definition of RE was given in the software engineering area [22]. It was qualified as visionary, referenced by many authors[24],[25] and specifies that: "requirements definition is a careful assessment of the needs that a system is to fulfil. It must say why a system is needed, based on current or foreseen conditions, which may be internal operations or an external market. It must say what system features will serve and satisfy this context. And it must say how the system is to be constructed" [26]. [26] stated that "requirements definition must encompass everything necessary to lay the groundwork for subsequent stages in system development".

Thus, RE must address firstly the 'why' dimension, justifying the existence of the system, which many authors translate to 'goal' or 'objective' [19], [27,28,29]. Then it addresses the 'what' dimension, specifying the system's functions to fulfil the goals [27,28,29]. Besides, RE must take into account the 'how' dimension, by specifying the constraints to be applied on the system under consideration. Lamsweerde [27] has

added the ‘who’ dimension, to address assigning responsibilities to humans, devices or softwares. While Zave [29] claimed that RE deals also with the evolution of the software’s specifications over time. In RE, two types of requirements exist: Functional and non-functional. The former describes the functions to be performed by the system. The latter defines constraints on the way the functional requirement should be satisfied. A taxonomy of non-functional requirements can be found in [27].

The above-presented definition is taken from the software engineering field. In the literature, the aspects of RE engineering highlighted by the authors above were taken back in other fields: information systems [20], [30,31,32], DWs [17], [19], [33]. Since our interest is in the field of DWs, RE for DWs is detailed in section 3.

2.2 The RE Process

The RE process is composed of several activities highly intertwined. This property is observed at the different RE process models proposed [27], [34,35,36], while other authors [27], [34], [36] affirm that the RE process is iterative and incremental. Two common concepts frequently used in RE : system-as-is and system-to-be [27]. The former means the system as it exists now while the latter means the system as we want it to be. The role of RE is to identify requirements that will change the system from the as-is state to the to-be state. We consider that a system is a set of components (human, software, hardware...) interacting with each other to satisfy a purpose.

Nuseibeh and Easterbrook [37] proposed a RE process that consists of six activities: elicitation, modelling, analysis, specification, validation and management of requirements. Other authors, in particular Kotonya and Sommerville [34] have highlighted all the above activities except modelling activity which was included in the elicitation activity. The standard (ISO/IEC/IEEE/29148:2011) [38], proposed four activities for the RE process which are: defining the requirements of the stakeholders, requirements analysis, verification and validation, and finally the requirements management.

These differences are not necessarily justified by an omission or addition of activities, but can be considered as different ways of seeing the process. In the following, the activities of the RE process are described based on [27], [36] and [38].

Domain Understanding and Requirements Elicitation. Domain understanding consists of studying the system-as-is within its organizational and technical context. It leads to understand the domain in which the problems are rooted and identify the roots of the problems [27]. As a result:

- Stakeholders involved in the RE process must be identified;
- A comprehensive picture, of the organization’s objectives, actors, roles and dependencies among them, in which the system-as-is takes place, is formed;
- The scope of the system-as-is is defined (objectives, components, information flowing through it and constraints);

- Strengths and weaknesses of the system-as-is, as perceived by the identified stakeholders are determined;
- A glossary of terms should be established to provide definitions of key concepts on which everyone should agree.

This result will be utilized for the rest of RE activities. Once the requirements engineer acquires some knowledge about the domain, he starts eliciting requirements. Elicitation is “a cooperative learning process in which the requirement engineer and the system stakeholders work in close collaboration to acquire the right requirements. This activity is obviously critical. If done wrong, it will result in poor requirements and, consequently in poor software” [27]. In this activity, the requirement engineer aims to collect, capture, explore and model the requirements of the system-to-be from a multitude of sources. Modelling is important in this activity because the system needs to be represented faithfully, so that this representation can be understandable by users.

To perform the elicitation activity, a variety of techniques exists: interviews, questionnaires, surveys, prototyping, observation... These techniques has been classified in [27], [37].

Evaluation and Agreement. This activity aims to examine and interpret the elicitation phase results, in order to:

- Clarify the requirements, remove inconsistencies and ensure completeness and non-redundancy;
- Identify and resolve conflicting concerns;
- Assess and resolve risks associated with the system that is being shaped;
- Compare the alternative options identified during elicitation with regard to quality objectives and risks, and select best options on that basis;
- Prioritize requirements in order to resolve conflicts or avoid exceeding budget and deadlines etc...

To support the evaluation activity, a variety of qualitative and quantitative techniques is presented in [27].

Specification and Documentation. The agreed requirements emerging from the evaluation activity must be detailed, structured and documented in the specification document. So that they can be understood by all users involved in the RE process. Specification can be formal, semi-formal or informal, see [27] for more details. The specification document is the main product of RE [36], [38]. It traces the process and includes descriptions of various elements, techniques and tools that have led to the result. Requirements must be classified by users to prepare the validation step [38].

Requirements Consolidation. Also called validation activity, as referred by [37,38]. Requirements engineer detects and corrects errors. He certifies that the requirements

meet the expectations of users, and define the expected functionality of the system. A variety of verification method is proposed by the standard (ISO/IEC/IEEE 29148:2011) [38]. Among the products of this step: a corrected version of the requirements produced by the previous activity; a set of acceptance test sets produced from the requirements specification; and an eventual prototype of the system-to-be.

Requirements Evolution. This activity considers the different versions of requirements. Indeed, Requirements may change due to different causes. Thus requirements before change and after change as well as the causes of change have to be noted in the specification document. Therefore a new version of this document is produced at each change. In [30], [38], a whole process for change management is proposed. Requirements change is inevitable, it should be anticipated from the beginning as well as requirements traceability should be maintained. The former is guaranteed by assigning an attribute to each requirement, in order to specify whether it is stable or may change. The latter, has to be planned from the beginning of the project for two reasons: (i) trace the evolution of requirement and justify any change and (ii) track back the requirement into the initial objectives so that one can argue that they are satisfied.

These activities compose the classical RE process, which emanate from the software engineering field. A DW can be seen as a software system having the specificity of supporting decision making. Engineering the requirements of DWs is a step of DW design known as requirement analysis for DW design. In the next section, this step is discussed through goal-oriented approaches. A set of criteria to evaluate these approaches are described, and comparative analysis is made among six famous goal-oriented approaches.

3 RE for DWs

In this section, a link is made between RE process seen above (section 2), and requirement analysis for DW design. First of all, it is clear that the system-as-is, is represented by the organization before building the DW, while the system-to-be is the DW within the organization. Second, talking about the RE dimensions mentioned above “why, what, who and how” (subsection 2.1); the “why” dimension concerns identifying the high-level objectives and goals of the stakeholders and decision makers involved in the DW development project [16]. While the “what” dimension, is concerned by identifying what information is relevant for decision making [18]. We call that “useful information” for decision making, which should be stored in the DW. The “who” dimension, cares about identifying the stakeholders and decision makers involved in the DW development project. Finally the “how” dimension is not introduced in DWs. We assume that it is concerned about implementation constraints to be applied on the DW. The concept requirement introduced above (section 2) represent, for DW, information requirements that supports decision-making [19], [33], [39]. In the following, we use the term requirement to refer to information requirement.

Despite the large number of goal-oriented DW design approaches proposed, as yet, there is no common strategy of requirement analysis in DW design [23]. Besides, we

argue that there is no common RE process for DW. [40] Proposed a set of activities for goal-oriented approaches, with various models for each activity. This work was exploited by [41] in a comparative study of goal-oriented DW design approaches. The authors evaluated these approaches according to the models used in each step of requirements analysis. Our purpose, is to extract the invariant steps from the classical RE process to be applied in DW requirement analysis, in order to identify a set of criteria to evaluate goal-oriented DW design approaches. In the following subsections, those criteria will be described, and will be used to compare the goal-oriented DW design approaches. Then we give a brief description of the compared approaches, and discuss the result.

3.1 Evaluation Criteria

The context of RE for DW is specific, since DW is dedicated to decision making [33]. We assume that the classical RE process is not completely applied in DW requirement analysis. Thus, in order to see what are the current practices in this field, we studied this classical RE process in the context of DW, and extracted a set of evaluation criteria, then assigned for each criterion a coefficient that reflects its weight in the process of requirements analysis for DWs. The assigned weights are of three types:

- Elementary: criterion qualifies an elementary activity of the RE process. (weight 1)
- Important: Elementary and requires great importance. (weight 2)
- Mandatory: Important and must qualify each approach. (weight 3)

The criteria we suggest, include the following:

Elicitation: In goal-oriented DW design approaches, requirement elicitation is the most complex activity [41] for the following reasons: in one hand DWs are used exclusively for decision making [15], [19], [42]. In the other hand, goal-oriented DW design approaches are based on the analysis of high-level goals [27]. The problem, at this level, is in extracting the goals from decision makers. If in case a decision maker knows how to express his goals, which is not often the case, in some other cases, decision makers poorly express their goals, or less, they are not able to formulate them. The requirement elicitation is the first activity of the RE process, the remaining steps depend on it. If the goals are poorly defined, the DW may not meet the needs of decision makers. Considering the importance of this phase it will have weight (2).

Specification: this criterion qualifies the specification activity, where the elicited goals are analyzed (conflict detection, errors, redundancy) and modelled. The concerns of the requirements engineer is to find, according to the decision makers, which models may be used to specify their needs so that they can be able to understand. It is about mapping the real-world needs into a requirements model [40]. It is a core activity for the RE process and prepares for validation step, therefore it bear the weight (2).

Validation: A consensus on the elicited goals, between the requirement engineer and decision makers must be established through validation. Validation of requirements is paramount for further stages of DW design. If requirements are not validated

by decision makers, the risk that the DW will not address their needs increases, which will bring the project to failure. Therefore, validation is mandatory and deserve the weight (3).

Requirements' evolution management: One will not flee the fact that requirements evolve throughout the requirement analysis in DW design. Besides, it's not impossible that they evolve even after validation. A DW not taking into account the evolving requirements is certainly not at the same effectiveness as another one supporting it. Furthermore, a decision maker, always, seeks to meet its objectives in one way or another. Elsewhere, it does not include the fact that he succeeds to express all his needs. It is important to plan, from the beginning of requirement analysis, for alternatives to the defined requirements [27], [42]. Also, anticipate requirements subject to change or evolution. This criterion represent the requirement evolution activity in the classical RE process. Besides, regarding its contribution to the effectiveness of DW, this criterion deserves the weight (2).

Traceability: How will it be possible to affirm that a goal is satisfied? How to define to which goal is associated a given requirement? To answer this, traceability is introduced. It consists on tracing the path from the goal to the relevant information in DW [43]. Traceability helps assessing the impact of changes and rationale comprehension, by identifying which parts of the implementation belong to which requirement [44]. It also supports the reusability and maintainability of DW, since the scope of each part of the project is known and defined thanks to the traces. In turn, these benefits help lowering the costs associated with the project [45,46]. Distinction is made between post-traceability and pre-traceability [47]. The former is about the traceability of the requirement, its deployment, and its use. Whereas the latter is the traceability of a requirement back to its origin which is goal in our context. Thus, since the first RE's task, it is essential to think about keeping trace of everything. This is necessary to justify delays and possibly identify the cause of failures. For all these reasons, the weight (3) is the most suitable for this criterion.

Reusability: DW implementation is a complex and costly activity in resources and time [48]. It also requires specific developments to the characteristics and needs of the organization. However, decision-making projects for the same field of activity or even different business areas have similarities [49]. It is certainly possible to find situations which we have already faced; avoid falling in unrealistic requirements on the basis of earlier experiences; or even propose to decision makers new requirements through anticipation [8]. Reusing requirements, or reusing existing Data marts [8] or even DWs, promote saving time and reliability in future projects. Therefore, this is elementary for each approach and carries the weight (1).

3.2 Comparative Analysis

Six famous goal-oriented DW design approaches have been studied. The study consists of capturing the satisfied criteria for each approach. In the following, these approaches are briefly described.

1. (Bonifati & al 2001) [15]: the approach starts by gathering information from business analysts and/or managers about the company goals and needs [15]. This is accomplished through the Goal/Question/ Metrics paradigm. The goals obtained are aggregated and refined, until a small number of goals subsume all the collected information. Each aggregated goal is specified by an abstraction sheet, which expresses its characteristics in great detail. From abstraction sheets, it is possible to extract the specifications of ideal star schemas which represent users' information requirements for the DW.
2. (Paim & al 2003) [42]: The approach is named DWARF. The authors adopted the classical RE process for DW, adding traceability and compliance of requirements. DWARF is divided into a series of well-defined stages. Each stage presented in a development cycle, applies different levels of abstraction that detail the application more deeply each time, with the goal of creating a baseline for requirements. The latter are specified for data marts and grouped to specify the DW. The authors insisted on the documentation activity of each step of the approach.
3. (Gam & Salinesi 2006) [8]: the approach is called CADWA. In CADWA, the information requirement are extracted from (i) the goals presented by the strategic plan of the organization, (ii) decision makers business plans, (iii) transactional systems, and (iv) the existing DW or data marts models that can be reused. In the next stage the authors create a model using the MAP goal model, to represent the current and future information requirements of decision makers.
4. (Mazon & al 2007) [33]: This approach starts by identifying a hierarchy of goals with tree levels of abstraction: Strategic goals which are fulfilled by decisional goals which are in turn fulfilled by information goals. From information goals, the information required for decision making is directly derived. The authors adapted the strategic rational (SR) model of the i* modelling framework [50] and used it to specify goals and information requirements of decision makers. Traceability does not appear explicitly in the approach. However, the models used allowed pre-traceability of requirement from goals to information requirements.
5. (Giorgini & al 2007) [17]: the approach is called GRanD. It adopts two perspectives for DW requirement analysis. The former is organizational modeling centered on stakeholders and aims to shape the organization. The latter is decisional modeling which is directly related to the information needs of decision makers [17]. Traceability is not made explicit by the approach. However, GRanD is based on an adapted i* modelling framework [50]. Therefore, pre-traceability of requirements can be guaranteed through the proposed models.
6. (Prakash & Gosain 2008) [19]: The authors focus first on the context of the organizational goals. Thus, A GDI (Goal Decision Information) model was proposed with three levels of abstraction. It starts by identifying organizational goals. A goal enables to identify the set of decisions that are relevant. For each decision, a set of

required information to make it, is determined. This organizational view is translated into a technical view by the use of the informational scenario. The latter is written for each decision available in the GDI organization scheme, to capture the required information for decision making.

Table 1. A comparative study of the goal-oriented DW design approaches.

| The approach | Elicitation (2) | Specification (2) | Validation (3) | evolution management (2) | Traceability (3) | Reusability (1) | weight |
|----------------------------------|-----------------|-------------------|----------------|--------------------------|------------------|-----------------|--------|
| Bonifati & al 2001 | X | X | | | | | 4/13 |
| Paim & Castro 2003 | X | X | X | X | X | | 12/13 |
| Gam & Salinesi 2006 | X | X | | | | X | 5/13 |
| Mazon & al 2007 | X | X | | | X | | 7/13 |
| Giorgini & al 2007 | X | X | | | X | | 7/13 |
| Prakash & Gosain 2008 | X | X | | | X | | 7/13 |

A set of conclusions is made on basis of table 1. First, all the approaches focus on the elicitation and specification activities of the RE process. These two activities are basic for the RE process. Second, validation criterion which represents validation activity, has not shown great importance from the approaches. It is mentioned above that it is of great importance (section 3.1). Besides, it refers to a basic activity of the classical RE process. Consequently this criterion needs more importance for next approaches. Third, traceability is not well addressed. It is made implicit by the models proposed. More efforts has to be made to satisfy that criterion, due to its contribution to the

proper conduct of the RE process. Forth, requirement evolution management criterion is only satisfied by the DWARF approach. It was addressed by a horizontal activity since the beginning of the approach until the end. Finally, concerning reusability criterion, only CADWA [8] applied it by reusing existing structures of DWs or data marts.

DWARF [42] has encompassed the large number of criteria since it applied the classical RE process. Consequently, it has the highest weight among the approaches. [17], [19] and [33] has well addressed the elicitation and specification criteria. This what made of them powerful approaches, but still, they have to incorporate validation activity in the process of the approach, and plan for a better traceability.

4 Conclusion

In this paper, a comparative study was made among goal-oriented DW design approaches. We have investigated to what extent these approaches went well with respect to the classical RE process. Our study was based on six evaluation criteria, which were defined directly from the RE process for many reasons. We argue that a DW is more than a software system, it has the specificity of providing useful information to support decision-making. Thus, RE process for DWs has to be applied carefully. In addition, there is no standard approaches for DW design despite the considerable efforts made in the field. The main motivation of this work is to serve as a starting point for researchers to think at developing a standard RE process for DW design. Consequently, this comparative study can be useful for researchers in achieving a common understanding in the field and providing a solid foundation for the research community.

References

1. Inmon, W.H.: Building the data warehouse (2nd ed).(1996).
2. Golfarelli, M., Maio, D., Rizzi, S.: The dimensional fact model: a conceptual model for data warehouses. *International Journal of Cooperative Information Systems* 7, 215 - 247.(1998).
3. Hüsemann, B., Lechtenbörger, J., Vossen, G.: Conceptual data warehouse design. In:Proceedings DMDW, Stockholm, Sweden, pp. 3 - 9.(2000).
4. Moody, D., Kortink, M.: From enterprise models to dimensionalmodels: a methodology for data warehouse and data mart design. In: Proceedings DMDW, Stockholm, Sweden. (2000).
5. List, B., Bruckner, R.M., Machaczek, K., Schiefer, J.: A Comparison of Data Warehouse Development Methodologies Case Study of the Process Warehouse. In: DEXA 2002, LNCS 2453, pp. 203 - 215. Springer-Verlag Berlin Heidelberg.(2002).
6. Bruckner, R., List, B., Scheifer, J.: Developing Requirements for Data Warehouse Systems with Use Cases. In: AMCIS 2001 Proceedings., pp. 329 - 335.(2001).
7. Winter, R., Strauch, B.: A Method for Demand-driven Information Requirements Analysis in Data Warehousing Projects. In: the Hawai'i International Conference on Systems Sciences , January 6-9.(2003).
8. Gam, I., Salinesi, C.: A Requirement-driven Approach for Designing Data Warehouses. Foundations for Software Quality, Jun 2006, Luxem-bourg, Luxembourg., pp. 1 - 15 (2006)

9. Prakash, N., Gosain, A.: Requirements Driven Data Warehouse Development. 1st JIIT A10, Sector 62 NOIDA 201307, India.(2003)
10. Golfarelli, M.: Data warehouse life-cycle and design. Encyclopedia of Database Systems, pp. 658-664. Springer.(2009).
11. List, B., Schiefer, J., Tjoa, A.M.: Process-Oriented Requirement Analysis Supporting the Data Warehouse Design Process A Use Case Driven Approach.(2000).
12. Schiefer, J., List, B., Bruckner, R.: A holistic approach for managing requirements of data warehouse systems. AMCIS 2002 Proceedings 13.(2002).
13. Kimball, R., Ross, M.: The data warehouse toolkit: the complete guide to dimensional modeling. John Wiley & Sons.(2011).
14. 24765:2010, ISO/IEC/IEEE.: Systems and software engineering Vocabulary.(2010).
15. Bonifati, A., Cattaneo, F., Ceri, S., Fuggetta, A., Paraboschi, S.: Designing Data Marts for Data Warehouses. ACM Transactions on Software Engineering and Methodology, Vol. 10, No. 4, October 2001, Pages 452-483.(2001).
16. Giorgini, P., Rizzi, S., Garzetti, M.: Goal-Oriented Requirement Analysis for Data Warehouse Design.(2005).
17. Giorgini, P., Rizzi, S., Garzetti, M.: GRAnD: A goal-oriented approach to requirement analysis in data warehouses. Decision Support Systems 45, 4 - 21.(2007).
18. Mazón, J.-N., Trujillo, J.: An MDA approach for the development of data warehouses. Decision Support Systems 45, 41-58.(2008).
19. Prakash, N., Gosain, A.: An approach to engineering the requirements of data warehouses. Requirements Eng 13, 49 - 72.(2008).
20. Rolland, C.: Reasoning with goals to engineer requirements. Enterprise Information Systems V, pp. 12-20. Springer.(2005).
21. Giorgini, P., Mylopoulos, J., Nicchiarelli, E., Sebastiani, R.: Formal reasoning techniques for goal models. Journal on Data Semantics I, pp. 1-20. Springer.(2003).
22. Stefanov, V., List, B.: Business Metadata for the DataWarehouse. In: Enterprise Distributed Object Computing Conference Workshops, 2006. 10th IEEE International, pp. 20-20. IEEE. (2006).
23. Cravero Leal, A., Mazón, J.N., Trujillo, J.: A business-oriented approach to data warehouse development. Ingeniería e Investigación 33, 59-65.(2013).
24. Lamsweerde, A.V.: Requirements engineering in the year 2000: A research perspective. In: 22nd International Conference on Software Engineering, Invited Paper. ACM Press.(2000).
25. Mylopoulos, J., Borgida, A., Yu, E.S.K.: Representing Software Engineering Knowledge. Automated Software Engineering, Kluwer 4, 291-317.(1997).
26. Ross, D.T., Schoman, K.E.: Structured Analysis for Requirements Definition. IEEE transactions on software engineering 3, 10. (1977).
27. Van Lamsweerde, A.: Requirements engineering: from system goals to UML models to software specifications. (2009).
28. Rolland, C., Prakash, N.: From Conceptual Modelling to Requirements Engineering. Annals of Software Engineering on Comparative Studies of Engineering Approaches for Software Engineering.(2001).
29. Zave, P., Jackson, M.: Classification of Research Efforts in Requirements Engineering. ACM Computing Surveys 29, 7.(1997).
30. Dardenne, A., Lamsweerde, A.V., Fickas, S.: Goal-directed requirements acquisition. Science of Computer Programming 20, 3-50.(1993).
31. Rolland, C., Grosz, G., Kla, R.: Experience With Goal-Scenario Coupling In Requirements Engineering. In: Fourth IEEE International Symposium on Requirements Engineering (RE'99), 7-11 June 1999.(1999).
32. Si-Saïd, S., Rolland, C.: Formalising guidance for the crews goal-scenario approach to requirements engineering. In: the 8th European - Japanese Conference on Information Modelling and Knowledge Bases. May 25-29, pp. 20.(1998).

33. Mazón, J.-N., Pardillo, J., Trujillo, J.: A model-driven goal-oriented requirement engineering approach for data warehouses. *Advances in Conceptual Modeling—Foundations and Applications*, pp. 255-264. Springer (2007).
34. Kotonya, G., Sommerville, I.: *Requirements Engineering : Process and Techniques*. (1998).
35. Loucopoulos, P., Karakostas, V.: *System Requirements Engineering*, McGraw-Hill Book Company Europe (1995).
36. Rolland, C.: De la modélisation conceptuelle à l'ingénierie des exigences. pp. 23. *Journal Techniques de l'Ingénieur* (2011).
37. Nuseibeh, B., Easterbrook, S.: *Requirements Engineering : A Roadmap*. In: *International Conference on Software Engineering*, ACM Press . 4-11 juin, (2000).
38. 29148:2011, ISO/IEC/IEEE: *Systems and software engineering — Life cycle processes — Requirements engineering*. pp. 95 (2011).
39. Winter, R., Strauch, B.: *Information Requirements Engineering for Data Warehouse Systems*. *ACM Symposium on Applied Computing SAC'04*, March 14-17, 2004, Nicosia, Cyprus, pp. 1359 - 1365 (2004).
40. Kavakli, E., Loucopoulos, P.: *Goal driven requirements engineering: evaluation of current methods*. In: *Proceedings of the 8th CAiSE/IFIP8*, pp. 16-17. (2003).
41. Leal, A.C., Sepúlveda, S., Mate, A., Mazón, J.-N., Trujillo, J.: *Goal oriented requirements engineering in data warehouses: A comparative study*. *Ingeniería E Investigación* 34, 66 - 70 (2014).
42. Paim, F.R.S., Castro, J.F.B.d.: *DWARF: an approach for requirement definition and management of DW systems*. In: *11th IEEE International Requirement Engineering Conference*. (2003).
43. Hull, E., Jackson, K., Dick, J.: *Requirements Engineering*, 2 nd edition. pp. 201. Springer (2005).
44. Antoniol, G., Canfora, G., Casazza, G., De Lucia, A., Merlo, E.: *Recovering traceability links between code and documentation*. *Software Engineering, IEEE Transactions on* 28, 970-983 (2002).
45. Ramesh, B., Stubbs, C., Powers, T., Edwards, M.: *Requirements traceability: Theory and practice*. *Annals of software engineering* 3, 397-415 (1997).
46. Ramesh, B., Jarke, M.: *Toward reference models for requirements traceability*. *Software Engineering, IEEE Transactions on* 27, 58-93 (2001).
47. Pohl, K.: *PRO-ART: enabling requirements pre-traceability*. In: *Proceedings of the Second International Conference on Requirements Engineering*, pp. 76-84. (1996).
48. Carneiro, L., Brayner, A.: *X-META: A Methodology for Data Warehouse Design with Metadata Management*. In *Design and Management of Data Warehouses (DMDW)* 13 - 22 (2002).
49. Annoni, E., Ravat, F., Teste, O., Zurfluh, G.: *Les systèmes d'informations décisionnels : une approche d'analyse et de conception à base de patrons*. *Revue des Sciences et Technologies de l'Information, série ISI, «Méthodes Avancées de Développement des SI »* 10, 81 – 106 (2005).
50. Yu, E.S.: *Towards modelling and reasoning support for early-phase requirements engineering*. In: *Requirements Engineering, 1997., Proceedings of the Third IEEE International Symposium on*, pp. 226-235. IEEE, (1997).