Model-Based Requirements Engineering for Data Warehouses: From Multidimensional Modelling to KPI Monitoring

Azadeh Nasiri[1,2(✉)], Robert Wrembel[1], and Esteban Zimányi[2]

[1] Institute of Computing Science, Poznan University of Technology, Poznan, Poland
{azadeh.nasiri,robert.wrembel}@cs.put.poznan.pl, nazadeh@ulb.ac.be
[2] Department of Computer and Decision Engineering,
Université Libre de Bruxelles, Brussels, Belgium
ezimanyi@ulb.ac.be

Abstract. A Data Warehouse (DW) is one of the main components of every BI system. It has been convincingly argued that the success of BI projects can be strongly affected by the Requirements Engineering (RE) phase, when the requirements of a DW are captured. Multiple RE methods for DWs have been proposed which have goal models in the core of their approach. Existing methods cover RE up to the static part of a DW, where the Multidimensional (MD) model is obtained. However, the RE for the dynamic part of the DW, where the requirements of operations on the DW are captured, has been neglected in the literature. In this paper, we propose a RE method, covering both the static and the dynamic part of a DW in an integrated manner. Our approach is to use the concept of a Key Performance Indicator (KPI). We initially use KPIs as the main driver to obtain the MD model and then discuss how decision-makers analyse them in order to measure the success of an organisation. In our method, the goal model from the i* framework was extended with UML use case diagrams.

Keywords: Data warehouse · Requirements Engineering · Key performance indicators

1 Introduction

One of the main components of BI systems is a DW, which integrates data from different data sources and structures them to be used in analytical systems. DW systems are sometimes developed based on an incomplete and inconsistent set of requirements, causing many BI projects to fail [13,16]. This means that the success of BI projects can be strongly affected by the Requirements Engineering (RE) phase. In general, RE is defined as the process of discovering the needs of involved stakeholders and supporting those needs by modelling and documenting them in a form that is analysable and communicable. In order to support RE for DWs, multiple methods have been proposed in the literature, which are based mainly on the Goal-Oriented Requirements Engineering (GORE) approach

M.A. Jeusfeld and K. Karlapalem (Eds.): ER 2015 Workshops, LNCS 9382, pp. 198–209, 2015.
DOI: 10.1007/978-3-319-25747-1_20

[4,6,10–12]. This approach uses goals for eliciting, modelling, analysing, negotiating, and modifying requirements.

Most of the RE methods for DWs use GORE frameworks like i*, URN, and KAOS to capture strategic goals of an organisation, alternative decisions to achieve such goals, and required information to support the analysis needed for decision making. Eventually, information requirements are structured into facts (the focus of analysis) and dimensions (the context of analysis), which are the elements of the multidimensional (MD) model. Current research works cover RE up to obtaining the MD model of a DW [2–8,11,15]. The MD model refers to the static part of a DW. RE for the static part has received much attention while RE for the dynamic part, defined as the operations conducted on a DW, has been neglected in the literature.

To the best of our knowledge, there is no RE method in the literature that covers both the static part and the dynamic part of a DW. The goal of this paper is to develop a RE method for DWs covering both the static part of DWs, from where the MD model is obtained, as well as the dynamic part of the DW, where the requirements of operations on the DW are captured in a coherent and integrated manner. This integrated approach helps to align the data required to be stored in the DW with the analytics conducted over the data.

Our approach uses the concept of KPIs. KPIs are complex measurements used to monitor the performance of business processes and strategies in an organisation. KPIs are usually included in dashboards, providing a detailed view of each specific area of the organisation [14]. Based on how KPIs are calculated, they are called atomic or composite [1]. Atomic KPIs are those whose values are obtained from data sources. Composite KPIs are those whose values are obtained from other KPIs (called component KPIs). In the RE context, we use the concept of the composite KPI to evaluate the degree of fulfillment of a strategic goal, and the concept of the component KPI to evaluate the performance of various business processes.

Our method uses KPIs as the main driver to obtain the MD model and then discuss how decision-makers analyse KPIs in order to measure the success of an organisation. Our method has the following key features:

- We offer a model-based RE method for DWs, where models provide the basis of requirements artefacts.
- We apply the GORE approach to develop our RE method. The GORE approach is useful to represent how an intended BI system meets organisational goals [4]. Among the existing GORE frameworks, we chose goal modelling techniques of the i* framework to cover the RE for the static part of a DW.
- We extend goal models with modelling techniques of the object-oriented approach like UML, when the RE for the dynamic part of a DW is captured.

2 Running Example

We illustrate the proposed method with a concrete example from the health care sector. In the pharmaceutical industry, Adverse Events (AEs) are crucial for pharmaceutical companies needing the assessment of drugs. An AE is an unexpected and harmful reaction resulting from the use of a prescribed medication. Pharmaceutical companies typically need to collect AE data and carry out analytics over AE data to make well-informed decisions in order to avoid AEs. In this regard, we aim to analyse requirements of a BI system which enables a company to follow up its strategic goals relevant to AEs, to observe the current status regarding goals, and to find the factors influencing them. The company decided to track the AE-related KPIs to meet the aforementioned requirements. To this end, we need to store AEs data in a DW repository and analyse them in a dashboard where KPIs are monitored.

3 Static Part of a DW

RE for the static part of a DW discusses what data and in which form is of particular interest for decision makers to be stored in a DW. Our approach to derive the information requirements is to bring together strategic goals, business processes, and KPIs under the structure of a goal model adapted from the i* framework (the i* framework offers two models to represent organisational context: the Strategic Dependency (SD) model and the Strategic Rationale (SR) model). Eventually, we derive the information requirements from KPIs.

3.1 Strategic Dependency Model

To adapt the SD model in the DW domain, our approach is to incorporate a DW and a dashboard as actors in the i* model, since the organisation depends on a DW to obtain the proper information and depends on a dashboard to analyse the achievement of its strategic goals. The following guidelines are provided to adapt the SD model in the DW domain (see Fig. 1).

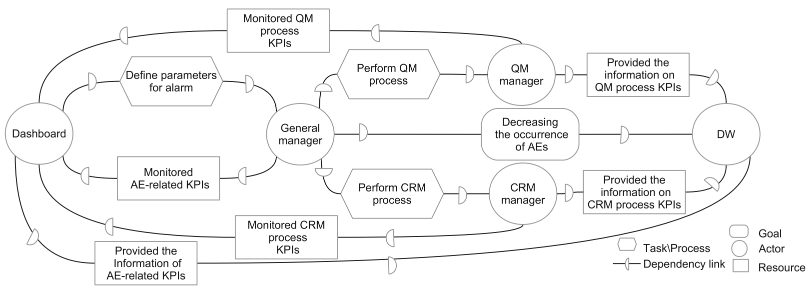

Fig. 1. The SD model adapted in the DW domain

Guideline 1: Business actors are decision makers (e.g. managers, top executives) and process owners. We also represent a DW and a dashboard under development as actors in the SD model.

Guideline 2: Strategic goals of the organisation are determined by decision makers. DWs provides decision makers with the capability of access and analysis of data to evaluate the status of strategic goals. Therefore, we represent strategic goals using a goal between a decision-maker and a DW. For example, "General manager" depends on the "DW" in order to analyse the status of the goal "Decreasing the occurrence of AEs".

Guideline 3: Strategic goals are achieved through performing some business processes. We represent the processes involved by means of a task between decision makers and process owners. For example, "General manager" depends on "QM manager" and "CRM manager" performing "QM process" and "CRM process", respectively, to accomplish "Decreasing the occurrence of AEs".

Guideline 4: Process owners use KPIs to evaluate the performance of their business processes. We represent the information required for KPIs by process owners as a resource dependency between process owners and the DW, since this information is provided by DWs. For example, the "QM manager" depends on the DW for the resource "Provided the information on QM process KPIs".

Guideline 5: Decision makers depend on dashboards to monitor the status of strategic goals. We represent this dependency using a resource between a decision maker and a dashboard, since this monitoring is provided by dashboards. For example, "Monitored AE-related KPIs" is the resource that relates "General manager" to "AE dashboard".

Guideline 6: Dashboards provide process owners with the capability of monitoring KPIs, representing the status of a process. We represent this dependency using a resource between a process owner and a dashboard. For example, "QM manager" depends on "AE Dashboard" for the resource "Monitored QM KPIs".

Guideline 7: Dashboards depend on decision makers to define certain parameters for generating alarms when KPIs are monitored. We represent this dependency using a task between dashboards and decision makers. For example, "AE dashboard" depends on "General manager" to perform "Define parameters for alarm".

Guideline 8: Dashboards depend on DWs to provide the required information to monitor KPIs. We represent this dependency by a resource between a dashboard and a DW. In the example, "AE dashboard" is connected to "DW" through the resource dependency "Provided information of AE-related KPIs".

3.2 Strategic Rational Model

We provide the following guidelines to develop the SR model for the DW actor, where KPIs are used to derive information requirements. Figure 2 illustrates the application of the guidelines to the running example.

Guideline 10: For each strategic goal defined according to Guideline 2, the information on the identified KPI, representing the goal fulfillment, needs to be

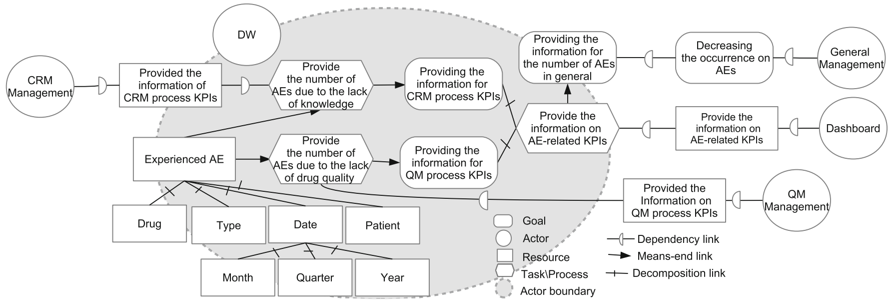

Fig. 2. The SR model for the DW actor

provided by a DW. Typically such KPI is a composite KPI as it measures a high-level goal in the organisation. We represent a goal to provide such information. For example, for "Decreasing the occurrence of AEs", the goal of the DW is "Providing the information on the number of AEs in general".

Guideline 11: For each goal defined according to Guideline 10, the information of corresponding process KPIs needs to be provided in order to achieve the goal. We represent a task, connected with a means-end link, to provide such information. For example, "Providing the information on AE-related KPIs" explains how "Providing the information on the number of AEs in general" is achieved.

Guideline 12: For each task defined according to Guideline 11, providing the KPI information of each process related to the strategic goals is an objective for the DW actor. This information are represented once via a resource dependencies in Guideline 4. We represent a sub-goal for each resource dependency, representing the information on relevant KPIs for each process involved. For example, "Providing the information on CRM process KPIs" and "Providing the information on QM process KPIs" are both sub-goals obtained from "Provide the information on AE-related KPIs".

Guideline 13: For each sub-goal defined according to Guideline 12, the value of each component KPI needs to be aggregated by the DW actor to achieve the sub-goal. We represent a task for each component KPI and connect it to the corresponding sub-goal via a means-end link. For example, "Provide the number of AEs due to the lack of drug quality" explains how "Providing information for the QM process" is achieved.

Guideline 14: For each task defined according to Guideline 13, an analysis needs to be conducted, because each task represents a KPI which is a measure employed to show the status of a process or a strategic goal. The analysis is conducted over some data considered as resources. We illustrate the focus of the analysis as a resource connected to the task with a means-end link. For example, the task "Provide the number of AEs due to the lack of drug quality" defines a measure that requires the resource of "AEs experience" as the focus of the analysis.

Guideline 15: For each resource defined according to Guideline 14, there are contexts that analysis of each measure occurred within them. We represent the concept of a context for the analysis as a resource and we connect them to the focus of analysis by a decomposition link. For example, "Provide the number of AEs due to the lack of knowledge" reveals that AEs can be analysed based on two variables: the AE type, and the patients who are experiencing AEs. For the other KPI it is necessary to analyse AEs in the context of drugs. Both KPIs are evaluated monthly as it has been mentioned in Sect. 2, so that the dates of AE occurrences can be another context to analyse AEs.

Guideline 16: For some of the resources defined according to Guideline 15, which represent a context of the analysis, there are several aggregation levels. We represent these levels of aggregation as resources connected via a decomposition link to the corresponding resource representing a context of the analysis. For example, "Date" as a context for the analysis of AEs can include "Month", "Quarter", and "Year".

3.3 Deriving the MD Model from a Goal Model

In this section, we provide guidelines to derive the MD model from the goal model for the DW actor. The MD schema for the running example is illustrated in Fig. 3.

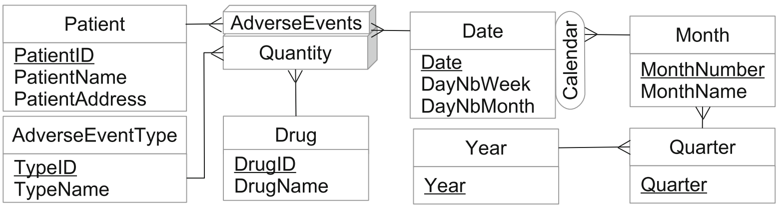

Fig. 3. The MD model derived from the goal model

Guideline 17: Resources representing the focus of the analysis defined according to Guideline 14, are represented as facts in the DW schema. Resources representing the context of the analysis defined according to Guideline 15 are dimensions in the MD model. Resources representing the levels of aggregation defined according to Guideline 16 are represented as hierarchies of dimensions in the DW schema.

4 Dynamic Part of a DW

In this section, we focus on the analytics of data collected in a DW. We narrowed down the scope to KPIs included in a dashboard. We first develop a strategic dependency model for a dashboard. Then, we discuss how to derive a use case diagram, representing the interaction of the business users with the DW, from the SR model.

4.1 SR Model for the Dynamic Part

In this part we provide some guidelines to develop the SR model for a dashboard. A dashboards that monitors KPIs typically provides decision makers and process owners with Performance Level (PL) of KPIs. The PL defined as the current value of a KPI compared against a set of parameters: a target value, a threshold value and a minimum value (refer to [9] for more details). Figure 4 illustrates the SR model of a dashboard actor for the running example.

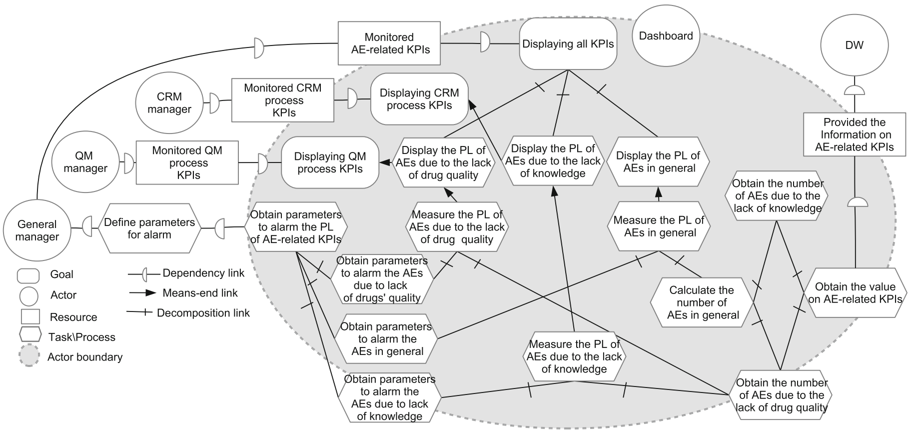

Fig. 4. The SR model for the dashboard actor

Guideline 18: Dashboards provide decision makers and process owners with a KPI monitoring service according to Guidelines 5 and 6; we map each resource defined in the SD model for this purpose to a goal for the dashboard actor. For example, the objective of "AE dashboard" is: (1) "Displaying AE-related KPIs" to "General manager" and (2) "Displaying QM process KPIs" to "QM manager".

Guideline 19: Goals defined according to Guideline 18 are achieved if a dashboard is able to represent the Performance Level (PL) of relevant KPIs for the corresponding decision-maker and business owner actors. We define a task for each KPI to be displayed by a dashboard and connect them with a means-end link. For example, "AE dashboard" performs task "Display the PL of AEs due to lack of drug quality", which explains how goal "Displaying QM process KPIs" is achieved. Thus, a means-end link is created between them.

Guideline 20: Tasks defined according to Guideline 21 are performed if a dashboard is able to measure the PL of KPIs. We represent a task for each KPI to be measured and connect it to the corresponding task defined in Guideline 21 by a means-end link, since it explains how the PL of a KPI is displayed. For example, "Measure the PL of the AEs due to lack of knowledge" is defined and connected by a means-end link to "Display the PL of the number of AEs due to lack of knowledge".

Guideline 21: DWs provide dashboards with the information of atomic KPIs according to Guideline 8; we map the resource defined in the SD model for this purpose to a task for the dashboard actor. Dashboards need this information to measure the PL of KPIs. We represent a task to obtain this information. For example, "Obtaining the value of AE-related KPIs" is defined as a task for "AE dashboard".

Guideline 22: The task defined according to Guideline 21 is decomposed into sub-tasks, each representing the information of an individual atomic KPI. For example, "Obtain the value of AE-related KPIs" is decomposed into "Obtain the value of the number of AEs due to lack of knowledge" and "Obtain the value of AEs due to lack of drug quality".

Guideline 23: Decision makers provide dashboards with parameters according to Guideline 7 in order to measure the PL of KPIs; we map the task defined in the SD model for this purpose to a task for the dashboard actor to obtain these parameters. For example, "Obtain the parameters to alarm the PL of AE-related KPIs" is defined as a task for the "AE dashboard".

Guideline 24: Tasks defined according to Guideline 23 are decomposed into sub-tasks, each representing the parameters required to measure the performance level of a KPI. For example, "Obtain the parameters of AE-related KPIs" is decomposed into "Obtain parameters to alarm the PL of AEs due to lack of knowledge", "Obtain parameters to alarm AEs due to lack of drugs quality", and "Obtain parameters to alarm the AEs in general".

Guideline 25: Tasks defined according to Guideline 20 measure the PL of a KPI. To do that, the dashboard actor needs the current value of the given KPI, as well as parameters defined by decision makers. Depend on what a task represents (an atomic or a composite KPI), we take the following actions:

- For an atomic KPIs, we connect the task defined in Guideline 20 with relevant tasks, defined according to Guidelines 24 and 26, via decomposition links. For example, "Measure the PL of AEs due to the lack of knowledge" is decomposed to "Obtain parameters to alarm the AEs due to the lack of knowledge" and "Obtain the number of AEs due to the lack of knowledge".
- For a composite KPI, we initially define a new task for the dashboard actor to calculate the value of a composite KPI. To calculate this, the dashboard actor needs the value of its component KPIs (obtained from the DW). Thus, we connect the new task with corresponding tasks defined in Guideline 22. Then, we connect each task defined for a composite KPI in Guideline 20, with the new task and corresponding tasks defined in Guideline 24, via decomposition links. For example, a new task is defined to "Calculate the number of AEs in general". This task is decomposed to "Obtain the number of AEs due to the lack of knowledge" and "Obtain the number of AEs due to the lack of drug quality". Also, the task which "Measure the PL of AEs in general" is

decomposed into "Calculate the number of AEs in general" and ""Obtain parameters to alarm the number of AEs in general".

4.2 Deriving the Use Case Diagram from the Goal Model

As we continue the development process, we need to focus on how users interact with the system. For this purpose, we initially adapt the use case. A use case contains steps that define interactions between an actor and a system, to achieve a goal. Figure 5 illustrates how the following guidelines are applied to map from a goal model to a use case diagram in the running example (to avoid, complexity we only show some examples mapped, using dashed-arrows).

Guideline 26: Actors of the SD model, which are connected to the dashboard with a dependency link, are candidates for being a use case actor. The boundary of the system is the dashboard, presenting the dynamic part of the DW. Decision makers, process owners, and DWs are actors in a use case diagram.

Guideline 27: For each task and resource dependency with decision makers in the SD model, a task and a goal is defined for the dashboard according to

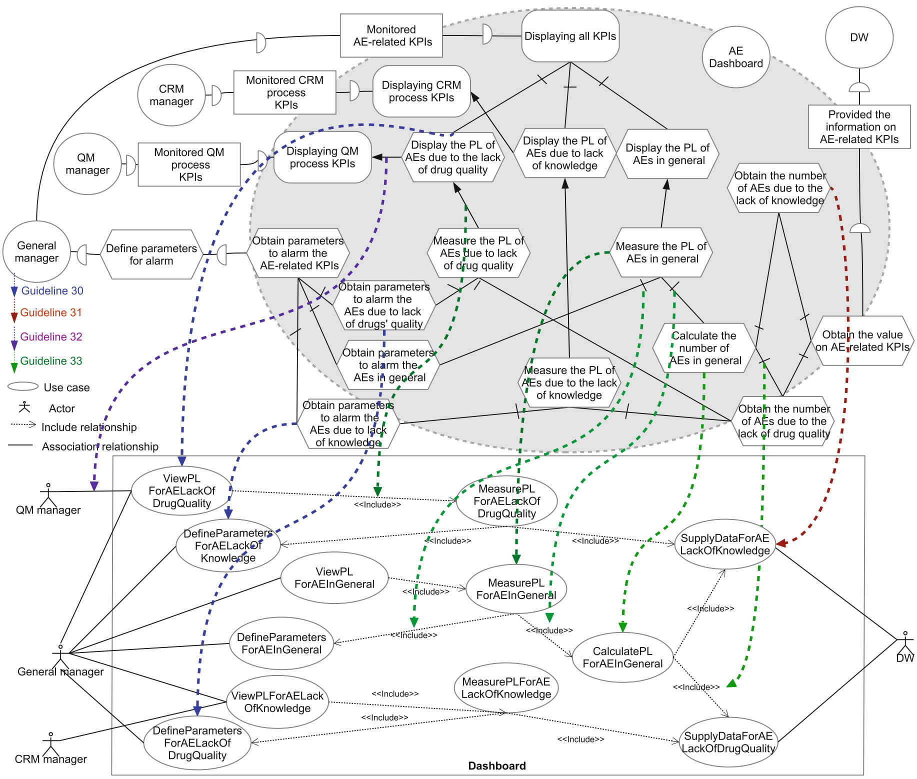

Fig. 5. Use case diagram for the dashboard

Guidelines 18 and 23, respectively. The SR model helps to investigate if there are decomposition links for the defined task and the defined goal. The lowest level tasks are candidates for being a use case as the modelling principle to develop a use case diagram is to ensure that each use case represents a single user goal. Each use case is connected with a corresponding decision-maker actors via an association link.

Guideline 28: For each resource dependency with the DW in the SD model, a task is defined for a dashboard according to Guideline 21. The SR model helps to investigate, if there are decomposition links for the defined task. The lowest level tasks are candidates for being a use case.

Guideline 29: For each resource dependency with process owners in the SD model, a goal is defined for the dashboard according to Guideline 21. In the SR model, these goals are connected with some tasks via means-end-links. The tasks are already candidates according to Guideline 30. We translate means-ends links of the SR model to the association links of a use case diagram, connecting use cases with relevant process owners.

Guideline 30: For each low-level task in the SR model, which is not directly derived from a dependency link of the SD model, we represent a use case. It is important to notice that we can not draw a direct association links between these newly defined use cases and actors. We need to connect them with already existing use cases using an include link (an include link is used to insert the behaviour of a use case to the behaviour of another one called base use case). To define a proper connection, we investigate the discussed tasks in the SR model in terms of links exiting from them as follows: (1) If it is a means-end link, we draw an include link between the newly defined use case and the use case which represents the task in the end side of the means-end link of the SR model (the latter use case is the base use case); (2) If it is a decomposition link, we draw an include link between the newly defined use case and the use case which represents the decomposed task of the SR model (the newly defined use case is the base use case).

5 Conclusion

In this paper, we have addressed the RE regarding the dynamic and the static part of a DW. In our approach models provide the basis Modelling is the core of our approach for requirements artefacts. For the static part, we adapted the goal model from the i* framework to derive a MD schema. We found that, goal models alone are not adequate to deal with the RE of the dynamic part of a DW. Therefore, we extended the goal model with the use case diagram to illustrate how users can interact with a DW through a KPI monitoring dashboard. We initially use KPIs as the main drivers to obtain our MD model and then discuss how decision-makers analyse KPIs in order to measure the success of an organisation. This approach aligns data required to be stored in a DW with the analytics conducted over the data, since both use the concept of KPI to derive the requirements. As future work, our method will be completed with use case scenarios and other behavior and interaction diagrams of UML in order to visualize all aspects of the dynamic part of a DW.

References

1. Barone, D., Jiang, L., Amyot, D., Mylopoulos, J.: Composite Indicators for Business Intelligence. In: Jeusfeld, M., Delcambre, L., Ling, T.-W. (eds.) ER 2011. LNCS, vol. 6998, pp. 448–458. Springer, Heidelberg (2011)
2. Bonifati, A., Cattaneo, F., Ceri, S., Fuggetta, A., Paraboschi, S.: Designing data marts for data warehouses. ACM Trans. Softw. Eng. Methodol. **10**(4), 452–483 (2001)
3. Gallardo, J., Giacaman, G., Meneses, C., Marbán, Ó.: Framework for decisional business modeling and requirements modeling in data mining projects. In: Corchado, E., Yin, H. (eds.) IDEAL 2009. LNCS, vol. 5788, pp. 268–275. Springer, Heidelberg (2009)
4. Ghezzi, C., Jazayeri, M., Mandrioli, D.: GRAnD: a goal-oriented approach to requirement analysis in data warehouses. Decis. Support Syst. **45**(1), 4–21 (2008)
5. Malinowski, E., Zimányi, E.: Requirements specification and conceptual modeling for spatial data warehouses. In: Meersman, R., Tari, Z., Herrero, P. (eds.) OTM 2006 Workshops. LNCS, vol. 4278, pp. 1616–1625. Springer, Heidelberg (2006)
6. Mazón, J.-N., Pardillo, J., Trujillo, J.: A model-driven goal-oriented requirement engineering approach for data warehouses. In: Hainaut, J.-L., et al. (eds.) ER Workshops 2007. LNCS, vol. 4802, pp. 255–264. Springer, Heidelberg (2007)
7. Mazón, J., Trujillo, J., Lechtenbörger, J.: Reconciling requirement-driven data warehouses with data sources via multidimensional normal forms. Data Knowl. Eng. **63**(3), 725–751 (2007)
8. Mazon, J., Trujillo, J., Serrano, M., Piattini, M.: Designing data warehouses: from business requirement analysis to multidimensional modeling. In: Proceedings of International Conference on Requirements Engineering for Business Need and IT Alignment, pp. 44–53 (2005)
9. Nasiri, A., Zimányi, E., Wrembel, R.: Requirements engineering for data warehouses. In: Proceedings of Conference on Journées Francophones Sur Les Entrepôts de Données et l'Analyse en ligne (2015)
10. Prakash, N., Bhardwaj, H.: Early information requirements engineering for target driven data warehouse development. In: Sandkuhl, K., Seigerroth, U., Stirna, J. (eds.) The Practice of Enterprise Modeling. LNBIP, vol. 134, pp. 188–202. Springer, Heidelberg (2012)
11. Silva, V., Mazón, J., Garrigós, I., Trujillo, J., Mylopoulos, J.: Monitoring strategic goals in data warehouses with awareness requirements. In: ACM Symposium on Applied Computing, pp. 1075–1082. ACM (2012)
12. Singh, Y., Gosain, A., Kumar, M.: From early requirements to late requirements modeling for a data warehouse. In: Proceedings of IEEE International Joint Conference on INC, IMS and IDC, pp. 798–804 (2009)
13. Stroh, D., Winter, R., Wortmann, F.: Method support of information requirements analysis for analytical information systems. Bus. Inf. Syst. Eng. **3**(1), 33–43 (2011)
14. Vaisman, I., Zimányi, E.: Data Warehouse Systems: Design and Implementation. Springer, Heidelberg (2014)

15. Winter, R., Strauch, B.: A method for demand-driven information requirements analysis in data warehousing projects. In: Proceedings of IEEE International Conference on System Sciences, pp. 9–18 (2003)
16. Winter, R., Strauch, B.: Information requirements engineering for data warehouse systems. In: ACM Symposium on Applied, Computing, pp. 1359–1365 (2004)