CrossMark

ORIGINAL ARTICLE

A multifactor approach for elicitation of Information requirements of data warehouses

Deepika Prakash[1] · Naveen Prakash[2]

Abstract Whereas requirements engineering for transactional systems aims to discover the functionality of the system-to-be, data warehouse requirements engineering aims to discover the Information contents of the data-warehouse-to-be. Though notions of goals, Decisions, business processes, business events have been used to set the context for Information discovery, the move from these to obtain the relevant Information is largely ad hoc, unguided, and does not provide traceability of Information. We propose four elicitation techniques that are inferred from manager concerns during Decision making and that provide guidance and traceability. These form a suite such that each augments the set of already discovered Information. Consequently, the possibility of missing requirements is reduced, thereby making for more effective requirements engineering.

Keywords Information · Decision · Decision–Information · CSFI · ENDSI · MEANSI · Outcome feedback

1 Introduction

In recent years much attention has been paid to the issue of data warehouse requirements engineering, DWRE. There is a fundamental difference between traditional requirements engineering, RE, for transactional systems and that for data warehousing. The former is oriented toward discovering the functionality of the system-to-be. The discovered functionality is then implemented or operationalized in the system to be built. In contrast, the problem of DWRE is to determine the Information contents of the data-warehouse-to-be. This Information is to be structured into multi-dimensional form. Thus, DWRE aims at the determination of facts and dimensions comprising the data warehouse.

Much interest in RE for transactional systems is on goal-oriented [1, 2] and scenario-oriented techniques [3, 4]. These were coupled together to yield the goal–scenario coupling technique [5, 6]. Goal orientation uses Means–Ends analysis to reduce goals, and the goal hierarchy identifies the goals that are to be operationalized in the system. Notice the near absence of the data/Information aspect in goal orientation. Scenario orientation reveals functionality and its variations by identifying typical interaction between the system and the user. Even though example data are shown to flow across the system–user interface, focus is not on the data aspect; data and their modeling are largely ignored in scenario-oriented RE. Goal–scenario coupling allows development of a scenario for a goal of the goal hierarchy. Consequently, variations of goals are discovered in its scenario. Due to this variation, any new functionality indicated by the scenario is introduced in the goal hierarchy. Thus, a mutually cooperating system is developed to better discover system goals. Again, notice that data are largely ignored.

A number of proposals for goal-oriented DWRE are available, and all of these link goals with data, that is, all are aimed at obtaining the multi-dimensional structure of data warehouses from goals [7–13]. Other than goal-oriented approaches, DWRE can also be based on Key Performance Indicators, KPIs. The idea [14, 15] is to determine the Information required to estimate these indictors. Notions

✉ Deepika Prakash
dpka.prakash@gmail.com

Naveen Prakash
praknav@hotmail.com

[1] School of Mathematics, Statistics and Computational Science, Central University of Rajasthan, Kishangarh, India

[2] ICLC, 21/8 S. Bhagat Singh Marg, New Delhi 110001, India

🖄 Springer

of business processes/events form the basis of the BEAM* approach [16], and the elicited Information pertains to these concepts.

Our analysis of DWRE approaches, presented in the next section, shows that even though these approaches attempt to elicit Information, the method of elicitation remains largely ad hoc and undefined. In other words, there is no articulation of the methods, tools, and techniques that can be deployed in discovering relevant Information.

Whereas in goal-oriented transactional requirements engineering techniques, the stakeholder is at least asked to concentrate on goal achievement; in Information elicitation such a focal point is missing, and Information elicitation is overly dependent on stakeholder experience. Our attempt here is to provide support in the Information elicitation task by defining focal points. Notice that we are looking for more than one focal point. This is to better cover the range of Factors that contribute to Information elicitation. Further, we assume that our focal points should have high buy-in for the stakeholder. Therefore, we begin by identifying high-stake issues in an organization, treat each issue as a focal point, and then develop a stepwise approach to elicit Information for each focal point.

The layout of the paper is as follows. In the next section, we analyze DWRE approaches to show that Information elicitation is ad hoc. Thereafter, in Sect. 3, we identify some important managerial concerns. Our elicitation techniques address these concerns, and therefore, there shall be high manager buy-in in the elicitation process. Further, these concerns yield a suite of techniques that can be deployed to minimize the chances of missing requirements. Section 4 contains a discussion of our Decision requirement model. This model forms the technological basis for the elicitation techniques as presented in Sect. 5. In Sect. 6, the concepts that form the basis of our elicitation techniques are compared with similar concepts found in MIS. Section 7 discusses the lessons learned from an application of our methods in the hospital domain. Finally, Sect. 8 is the concluding section.

2 Analysis of DWRE methods

Boehnlein and Ulbricht [7, 8] rely on the semantic object model (SOM) framework. After building a goal model for the business at hand, the business processes that are performed to meet the goals are modeled. The business application systems resulting from these are then used to yield a schema in accordance with the Structured Entity Relationship Model, SERM. Business objects get represented as entities of SERM, and dependencies between entities are derived from the task structure. Thereafter, a special fourth stage is added to SOM in which only those attributes that are relevant to Information analysis required for Decision making

are identified. The authors then convert the SERM schema to facts and dimensions; facts are determined by asking the question how can goals be evaluated by metrics. Dimensions are identified from dependencies of the SERM schema.

Bonifati [9] carries out goal reduction by using the Goal–Quality–Metric approach. Once goal reduction is done, abstraction sheets are built. These sheets contain Information, among other Information, about the quality focus of the goal and variation Factors. The former delivers measures for quality of goals that become facts, whereas the latter yield dimensions. Quality is considered as Factors that are of relevance to the goal. There is no guidance on what constitutes quality, but some examples are provided. These are cost, performance, resources required, etc.

In [11], Decisions are associated with goals and for each Decision, relevant Information is obtained by writing Informational scenarios that are sequences of Information requests expressed in an SQL-like language. An Information scenario is thus a typical system–stakeholder interaction to identify Information required for a Decision. The Information obtained is then converted into an ER diagram for conversion into fact–dimension schema using Golfarelli's algorithm. Typical Information retrieval requests use the rather fuzzy notion of "relevant Information." What constitutes "relevance" is not spelled out.

Yet another approach is to modify the *i** model, to yield [12] the "*i** for DW Profile." Goals are at three abstraction levels, strategic goals, Decision goals, and Information goals. Strategic goals refer to the main goals of the business, for example analyze sales; Decision goals, for example open new store, are for achieving strategic goals; and finally Information goals, for example analyze purchases, specify the nature of the analysis to be made. Further, notions of business process, measure, and context are introduced. Measures and contexts are then transformed into facts and dimensions.

In GRAnD [13], the early phase of Tropos has been extended to the requirements engineering of data warehouses. Actor and Rationale diagrams are developed as in Tropos. The goals in the latter are associated with facts. Facts are the recordings that have to be made when the goal is achieved. Additional attributes relevant to goals are discovered and attached to goals. These attributes are the data associated with goals. The next stage is of Decision modeling. Here, the rationale diagram is viewed from the point of view of Decision makers. Decision maker goals for analyzing are set up and associated with facts. Facts are objects of analysis and correspond to business events in the organization. Often, facts are obtained from the first phase. Thereafter, dimensions are associated with facts by examining leaf goals.

The approach of Information goals [12] considered above was extended for better alignment of the data warehouse with the business. This was done [17] by front

ending it with vision, mission, objective, strategy, tactic (VMOST) analysis of the business. Once this is done, the approach of Information goals is then followed.

Though there is heavy momentum behind the goal-oriented approach [18], there are other techniques that have also been proposed. One of these is business indicator based. We are aware of two proposals here. The first proposal [14, 19] models business indicators as functions and identifies the needed parameters and return type. That is, input and output Information needed to compute a business indicator is determined. However, this is only a part of their total proposal. The remaining part, that of determining Decision alternatives, has not been reported yet. Therefore, at this stage they do not tell us how Information relevant to these Decisions is obtained. Nasiri et al [15] propose to link Key Performance Indicators, KPIs, with goal orientation. A KPI is used as a way to indicate goal achievement. Thereafter, techniques as in goal orientation or obtaining facts and dimensions for measuring goal achievement (as brought out above) are used.

The BEAM* approach [16] gives prominence to business events that comprise a business process. Each business event is represented as a table, and the RE problem now is to identify the table attributes. This is done by using the 7W framework that provides for asking questions of seven types, namely (1) Who is involved in the event? (2) What did they do? To what is done? (3) When did it happen? (4) Where did it take place? (5) Why did it happen? (6) HoW did it happen—in what manner? (7) HoW many or much was recorded—how can it be measured? Out of these, the first six supply dimensions, whereas the last one supplies facts.

From the foregoing, we see that there is a clear attempt to obtain the organizational context in which facts and dimensions carry meaning. This context is explored through a variety of concepts like goals, Decisions, business processes, business events, and KPIs. Once this is done, attention turns to obtaining data warehouse Information. The techniques for this second part are summarized in the table below.

Approach	Obtaining multi-dimensional model
Boehnlein and Ulbricht	Business objects and attributes relevant to analysis
	Edges of SERM schema
Bonifati	Quality focus
	Variation Factors
Prakash and Gosain	Information scenarios
Mazón et al.	Measure
	Context
Georgini et al.	Goal achievement measures
	Dimensions from leaves of goal hierarchy
Nasiri et al.	Follows Mazón et al.
Corr and Stagnitto	7W framework

The chief difficulty with Boehnlein and Ulbricht is the absence of any model or guideline to discover the attributes relevant to the analysis of interest. Indeed, the authors do not tell us how stakeholders articulate the analysis to be performed. In the absence of this, attribute identification becomes an unfocused activity. Further, as the authors themselves state, the approach is for obtaining "nominal" Information for the company as a whole. Therefore, individual stakeholder's Information needs are de-emphasized.

Bonifati relies on quality focus and variation Factors. Merely asking "how quality focus can be detailed" and "what Factors can influence quality focus" is, we believe, not enough. We need some structure, some models, around which the investigation could be made. This is necessary to provide guidance and direction in the task.

The structure of queries in Information scenarios of Prakash and Gosain is SQL-like, but there is no guidance on what Information to ask for and what Factors to consider. Thus, the approach relies heavily on the experience of the scenario writer.

Obtaining measures and contexts for Information goals as in Mazón et al. relies on determining what is relevant to the analysis to be performed. This is again an ad hoc activity and relies completely on the experience of the stakeholder. Similarly, we have no guidance in Georgini et al. on how to analyze leaf goals and what aspects to consider in arriving at dimensions.

Finally, the 7W framework used in Corr and Stagnitto is, we believe, rather simplistic. Compared to this, the other techniques discussed here at least provide some structure (quality, measure, context, etc.), for obtaining Information needs.

We surmise that there is a need to develop Information elicitation techniques that can be systematically deployed to elicit Information needs. As already mentioned in Introduction, the developed techniques should take into account manager concerns so as to obtain their buy-in. Further, since each technique addresses a different concern, the set of techniques developed form a collection that reduces the risk of missing Information requirements.

We emphasize that our work addresses the Information elicitation part, of DWRE. In other words, our technique comes into play once the first part has yielded the Decisions of interest. Therefore, the technique is neutral to the manner in which these Decisions are arrived at: whether through [10, 11, 20] or any other. Further, our proposals are also neutral to the origin of these Decisions. They may originate from Decision making for operational, policy enforcement [21], or policy formulation systems [22].

3 Manager Factors

We identify the important Factors by considering the role of Decision making in an organization. Interest of a Decision maker is in determining the gap between the current situation of an organization and the expected situation. The former is obtained by keeping a trace of organizational activities, and this trace is obtained from On Line Transaction Processing, OLTP, systems that keep track of the transactions performed.

The expected situation, on the other hand, lies in the intentions of managers: What does a manager want to achieve. First and foremost, a manager must be able to meet the goals set for him. Further, this must be done efficiently and effectively. Having taken a Decision that contributes to these broad objectives, the manager should be able to assess the impact of the Decisions, and this assessment may form the basis of subsequent Decision making. We propose four Factors, one each for these four issues. This is summarized in Table 1.

We consider each row of Table 1 in turn.

3.1 Critical success Factors

Bullen and Rockart [23] consider a critical success Factor (CSF) as a key area of work in which success is essential for a manager to meet his goals. A manager should have full Information to determine whether work is proceeding well in the area. It has been pointed out that most managers have only a few critical success Factors; typically 4–8 [23, 24] lays down an interviewing technique for eliciting CSFs.

Our interest is not in defining the CSFs of a manager. The technique of [23] allows CSF definition to be carried out. Instead, given already defined CSFs, we are interested in obtaining Information for estimating CSF satisfaction and therefore in defining an elicitation technique for this Information.

Our use of CSF for Information elicitation has the following benefits:

- It is relevant to manager concerns. Therefore, there is likely to be strong engagement of the manager with the requirements engineer.

Table 1 Managerial issues and associated Factors

Driving force	Relevant Factor
Success	Critical success Factors
Effective	Ends achievement
Efficient	Means efficiency
Changing environment	Outcome feedback

- The DWRE task would be manageable because there is a limited number of CSFs per manager.

3.2 Ends achievement

Ends achievement can be considered in two different ways, depending upon the way one conceptualizes the notion of Ends. These are as follows:

1. An End is a statement about what is to be achieved, a goal. In this view, one can do Ends analysis by asking which Ends contribute to the achievement of which other Ends. When this is applied recursively, we obtain an Ends hierarchy. One technique used is *Means–Ends analysis*. In this, the problem solver begins by envisioning the End, or ultimate goal, and then determines the best strategy for attaining the goal in his current situation. A Means–Ends hierarchy is built in which nodes at a certain level are goals and those at the next lower level are Means of achieving it. Means–Ends analysis is recursively applied till the leaves of the hierarchy are reached.
   Notice that an End is different from a CSF in that the latter is a work area where success is critical, whereas as End is that which is to be achieved.
2. The second view of Ends achievement views an End as the result achieved by performing a task or as the intended result of a Decision. When compared with view (1) above, one does not ask which End achieves a given End. Instead, one asks what Information is needed to ensure the effectiveness of the End. In other words, Ends analysis here is the identification of Information needed to evaluate the effectiveness of the End. We refer to it as *ENDSI* elicitation.
   Again, there is a difference between the notion of a CSF and this view of Ends. Whereas a CSF is about success in a critical work area, an End is the expected result of a Decision. A CSF is a more "macro" issue, whereas an End is relatively more focused and is at a "micro-level."

Since our interest is in determining Information, we adopt the second view. In our context, "Ends" refers to the result achieved by a Decision. The Decision maker/requirements engineer interaction is centered round determining the Information for the effectiveness of the result. Therefore, the manager considers only those Decisions that contribute positively to Ends effectiveness. Again, we see that this ensures that the Ends Effectiveness technique is close to the manager's view of a business and that it directly relates to Decisions for promoting Ends effectiveness.

3.3 Means efficiency

Broadly speaking, a Means is a way of achieving the Ends. When considering Ends achievement in Sect. 3.2, we have mentioned the use of Means–Ends analysis. When applying this, a lower level in the hierarchy is the Means of achieving the immediately higher level. Both levels describe the same system, but in different terms.

There is yet another way of looking at Means. This view treats a Means as a first-class concept of the business world. A Means is of direct interest in the business world, just as an End is or a CSF is. It is the instrument, the process, activity, or task deployed to achieve an End. The interesting question for a manager is the efficiency of the deployed Means. Thus, Means Efficiency deals with identification of Information for evaluating the efficiency of the Means. We refer to obtaining this Information about Means as *MEANSI* elicitation.

Again, notice the Means Efficiency technique is close to the manager's view of the business and that it directly relates to Decisions for Means selection.

3.4 Feedback analysis

Studies in the area of dynamic Decision making have brought out the important role that feedback plays in the Decision-making task. Sterman [25] noted that the effect of a Decision is a change in the environment. The environmental changes alter the conditions of choice and eventually feed back into the Decision. A feedback cycle is thus formed. The example given in Sterman is that of a Decision to increase production. This changes the price, profits, and demand of goods; the labor and materials market may be affected; customers may also react. All these affect future production Decisions.

We interpret this feedback loop in terms of Information. Information about each element in the feedback loop is to be made available to the manager to take future production Decisions. Thus, for example, changes in price, profits and so on, are to be kept track of.

3.5 Summary

We believe that there are at least four major driving forces of a manager, namely (i) the manager must be seen as "successful," (ii) the results delivered must be "effective" and beneficial to the organization (iii) the manager should be seen to be efficient, and (iv) the manager should cater to the changing environment of the business.

As we see it, a manager shall be motivated to *take those Decisions that result in maximization of the achievement parameters*. Therefore, the data warehouse should keep Information to estimate the achievement parameters for every manager. This belief forms the basis of our elicitation techniques.

4 The Decision requirement model

We base our Information elicitation technique on the Decision Requirement model. This model captures our view of the structure of a Decision and of Information as well as the relationship between these two. In this model, the Information that is relevant to a Decision is modeled as a Decision requirement. Thus, as shown in Fig. 1, a ecision requirement, textually written as ⟨Decision, Information⟩, is an aggregation of Decision and Information. This relationship is N:M since a Decision may have more than one piece of Information associated with it and a given piece of Information may be relevant to more than one Decision, D.

4.1 The notion of a Decision

The basic property of a Decision is that it is a member of a choice set. We model this in Fig. 1 by defining a relationship, *is member of*, between choice set and Decision. A Decision can be a member in more than one choice set, and a choice set can contain more than one Decision. Therefore, *is member of* is an N:M relationship as shown. A

Fig. 1 Basic Decision requirement meta-model
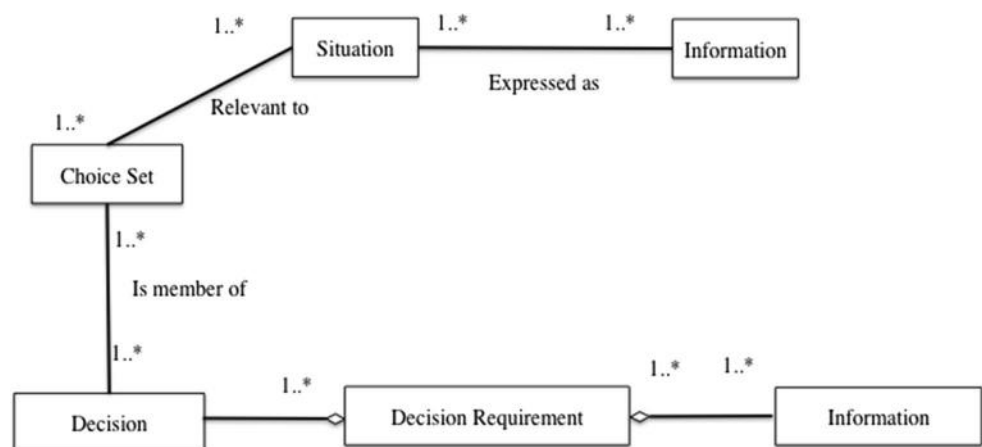
choice set is associated with a situation by the relationship, *relevant to*. A situation is found in the organization and may be a trace of what the organization has been doing, or it may indicate what is happening in the organization.

As an example of a situation, consider a health service that has a *rush of patients*. This situation says that a large number of patients were admitted and a similar number were turned away. To handle this situation, we associate a choice set with *rush of patients*. This choice set, *Reduce patient rush* = {register patients online, increase medical staff}. The first reduces the physical rush of patients on site, whereas the second enables the handling of a larger number of patients. Figure 1 shows that a choice set may be relevant to more than one situation. Evidently, our choice set is also relevant to the situation, *Improve medical service*.

The choice set, *Reduce patient rush* consists of two Decisions. This illustrates the 1:N relationship between a choice set and its member Decisions. Notice that increase medical staff, a member of our choice set, may itself be found as a member in another choice set that handles the situation, *Improve medical service*. Thus, we have a 1:M relationship between a Decision and more than one choice set. Taken together, we have the M:N relationship *member of* between choice set and Decision as shown in Fig. 1.

We define two constraints on a choice set, namely coherence and cardinality constraints. *Coherence* says that all elements of a choice set must achieve the same purpose. For example, consider the choice set, CSET = {Increase bed count, Optimize bed use, Increase units} for our health service that wants to handle its rush of patients. All elements of this set have the same intention "handle rush of patients." Such a choice set is coherent. As an example of an incoherent choice set, consider CSET1 = {Increase bed count, Optimize bed use, Open research unit}. The element, Open research unit, does not help in achieving the intention. Therefore, CSET1 is not coherent.

Cardinality of the choice set says that the number of elements in a choice set must be equal to or greater than two. Clearly, the choice set is undefined if its cardinality is zero. If this cardinality is unity, then there is exactly one way of achieving the ecision and there is no decisional problem. Since, we are concerned with providing Decision support in the data warehouse context, the cardinality of the choice set should be greater than unity. It is only in this case that the Decision maker needs to analyze the existing situation, refer to relevant Information, and use judgment to select the most appropriate element.

Figure 1 establishes a relationship between the situation and Information. This M:N relationship says that a situation is *expressed as* one or more pieces of Information and that a piece of Information may form part of more than one situation. We now consider the notion of Information in detail.

4.2 Information

The right hand side of Fig. 1 shows that Information is required to take a Decision. Our Information model is shown in Fig. 2, and we consider this model here. Let there be a set of Decisions $D = \{D_1, D_2\ldots, D_n\}$. Each D_i, $1 \leq i \leq n$, has its own choice set and participates in its own Decision requirements. Due to this participation, we can associate, with the corresponding Decisions D, sets of Information I_1, $I_2\ldots, I_n$. Then, the set of relevant Information to D, represented as Information in Fig. 2, is defined as the union of these Information sets:

$$Information = I_1 \cup I_2 \ldots \cup I_n$$
$$= \{I \text{ such that I belongs to } I_k,\ k \text{ between 1 and } n\}$$

We shall refer to I as an instance, member or element of Information interchangeably.

Now three kinds of Information are relevant to data warehousing [26, 27], detailed Information which is at the lowest level of granularity, summarized or aggregated Information, that is obtained from other detailed/aggregated Information, and historical Information that may be the history of detailed Information or of aggregated Information. This Information has its own dimensions.

Figure 2 introduces the corresponding typology of Information. Detailed Information is at the lowest level of granularity. It is raw unprocessed Information that has not been obtained, for example, through a computation procedure. Aggregate Information is obtained by computing from other Information that may itself be detailed, aggregate, or historical. This is shown in Fig. 2, by the specialization of Information into *Detailed* and *Aggregate* as well as by the "Computed from" relationship between Aggregate and Information. *Historical* Information shown in Fig. 2 has two properties, period and temporal unit. The former tells us the duration of the history, whereas the temporal unit tells us the time unit, month, year, etc., of the duration.

Figure 2 introduces the notion of composition of Information. A composition is a logically related association of Information that carries meaning. There are two kinds of compositions, namely reports and comparisons. A report is a collection of detailed, aggregate, historical Information as well as of comparisons. A comparison is a special kind of collection that, for example, (a) contains rankings (top ten, bottom ten) or (b) brings out the similarities/differences between pieces of Information.

Information can have many attributes, and an attribute can be a property of more than one instance of Information. This is the N:M relationship between attribute and Information of Fig. 2.

The Information model introduces the notion of dimension by defining a relationship, *Categorized by*. This
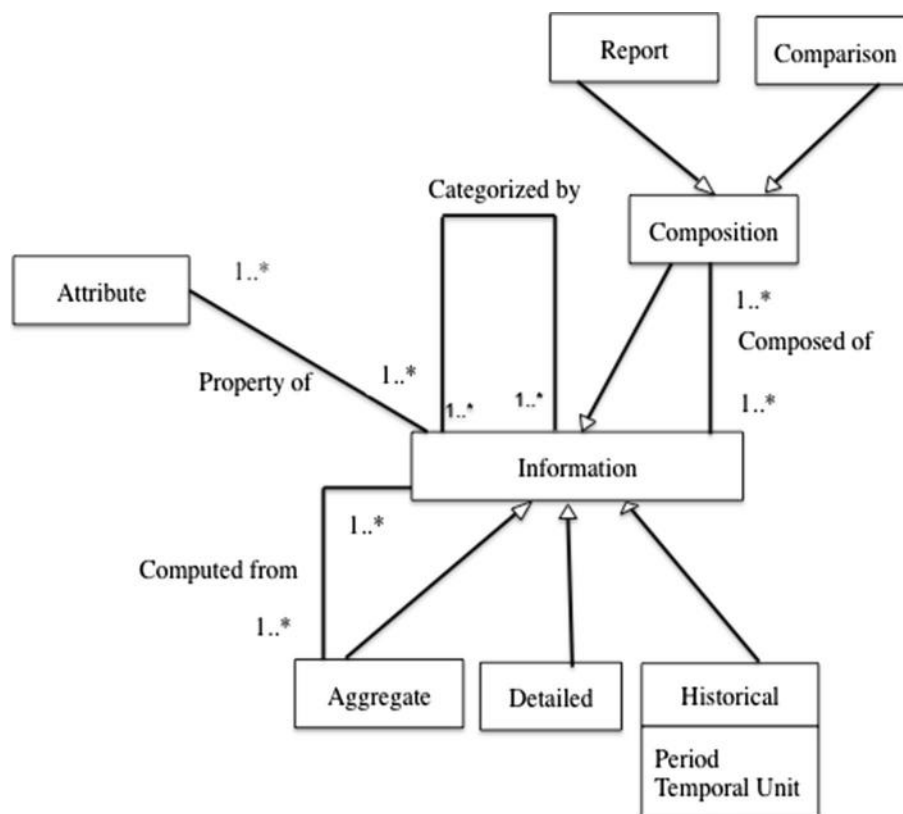
relationship says that Information can be categorized by other Information. Thus, we get sales by season, where sales and season belong to Information. As shown, this relationship is N:M.

5 Eliciting Decision requirements

In eliciting the required Information, we propose an elicitation technique for each Factor. Thus, we get four elicitation mechanisms, CSFI, ENDSI, MEANSI, and Outcome feedback.

As already mentioned, our Information elicitation approach is neutral to where Decisions come from. Thus, in eliciting Decision requirements, we assume a given set of Decisions, $D = \{D_1, D_2, …, D_n\}$.

We define the problem of eliciting Decision requirements as "for all Decisions D_i, where $1 \leq i \leq n$, elicit the set of Information relevant to D_i." The Information for the entire set D must be available in the data-warehouse-to-be.

Since a manager takes those Decisions that maximize achievement parameters, the manager has knowledge of which Decision affects which achievement parameter. Thus, our elicitation technique obtains the relevant achievement parameter from the manager for Di, and, thereafter, goes on to elicit the required Information for estimating it. The Information to be elicited is according to the model of Fig. 2.

In the rest of this section, we show our elicitation process and tool. In this paper, we provide only a flavor of the elicitation process and refer the reader to the detailed description of the tool that can be found in [28].

5.1 CSFI elicitation

The CSFI elicitation technique obtains Information required to assess progress in critical work areas. The essential question here is to identify the Variables that must be monitored to ensure that these Factors remain in control. This control is carried out by appropriate Decision making.

Table 2 shows the essence of the CSFI technique. In the first two columns, the Decision and the associated CSF are tabulated. (Recall that we assume that the technique of [23] has been applied to obtain the CSF.) The third column contains the Variables that go into assessing the CSF. Finally, the last column contains Information relevant to the Variables.

The example presented in Table 2 is for the Decision of adding a new pharmacy in the health service. The CSF associated with it is medicine delivery since it is a critical work area in the service. One Variable that helps in assessing the CSF is the waiting time of patients at the pharmacy. The Information needed for this Variable is the average waiting time categorized by patient type, and a weekly record of this Information needs to be kept for a 10-week duration.

Table 2 Obtaining Information from CSFI

Decision	CSF	CSF Variable	Information
Add new pharmacy	Medicine delivery	Waiting time of patient	**Aggregate**: average waiting time **Category**: patient type History: *Time unit:* week *Duration* 10 weeks

Note that, in general, there may be more than one Variable for a given CSF. However, we have exemplified our technique in Table 2 with one Variable.

Thus, CSFI elicitation is a three step process consisting of (a) CSF association with a Decision, (b) determination of CSF Variables, and (c) determination of Information in accordance with the model of Fig. 2. The tool interface for CSFI elicitation is shown in Fig. 3.

The top of the screen shows that Information for the Decision Add new pharmacy is being elicited. The left hand side of the screen allows the requirements engineer to either select an existing CSF from a displayed list of CSFs or enter a new CSF. The figure shows that the CSF, Medicine delivery, has been selected. The rest of the screen shows the Variable, waiting time for patients, and the corresponding Information as given in Table 2.

5.2 ENDSI elicitation

Recall that "Ends" refers to the result achieved by a Decision. ENDSI elicitation is the identification of Information needed to evaluate the effectiveness of the End to be achieved. The requirements engineering task is that of determining the Variables and Information of interest in estimating this effectiveness.

Table 3 shows the four aspects of ENDSI elicitation. In the first two columns, the End and the Decision with which it is associated are tabulated. The third column contains the Variables that go into assessing the effectiveness of the End. Finally, the last column contains Information relevant to the Variables.

We continue in Table 3 with the example for the Decision of adding a new pharmacy. The End associated with it is Full Utilization. An effectiveness Variable that helps in assessing the effectiveness of the End is the service provided. The

Fig. 3 The CSFI interface

Information needed for this Variable is the total sales medicine-wise. The second row shows additional Information, the number of transactions during each shift.

As for CSFI, there can be more than one effectiveness Variable per End and there can be many Ends for a Decision.

Since we have already shown the flavor of the user interface for the CSF, we do not repeat it here.

5.3 MEANSI elicitation

Recall that Means Information elicitation is the identification of Information needed to evaluate the efficiency of the Means adopted to produce the Ends. Thus, the requirements engineer/stakeholder interaction is now centered round eliciting Variables that provide Information on the efficiency of the Means adopted for each Decision.

We can again understand MEANSI elicitation through the four-column Table 4. As before, the first two columns associate the Means with a Decision. Thereafter, the efficiency of the Means is captured in a Variable, and finally, the Information is obtained.

The example in Table 4 is for the same Decision, Add New Pharmacy. The Means is to start completely afresh and not reuse an existing building. The efficiency Variables are the resources, civil, electrical, fixtures and furniture, etc., that shall be used. The Information needed is the cost for each resource. The second row of the table shows that efficiency

can be estimated as the time to set up the new pharmacy, and the total start-up time is the Information to be maintained.

As before, we do not show the user interface.

5.4 Feedback Information elicitation

Interest in feedback Information, FI, elicitation is in determining each element that shall be impacted by a Decision and the Information that should be maintained to study this impact. There are three aspects of interest as shown in Table 5, the Decision–feedback element association and the Information to be kept.

Consider, again, the Decision Add new pharmacy. This changes the perception about our health service, resulting in an increase in the registered patients of the health service, which may lead to requirement of additional medical staff that in turn affects the pharmacies of the service. Thus, we find a feedback cycle that starts off from the outcome of the Decision, goes through the organization, and returns back to the outcome.

Table 5 shows the feedback Variables and the Information required to study the impact.

5.5 The global elicitation process

Each of the techniques described above has its own elicitation process consisting of two or three steps as described. That is, the micro-level guided process is as explained above.

Table 3 Information obtained from ENDSI elicitation

Decision	End	End effectiveness Variable	Information
Add new pharmacy	Profitability	Service provided	**Aggregate:** total sales **Category:** medicine-wise
			Aggregate: number of transactions **Category:** shift-wise

Table 4 Obtaining Information from MEANSI elicitation

Decision	Means	Means efficiency Variable	Information
Add new pharmacy	Establish afresh	Resources used	Estimated cost Category: resource-wise
		Time	Setting up time

Table 5 Means Information

Decision	Feedback Variable	Information
Add new pharmacy	Increase in patients	**Aggregate:** number of patients **Category:** speciality-wise **History:** Time unit: month Duration: 12 months
	Additional medical staff	**Aggregate:** additional doctors **Category:** speciality-wise

However, as mentioned in Introduction, we believe that the use of multiple elicitation techniques, corresponding to the Factors of interest, shall be beneficial.

Our multifactor elicitation process expects that one or more, possibly all, of the four techniques shall be applied to every Decision in the input set of Decisions. In other words, the input to the elicitation process is the set of Decisions and, for each Decision belonging to this set, the requirements engineer determines the relevance of each Factor to the Decision. Thereafter, the corresponding elicitation technique is applied and the process is repeated for all Decisions in the input set.

There are two aspects to this process that are interesting. First, the stakeholder is guided to examine the relevance of all the Factors and perform complete requirements analysis. Second, if desired, the stakeholder can choose the Factors considered important and leave out the irrelevant ones.

5.6 The repository

The repository supporting the elicitation tool is in three parts, a Decision part, a Factor and Variable part, and the Information part. These three parts are related to one another (see Fig. 4).

The Decision base contains the Decisions; Factors and the Variables are available in the Factor and Variable base; the Information base contains all relevant Information for the Variable, namely aggregate, category, etc., as required by the Information model. The relationships between the three bases are as shown. A Decision affects one or more Factors/Variables, and a Factor/Variable may be affected by more than one Decision. Thus, there is an M:N relationship between these. Similarly, there is an M:N relationship between Decision and Information. Finally, Information is

used to assess Factors/Variables. Again, there is an M:N relationship between these.

We can provide traceability of Information in three ways. Information can be traced

(a) Directly to Decision
(b) Directly to Factor/Variable
(c) Transitively to Decision via Factor/Variable

5.7 Information structuring

Having obtained the required Information, we are now left with the task of converting it into a multi-dimensional form. The authors of [21] distinguish the elicited form from the multi-dimensional, by referring to the former as "early" Information.

The approach to convert early Information to structured form has been elaborated in [21]. The basic idea is to represent early Information into ER form and then use existing semiautomated techniques like those of Golfarelli [29], Moody and Kortink [30], and Hüsemann et al. [31] to obtain the star schema. An illustration of the use of the technique is given in [21].

6 Comparison

The four techniques used for requirements engineering here have their origins in the area of MIS. There were two concerns in MIS, Information systems planning and Information requirements analysis. Since our concern here is only with the latter, we look at the MIS experience with requirements analysis and relate it to our proposals.

Fig. 4 The structure of the repository
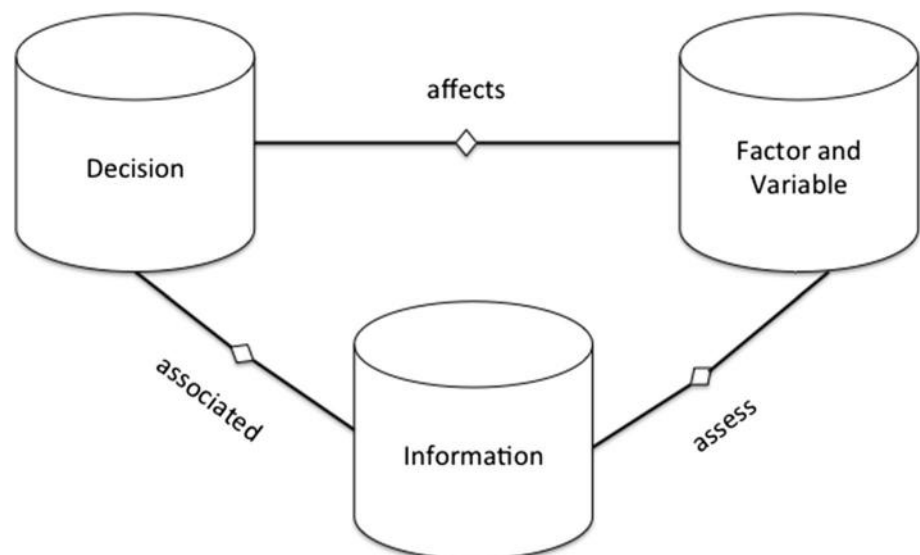
An MIS was oriented toward producing fixed reports at prescheduled times from the transactional data that were available. Reports could be for stakeholders who were individuals, departments, and entire organizations. Thus, requirements analysis in MIS was for obtaining the Information needed by each stakeholder so that relevant fixed reports could be prepared.

Boynton and Zmud [32] found that CSF analysis works well for higher-level managers, but not for others. This latter category is more concerned with day-to-day events within their area of responsibility rather than with a conceptual orientation of their environment and that of the organization. These authors opine that a successful requirements analysis effort would need supplementing CSF with other "more concrete" techniques.

Wetherbe and Davis [24] used multiple analysis techniques, namely BSP of IBM, CSF analysis, Ends analysis, and Means analysis. They found that BSP was useful in MIS planning, whereas the others were applicable to requirements analysis. Using multiple techniques was beneficial to (a) cater to manager preferences, (b) determine requirements in addition to those determined by any one method, and (c) address different cognitive levels of managers.

In determining Information requirements, MIS did not make explicit the role of the Decision-making task entrusted to the manager. This was implicit: Perhaps stakeholders would not only take into account the Factors that they were responsible for but would also consider the Decision-making task entrusted to them. The relationship between this job and the Factors was not explored.

In contrast, we have treated a Decision as a first-class concept of Decisional requirements engineering. It is explicitly linked to the Factors that it affects, either positively or negatively. Therefore, in making the Decision–Factor relationship explicit, we get

- Guidance in the task of eliciting Information requirements: For each Decision, all the techniques are used and the two- /three-step process outlined earlier is followed.
- The association between a Decision and the Information relevant to it, as well as the Information–Factor association.
- Traceability of Information back from Information to the Factor(s) that produced it and on to the Decision for which it is relevant.

Notice that the observations made in the area of MIS continue to apply. Thus, by extending a range of techniques to DWRE, we get all the advantages that Wetherbe and Davis [24] obtained.

7 Experience

We have applied our multi-analysis approach to a traditional medical system offering treatments in Ayurvedic medicine, Yoga, Unani and Naturopathy, AYUSH. In this system, we, along with a domain expert, elicited Information from three different Decision-making environments, operational Decision making reported in [33], Decision making for policy enforcement rules [21], and Decision making for policy formulation [22]. In all these environments, Decisions were first determined from the business context and the proposed elicitation techniques were applied to yield the required Information for these.

The main idea was to study whether

(a) Our elicitation techniques were applied equally well to each Decision-making environment or whether there were variations,
(b) New Information was obtained by using multiple elicitation techniques or not, and
(c) The same Decision present in the three different environments requires different Information or it does require the same Information.

Our observations are as follows.

7.1 Applicability of Information elicitation techniques

7.1.1 Applicability of Information elicitation techniques at policy formulation layer

Policy formulation is done at very high levels in the organization. There is little concern here with operational-level Decision making. First, consider the place of CSFI. Policy formulation is done by senior-most positions in management. The critical success Factors of such managers are strategic in nature and are closely affected by the policies that the organization adopts. Therefore, we found CSFI analysis to be a good source of Information for these positions.

Consider a policy Decision that "Degree of doctor, in every AYUSH hospital, must be MD in respective field." Decision to be taken is whether to {select, modify, delete} the components of this policy. CSFI analysis for "*select degree of doctor must be MD*" is given in Table 6. For reasons of space, we show only one piece of Information.

We would expect high concern with attainment of Ends. Indeed, we found that ENDSI analysis yielded Information for every policy Decision. For our example, ENDSI analysis has been shown below. One of the Ends of deciding to have doctors with MD degree is that the hospital will have staff in specialized fields. The effectiveness of this End is Service provided, and required Information is elicited. Again,

we show only one piece of aggregate Information here in Table 7.

We find that issues of the Means to be adopted are not of prime concern when formulating policies. For our Decision *Select* "degree of doctor must be MD," the Means by which this is achieved requires the formulation of another policy for which Decision to {select, modify, delete} is taken.

This result is in consonance with that in MIS. Policy issues are of relevance for high-level management and require good conceptualization skills. Effectiveness and critical work areas are the dominant Factors here.

7.1.2 Applicability of Information elicitation techniques at policy enforcement rule formulation layer

Policy enforcement rules lie between policy formulation and operational Decision making. The rules are formulated, to enforce policies, as actions in the WHEN-IF-THEN form. Decision makers at this level are highly influenced by Ends to be achieved and Means to measure the efficiency of the

End. Thus, we found ENDSI and MEANSI analyses useful to elicit Information.

Consider a Decision "Re-designate hospital." ENDSI analysis and MEANSI analysis are given in Table 8(a) and (b), respectively.

Now, the role of CSFI analysis for those managers who formulate policies enforcement rules is not so clear. We found that CSF is not applicable to all Decisions at this layer. Notice that the level of conceptualization required now is lower than that for deciding on policies.

For our example "Re-designate hospital," the domain expert found no applicable CSF during CSFI analysis. However, for the Decision "Add new pharmacy" considered in Sect. 5, CSFI analysis did in fact help elicit Information.

7.1.3 Applicability of Information elicitation techniques at the operational layer

Finally, for operational Decision making, CSFI, MEANSI, and ENDSI analyses are highly important. It is possible to find Information using all these elicitation techniques. Since

Table 6 CSFI analysis for Decision Select "degree of doctor must be MD"

Decision	CSF	CSF Variable	Information
Select "degree of doctor must be MD"	High-quality care	Patient health	**Aggregate:** survival rate of patients **History:** *Time unit:* monthly *Duration*: 2 years

Table 7 ENDSI analysis for Decision Select "degree of doctor must be MD"

Decision	End	End effectiveness Variable	Information
Select "degree of doctor must be MD"	Staff with focused experience	Service provided	**Aggregate:** total incoming referrals **Category:** specialty type **History:** *Time unit*: monthly *Duration*: 2 years

Table 8 (a) ENDSI analysis for Decision "Re-designate hospital," (b) MEANSI analysis for Decision "Re-designate hospital"

Decision	End	End effectiveness Variable	Information
(a)			
Re-designate hospital	Maximize economic return	Revenue generated	**Aggregate:** total consultation fee **Category:** specialty-wise **History:** *Time unit*: weekly *Duration*: 1 year

Decision	Means	Means efficiency Variable	Information
(b)			
Re-designate hospital	Select another specialty	Expertise needed	Number of diseases **Category**: specialty-wise

the foregoing gives a flavor of the application of our techniques, we do not show sample data elicited for an operational Decision.

7.2 Eliciting new Information by using different elicitation techniques

Tables 6, 7, and 8 show that when more than one elicitation technique is applicable to a Decision-making environment then use of one technique may discover Information not discovered by the other. This is the case in Tables 6 and 7 where CSFI and ENDSI applied to the same Decision yield different Information. This is also the case with Table 8(a) and (b) where the application of ENDSI and MEANSI produce different Information.

Indeed, our experience is that it was in very rare cases that multiple techniques yielded the same Information for the same Decision of a Decision environment.

7.3 Common Decisions across layers generate different Information

We found that Decisions can be common across the three levels of Decision making. Further, CSFs, Ends to be achieved, and Means to achieve it vary with the level of Decision making. Since the CSF, Ends, and Means are different, CSFI, ENDSI, and MEANSI analyses yields different Information for the same Decision.

As an example, consider the MEANSI analysis for the Decision "Expand private ward." Table 9(a) shows Information elicited at the *policy enforcement rule* layer and Table 9(b) for *operational level* of Decision making. Notice in the former the Means considered is "Remodel room." In contrast, at the operational level given in Table 9(b), the same Decision is looked at in terms of the actual construction to be performed. This difference between the abstraction levels of the Means results in a completely different perspective. The Information needed for the two perspectives is quite different as shown in Table 9(a) and (b).

8 Conclusion

The need for DWRE arises because requirements engineering techniques for transactional systems focus on determining system functionality and therefore make little or no effort in determining Information content of systems. However, interest in data warehousing is on the Information needed to support Decision making. DWRE approaches that have been proposed in the past bring this Information perspective into requirements engineering. Consequently, concepts of business events, goals, Decisions, KPIs, etc., have been associated with the notion of "relevant Information." We have shown that the chief drawback of DWRE approaches is that they do not provide any support for eliciting this Information.

In developing a technique that provides this support, the first issue is that of creating and sustaining enough interest in stakeholders to participate in the requirements engineering process. We create this stakeholder buy-in by determining important managerial Factors and developing elicitation techniques for each of these Factors. The second issue is as to how these techniques shall be used, and we propose here to treat these as a suite of techniques to be used collectively. This minimizes the possibility of missing Information requirements and covers all areas of managerial concern.

Our proposals are for eliciting relevant Information of each Decision. Thus, our four elicitation techniques are applied to each Decision of interest. The manner in which these Decisions were discovered and where they come from are therefore outside the purview of the Information elicitation techniques proposed here.

As for the origin of Decisions, three different sets of Decisions were obtained, one each for Decision making for

Table 9 (a) MEANSI analysis at policy enforcement rule layer for Decision "Expand private ward," (b) ENDSI analysis at operational layer for Decision "Expand private ward"

Decision	Means	Means efficiency Variable	Information
(a)			
Expand private ward	Remodel room	Resources needed	Space required Estimated cost **Category**: Labor-wise Material-wise
Decision	Means	Means efficiency Variable	Information
(b)			
Expand private ward	Construct extension	Cost incurred	Construction cost per sq. foot New material cost
	Break barrier	Cost incurred	Breaking cost per sq. foot

operations, policy enforcement, and policy formulation. The notion of a Decision requirement then led us to the application of the proposed elicitation techniques. Thus, our techniques are independent of the source of Decisions.

Regarding the manner in which Decisions were obtained, we have deployed a variety of methods for arriving at Decisions:

1. For policy Decisions, we adopted reusability of existing policies, constructed a policy hierarchy for each policy, and obtained Decisions from the nodes of the hierarchy. These Decisions formed the basis of Information elicitation.
2. For policy enforcement, we formulated rules that were applied to policies to yield policy enforcement rules. This resulted in Decisions to select or reject policy enforcement rules.
3. For operational Decision making, we again looked at each policy enforcement rule and then derived operational Decisions from these using yet another set of rules.

The structure of the repository of our elicitation tool allows tracing back from the elicited Information to the Decision(s) from which the Information originated. Additionally, it is possible to go forward from a Decision to determine the Information relevant to it. We expect to exploit this in future to deal with evolution of requirements of data warehouses.

References

1. Antón AI (1996) Goal-based requirements analysis. In: Proceedings of the second international conference on requirements engineering. IEEE, pp 136–144
2. Lamsweerde A (2000) Requirements engineering in the year 00: a research perspective. In: Proceedings of the 22nd international conference on Software engineering. ACM, pp 5–19
3. Sutcliffe AG, Maiden NA, Minocha S, Manuel D (1998) Supporting scenario-based requirements engineering. IEEE Trans Softw Eng 24(12):1072–1088
4. Lamsweerde A, Willemet L (1998) Inferring declarative requirements specifications from operational scenarios. IEEE Trans Softw Eng 24(12):1089–1114
5. CREWS Team (1998) The CREWS glossary, CREWS report 98-1. http://SUNSITE.informatik.rwth-aachen.de/CREWS/reports.htm
6. Liu L, Yu E (2004) Designing information systems in social context: a goal and scenario modelling approach. Inf Syst 29(2):187–203
7. Boehnlein M, Ulbrich vom Ende A (1999) Deriving initial data warehouse structures from the conceptual data models of the underlying operational information systems. In: Proceedings of workshop on data warehousing and OLAP. ACM, pp 15–21
8. Boehnlein M, Ulbrich vom Ende A (2000) Business process oriented development of data warehouse structures. In: Proceedings of data warehousing 2000. PhysicaVerlag HD, pp 3–21
9. Bonifati A, Cattaneo F, Ceris S, Fuggetta A, Paraboschi S (2001) Designing data marts for data warehouses. ACM Trans Softw Eng Methodol 10(4):452–483
10. Prakash N, Gosain A (2003) Requirements driven data warehouse development. In: CAiSE short paper proceedings, pp 13–17
11. Prakash N, Gosain A (2008) An approach to engineering the requirements of data warehouses. Requir Eng J 13(1):49–72
12. Mazón JN, Pardillo J, Trujillo J (2007) A model-driven goal-oriented requirement engineering approach for data warehouses. In: Hainaut JL et al (eds) Advances in conceptual modeling—foundations and applications. ER 2007. Lecture notes in computer science, vol 4802. Springer, Berlin, pp 255–264
13. Giorgini P, Rizzi S, Garzetti M (2008) GRAnD: a goal-oriented approach to requirement analysis in data warehouses. Decis Support Syst 45(1):4–21
14. Prakash N, Bhardwaj H (2014) Functionality for business indicators in data warehouse requirements engineering. In: Parsons J, Chiu D (eds) Advances in conceptual modeling. ER 2013. Lecture notes in computer science, vol 8697. Springer, Cham, pp 39–48
15. Nasiri A, Wrembel R, Zimányi E. Model-based requirements engineering for data warehouses: from multidimensional modelling to KPI monitoring. In: International conference on conceptual modeling 2015. Springer, Berlin, pp 198–209
16. Corr L, Stagnitto J (2012) Agile data warehouse design. DecisionOne Press, Leeds
17. Cravero Leal A, Mazón JN, Trujillo J (2013) A business-oriented approach to data warehouse development. Ingeniería e Investigación 33(1):59–65
18. Gosain A, Bhati R (2016) Goal oriented approaches in data warehouse requirements engineering: a review. In: International conference on smart trends for information technology and computer communications. Springer, Berlin, pp 244–253
19. Bhardwaj H, Prakash N (2016) Eliciting and structuring business indicators in data warehouse requirements engineering. Expert Syst 33(4):405–413
20. Giorgini P, Rizzi S, Garzetti M (2005) Goal-oriented requirement analysis for data warehouse design. In: Proceedings of the 8th ACM international workshop on data warehousing and OLAP. ACM, pp 47–56
21. Prakash D, Gupta D (2014) Eliciting data warehouse contents for policy enforcement rules. Int J Inf Syst Model Des (IJISMD) 5(2):41–69
22. Prakash D, Prakash N (2015) Towards DW support for formulating policies. In: IFIP working conference on the practice of enterprise modeling. Springer, Cham, pp 374–388
23. Bullen CV, Rockart JF (1981) A primer of critical success factors, CISR No. 69 Sloan WP No. 1220-81 Center for Information Systems Research, Sloan School of Management Massachusetts Institute of Technology
24. Wetherbe JC, Davis GB (1983) Developing a long-range information architecture. In: Proceedings of AFIPS, pp 261–269
25. Sterman JD (1989) Modeling managerial behaviour: misperceptions of feedback in a dynamic decision making experiment. Manag Sci 35(3):321–339
26. Inmon B (2005) Building the data warehouse, 4th edn. Wiley, New York
27. Kimball R, Ross M (2002) The data warehouse toolkit: the complete guide to dimensional modeling. Wiley, New York
28. Prakash N, Prakash D, Sharma YK (2009) Towards better fitting data warehouse systems. In: Persson A, Stirna J (eds) The practice of enterprise modeling. PoEM 2009. Lecture notes in business information processing, vol 39. Springer, Berlin, pp 130–144
29. Golfarelli M, Maio D, Rizzi S (1998) Conceptual design of data warehouses from E/R schemes. In: Proceedings of the thirty-first Hawaii international conference on system sciences, 1998, vol 7. IEEE, pp 334–343

30. Moody LD, Kortink MAR (2000) From enterprise models to dimensional models: a methodology for data warehouses and data mart design. In: Proceedings of the international workshop on design and management of data warehouses, Stockholm, Sweden, pp 5.1–5.12

31. Hüsemann B, Lechtenbörger J, Vossen G (2000) Conceptual data warehouse design. In: Proceedings of the international workshop on design and management of data warehouses (DMDW2000) Stockholm, Sweden

32. Boynton AC, Zmud RW (1984) An assessment of critical success factors. Sloan Manag Rev 25(4):17–27

33. Prakash N, Prakash D, Gupta D (2011) Decisions and decision requirements for data warehouse systems. In: Information systems evolution, pp 92–107