# Enabling Off-Line Business Process Analysis: A Transformation-Based Approach

**Arnon Sturm**

Department of Information Systems Engineering

Ben-Gurion University of the Negev, Beer Sheva 84105, Israel

sturm@bgu.ac.il

**Abstract.** Business process lifecycle consists of four stages: process specification, process implementation, process execution and monitoring, and process analysis. Usually, the relationships among the stages are loosely defined. Due to the frequent changes in today's market, business processes are also changing constantly, thus the need for synchronized the various business process stages is increased. In this paper, we proposed a method for constructing data warehouse schemata from business process specification in order to allow the off-line analysis of the business process execution. We define the relationship among the business process specification stage and its monitoring and analysis stages, in order to facilitate querying the business process performance to identify potential improvements. The semi-automated transformation process also adheres with the need to address frequent changes as changing the business process specification will result in change within its monitoring requirements as well as its analysis infrastructure (i.e., the data warehouse schemata).

## 1 Introduction

Business Process Management has gained a lot of attention in recent years. That attention had led to the development of many tools (and techniques) supporting the business process lifecycle. However, these tools (and techniques) focused on several separate activities neglecting the required integration and synchronization among them.

The classical business process life cycle consists of four stages: process specification, process implementation, process execution and monitoring, and process analysis. Process specification refers to the business process definition in terms of activities, data and control flow, performing roles, etc. Process implementation refers to the way according to which business processes are realized. For example, a business process may be realized using an Enterprise Resource Planning (ERP) package or using the emerging technology of Service-Oriented Architecture (SOA). The process execution and monitoring stage refers to the actual deployment of the business process and its data collection for various purposes. Finally the process analysis stage deals with the processing of the monitored data and studying its affect on the existing business process, which may results in changes within its specification. The process analysis stage may be on-line in terms of learning and changing the business process as it is executed, or may be off-line, which is actually an on-line analytical process (OLAP) procedure for measuring the business process performance.

In this paper we adopt a business process life cycle of the following kind:

- The process specification stage consists of an organizational model, a data model, and a process model (BPMN [12]).
- The process implementation stage is done by transforming the process model into a grounded (i.e., connected with web services) business process execution language (BPEL [11]).
- The process execution and monitoring stage is done by deploying the grounded BPEL to one of the existing BPEL engines (e.g., activeBPEL [1])
- The process analysis in this case refers to the off-line option in which a data warehouse [9] is used for measuring the business process performance.

We propose to utilize the various artifacts of the business process specification and the information gained within the process execution for facilitating the off-line business process analysis via a data warehouse. In particular, in this paper we propose a method for transforming business process specification into data warehouse schemata (i.e., Snowflake schemata). The goal of this transformation is to enable the identification of potential improvements to be introduced to the business.

The rest of the paper is organized as follows. Section 2 reviews existing techniques for creating data warehouse schemata, whereas Section 3 provides an overview of the Snowflake scheme definition and usages with respect to analyzing business processes. Section 4 introduces the principles on which the proposed method works, its expected input, and the transformation rules according to which the data warehouse schemata are constructed. Finally, in Section 5 we conclude and set the basis for future research.

## 2 Techniques for Transforming an Operational System Model to a Data Warehouse Model

The conceptual and logical design of the data warehouse is a complicated human-intensive task, which is usually assigned to systems engineers and analysts of the organization. Lately, several techniques were introduced in order to help data warehouse designers to cope with the task at hand. These techniques aim at utilizing the operational system specification. There are two major approaches for utilizing operational system models for the construction of data warehouse models: the structure-based approach and the process-based approach.

The structure-based approach is motivated by the understanding that since data warehouses depend on the respective underlying operational systems as their data supply sources, the latter become important for the conceptual and logical design of a data warehouse. The techniques that follow this approach (which appear in [5]) use mainly the entity-relationship concept [3] as the basis for the proposed solution. However, these techniques suffer from the following limitations: (1) the data warehouse designers are assumed to be familiar with the organization's business processes, (2) the behavioral aspects of the system are ignored, and (3) multiple manual decisions and transformations are needed to obtain a data warehouse model.

The process-based approach stems from the understanding that the fundamental role of data warehousing is to provide business performance measurement and redesign support [8, 10]. With this in mind, several techniques were offered to create data warehouse models from business process models ([2], [7]). However, since these techniques are manual and does not relate to the operational system structure, the Extract, Transform, and Load (ETL) process, which deals with loading the operational system data to the data warehouse system, is complicated.

A survey of the state-of-the-art methods for transforming operational system conceptual models to data warehouse models along with their evaluation is presented in [5]. That survey points out that the structural-based approaches mainly suffer from lack of guidance for constructing the data warehouse schemata and from lack of adequate analysis of the process perspective, which is the goal for which the OLAP notion was developed for.

In [6] the authors address exactly the problems aforementioned by utilizing a proprietary single view modeling language – the Object Process Methodology (OPM) [4], which considers the processes as "equal" actors within a model. That method refers to persistent data neglecting the impact of non-persistent data that might affect the process execution. In this paper we follow the principles suggested by that work, apply it to a standard modeling language, namely, BPMN, and discuss its integration with the business process lifecycle.

## 3 The Role of a Snowflake Schema in Analyzing Business Processes

"A Snowflake schema is a way of arranging tables in a relational database such that the entity relationship diagram resembles a snowflake in shape". At the center of the schema are fact tables which are connected to multiple dimensions. A fact table consists of the measurements, metrics or facts of a business process, whereas a dimension is a data element that categorizes each item in a data set into non-overlapping regions [13]. In the case of a Snowflake scheme, dimensions are elaborated with multiple levels. Having defined a Snowflake scheme within a relational database, its population is done following a procedure of Extract, Transform, and Load (ETL) from the operational system database. Then, off-line analysis of the data within the data warehouse can be performed.

Snowflake schemata are essentially cubes with multiple hierarchal dimensions. This enables users to view the data from different viewpoint and at various levels of granularity. A cube enables the following operations:

1. *Roll-up* which means summarizing or aggregating data from lower levels.

2. *Drill-down* which means going down to lower levels to detailed data or to new dimensions.
3. *Slice and dice* which means projection and selection of relevant data.
4. *Rotate the cube* which means looking at the data from different viewpoints.

Thus, a Snowflake scheme serves as a vehicle for analyzing business processes.

# 4 Transforming Business Process Specification to Data Warehouse Schemata

The business process specification is the core asset of the organization. Thus, in this paper we utilize that asset for deriving additional asset – data warehouse schemata – to facilitate the continuous improving of the business process by analyzing its performance. Nowadays, it is clear that today's markets are changing rapidly, thus adaptation of business processes should follow these changes (throughout the business process lifecycle) in a seamless manner.

Following that need, we proposed a method for transforming business process specification into data warehouse schemata. The business process specification consists of organizational model, data model, and a process model. In this section, we first describe the business process specification as the input for the proposed method, and then we present the transformation rules which are demonstrated via a case study.

## 4.1 Business Process Specification

As stated before, the business process specification, we consider, consists of the following: (1) an organizational model which consists of organizational units and associated roles[1]; (2) a data model which is a subset of the class diagram notations that include classes, their attributes, and the relationships among classes; and (3) a business process model which is a set of BPMN diagrams. We chose BPMN as the modeling language for that task due to the following reasons: (a) it is the emerging standard for business modeling and (b) it has a transformation to executable code and platforms, which can be used for populating the data warehouse schemata.

In the following a partial example of a business process of handling a travel request within an organization is described. The process starts when an employee fills a request and sends it to his department manager. The department manager needs to check the necessity of the travel. Then the marketing department performs its own examination of the request and might add further requirements. Next, the finance department checks that budget is available for that travel, and the final approval is done by the general manager (CEO). Upon all approvals the travel is ordered. At any

[1] The organizational model will not be elaborate here as we do not refer to its content explicitly, rather we use it meta model.

stage, there is a possibility that the travel will not be approved. Figure 1 depicts the business process of issuing a travel request.
In Figure 2 the data model associated with the travel request process is described.

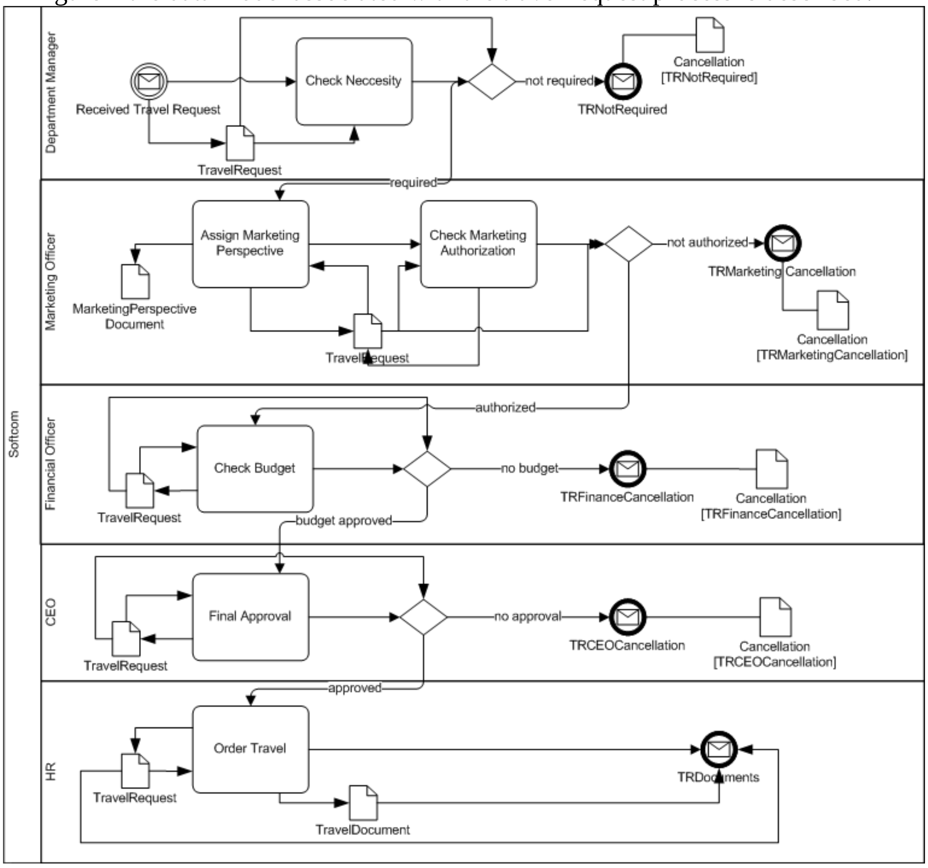


**Fig. 1.** The travel request process model

## 4.2 The Transformation Rule Based Method

A Snowflake scheme is the basic data warehouse structure that allows users to perform multidimensional data analysis. Basically, the proposed method can generate all possible Snowflake schemata for the business process specification. However, creating multiple schemata will overload the data warehouse designer because she will need to handle several ETL procedures and to manage irrelevant data. Thus, we advocate that the data warehouse designer should carefully select the required business processes to be analyzed. The method consists of two stages: the creation of the Snowflake schema and the population of the fact and dimensions with relevant attributes.

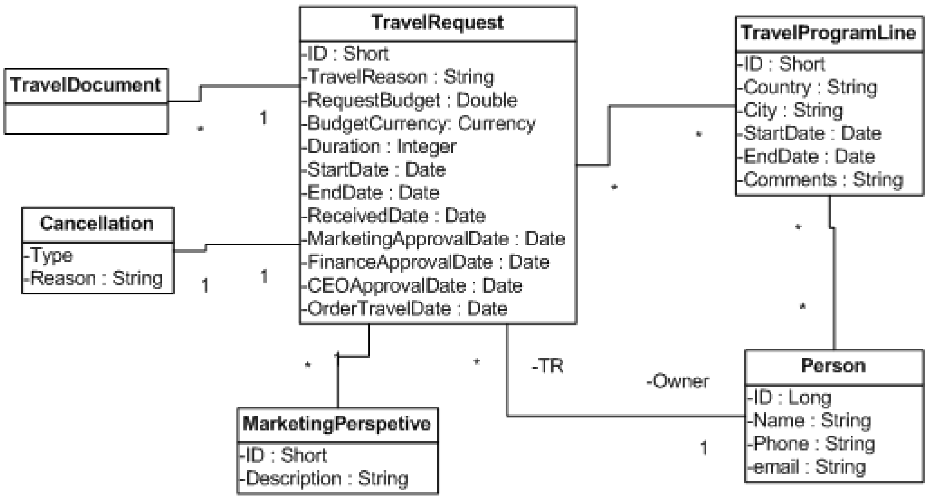


**Fig. 2.** The travel request data model

Upon selecting the desired business process the following rules should be applied.

**Stage 1: Creating a basic Snowflake schema**

Action Rule 1.1: Create an empty Snowflake schema for each data object which is affected by the selected process.

In the case of the travel request business process the following Snowflake schemata are created: Travel Request, Cancellation, and Travel Document (these data objects are associated with the outgoing messages of the business process[2].

Action Rule 1.2: Create a dimension for each data object that participates in the process (but not created by it).

In the case of the travel request business process the Travel Request Snowflake scheme consists of the Travel Request as a dimension since it is also affects the process (it is associated with the initiating message).

Action Rule 1.3: Create a dimension for role and unit hierarchy.

In general, since we would like to analyze the performers of the process we add the dimension of roles and units[3] irrespectively of the actual operational specification.

Action Rule 1.4: Create a dimension for each data object that is related to the fact data object within the data model.

In the case of the travel request business process according to the data model, the following dimensions are added: Travel Document, Cancellation, Travel Program Line, Marketing Perspective, and Person.

Having defined the core Snowflake scheme it should be enhanced with attributes and navigational properties. This is achieved in stage 2.

**Stage 2: Populating facts and dimensions**

---

2 In the following, we will refer only to the Travel Request Snowflake scheme.

3 These are originated from the meta model of the organizational model.

Action Rule 2.1: Create a fact for each quantitative attribute of the affected data object.
In the case of the Travel Request Snowflake scheme the fact table gets only two quantitative attributes as defined within the data model: RequestBudegt and Duration.
Action Rule 2.2: Add all attributes of dimensional data objects into the Snowflake schema.
Following the data model all dimension tables has their attributes as defined in the data model.
Action Rule 2.3: Define foreign keys in dimensional data objects as navigational attributes.
Attributes of dimensional data objects, which participate in an association with other data objects in the data model, are proposed as navigational attributes of those data objects in the Snowflake scheme. This way, a dimensional hierarchy is created (and may be collapsed in case of a Star scheme is required).
In the case of the Travel Request Snowflake scheme the Person dimension is added to the Travel Program Line.
Action Rule 2.4: Add basic dimensions. Basic dimensions include Time, Currency, and Measurement Unit.
In the case of the Travel Request Snowflake scheme, based on the type of the attributes, date dimensions are added as well as a dimension for the Currency.

The Snowflake scheme presented in Figure 3 is a result of applying the set of transformation rules defined before.
Note that the transformation rules do not defer between persistent and non-persistent data objects within the business process specification. Thus, an integrated Snowflake scheme is created. However, when dealing with the ETL procedure we should take this information into consideration as multiple sources are required for populating the data warehouse. That is, the operational system database and the execution platform logging and monitoring facilities.

Having the Travel Request Snowflake scheme (and of course its population) one can gain understanding of the following:

1. Which unit causes the highest rate of cancellations?
   This might lead to add a new step before entering the travel request process.
2. How long does it take to handle various approval steps?
   This might lead for improvement of the response time of each one of the handling units.
3. What is the most "consuming" destination in terms of budget allocation?
   This might lead to seek for cheaper travel packages for selected destinations.

Additional queries can be also executed on the data warehouse (based on the created Snowflake scheme) in order to gain additional understanding about the business process performance.

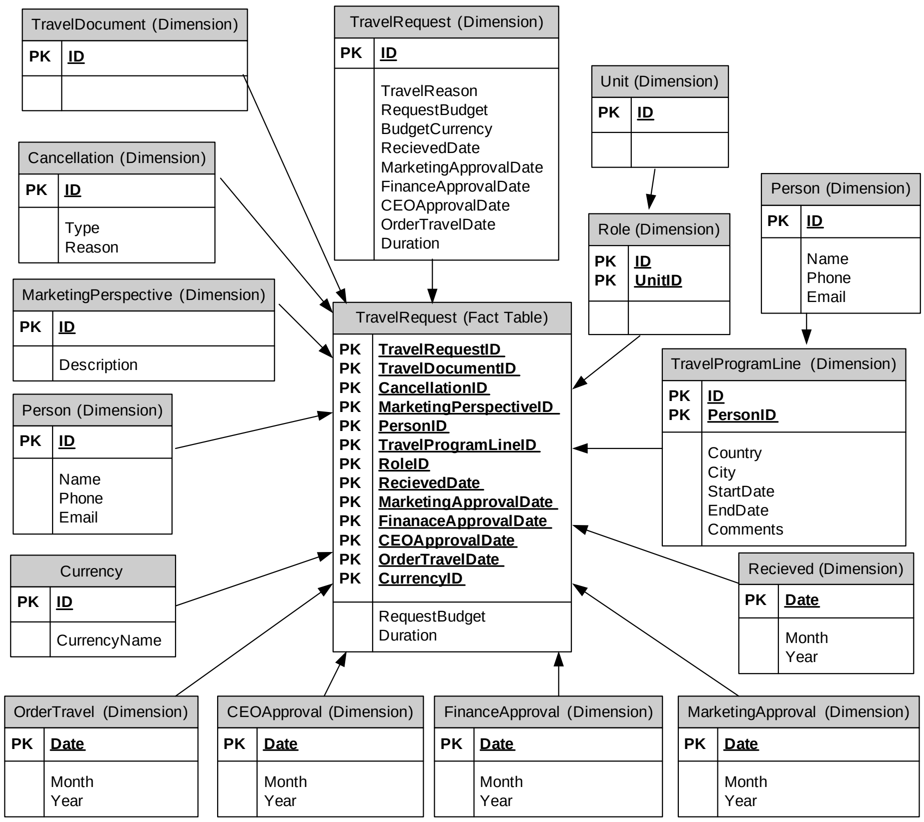


**Fig. 3.** The snowflake scheme for the Travel Request

Having defined a Snowflake scheme of the entire travel request process, we can further specify additional schemata for its inner processes or tasks. For example if we pick the Assign Marketing Perspective tasks we get two Snowflake schemata: one for the Travel Request and one for the Market Perspective Document. The construction process of the complete Snowflake scheme is done by applying the same transformation rules.

## 5 Summary

In the paper we present a method for transforming business process specification into a set of data warehouse schemata which enables off-line analysis of business processes in order to facilitate their improvement over time. Utilizing that method enables a business process to be adjusted according to the required changes. The adjustment relates to the construction of new data warehouse schemata whenever the business process changes with limited efforts.

The proposed method is based on a set of rules which determine that way according to which the data warehouse schemata will be constructed. It can be applied to any abstraction level of the process whether atomic or composite. We also demonstrated the proposed method via a case study.

The implementation of the proposed method is underway and in general it is a straight forward one. Yet, the implementation of the ETL procedure for populating the created schemata requires additional attention as it may involve several sources such as persistent and non-persistent information. We intend to examine the way according to which the executed platform can be configured for monitoring purposes based on the business process specification. In addition, we plan to study the extent to which various sources such as the operational database and the information extracted from the execution platform (e.g., BPEL engine) can be used for populating the data warehouse schemata generated based on the business process specification.

## Acknowledgement

This work was partially funded by Deutsche Telekom Laboratories at Ben-Gurion University.

## References

1. Active Endpoints: Active BPEL, http://www.active-endpoints.com/products-overview.htm, 2007.
2. Boehnlein, M. and Ende, A. U., Business Process Oriented Development of Data Warehouse Structures, In Proceedings of Data Warehouse 2000, Physica Verlag, Friedrichshafen, Germany, 2000, pp. 3-21.
3. Chen, P., The Entity-Relationship Model – Toward a Unified View of Data, QACM Transactions on Database Systems, 1(1), (1976) 9-36.
4. Dori, D., Object-Process Methodology: A Holistic Systems Paradigm. Springer Press, 2002.
5. Dori, D., Feldman, R., and Sturm, A., Transforming an Operational System Model to a Data Warehouse Model: A Survey of Techniques, In: Proceedings of the IEEE International Conference on Software – Science, Technology & Engineering (SwSTE) , 2005, Hertzlia, Israel, 2005, pp. 47-56.
6. Dori, D., Feldman, R., and Sturm, A., An OPM-based Method for Transformation of Operational System Model to Data Warehouse Model, In: Proceedings of the IEEE International Conference on Software – Science, Technology & Engineering (SwSTE) , 2005, Hertzlia, Israel, 2005, pp. 57-66.
7. Ferstl, O.K. and Sinz, E.J., SOM Modeling of Business Systems, in: Bernus, P.; Mertins, K.; Schmidt, G. (Eds.), Handbook on Architectures of Information Systems – International Handbook on Information Systems, Springer, 1998, pp. 339 – 358.
8. Kaldeich, C. and Oliviera e Sa, J., Data Warehouse Methodology: A Process Driven Approach, In: Proceedings of CAiSE 2004, 2004, pp. 536-549.
9. Kimball, R., The Data Warehouse Toolkit, John Wiley & Sons, 1996.

10. List, B., Schiefer, J., Tjoa, A.M. and Quirchmayr, G., Multidimensional Business Process Analysis with the Process Warehouse. In: Abramowicz, W. & Zurada, J. (Eds.): Knowledge Discovery for Business Information Systems, Kluwer Academic Publishing, 2000, pp. 211-227.
11. OASIS: Web Services Business Process Execution Language Version 2.0, http://docs.oasis-open.org/wsbpel/2.0/wsbpel-v2.0.pdf, 2007.
12. OMG-BMI, Business Process Modeling Notation Specification, 2006.
13. Wikipedia, http://en.wikipedia.org, 2007.