A Comparison of Data Warehouse
Development Methodologies
Case Study of the Process Warehouse

Beate List[1], Robert M. Bruckner[1], Karl Machaczek[1], Josef Schiefer[2],

[1] Institute of Software Technology and Interactive Systems
Vienna University of Technology
1040 Wien, Austria
{list, bruckner, machaczek}@ifs.tuwien.ac.at

[2] IBM Research Division, Thomas J. Watson Research Center
Yorktown Heights, NY 10598, United States
josef.schiefer@us.ibm.com

Abstract. Building a data warehouse is a very challenging issue because compared to software engineering it is quite a young discipline and does not yet offer well-established strategies and techniques for the development process. Current data warehouse development methods can fall within three basic groups: data-driven, goal-driven and user-driven. All three development approaches have been applied to the Process Warehouse that is used as the foundation of a process-oriented decision support system, which aims to analyse and improve business processes continuously. In this paper we evaluate all three development methodologies by various assessment criteria. The aim is to establish a link between the methodology and the requirement domain.

1   Introduction

During the last decade data warehouse systems have become an essential component of modern decision support systems in large organisations. Data warehouse systems offer efficient access to integrated and historic data from heterogeneous sources to support managers in their planning and decision-making. A data warehouse by itself does not create value; value comes from the use of the data in the warehouse. Hence, improved decision-making results from the better information available in the data warehouse. According to Watson and Haley, the greatest potential benefits of the data warehouse occur when it is used to redesign business processes and to support strategic business objectives. (see [12]). These are also the most difficult benefits to achieve, because of the amount of top management support, commitment, and involvement and the amount of organisational change required.

Building a data warehouse is a very challenging issue because compared to software engineering it is quite a young discipline and does not yet offer well-established strategies and techniques for the development process. A lot of projects fail due to the complexity of the development process. As yet there is no common strategy for the development of data warehouses. Current data warehouse development methods can fall within three basic groups: data-driven, goal-driven and user-driven.

In this paper all three development approaches have been applied to the Process Warehouse that is used as the foundation of a process-oriented decision support system, which aims to analyse and improve business processes continuously (see [10]). We evaluate these development methodologies by the means of application areas, supported management method, targeting organisational level, extent of end user involvement, duration of development and completion, skill level of data warehouse designer, complexity of data model, amount of source systems and longevity of data model. The aim is to establish a link between the methodology and the requirement domain.

The following section presents the related work of data warehouse development methodologies. A brief overview of the Process Warehouse is given in section 3. The data-driven development methodology is applied, analysed to the Process Warehouse in section 4, the user-driven development methodology in section 5 and the goal-driven development methodology in section 6. The paper ends with a conclusion and an assessment of the final results.

2   Related Work

Data-Driven Methodologies: Bill Inmon, the founder of data warehousing argues that data warehouse environments are data driven, in comparison to classical systems, which have a requirement driven development life cycle (see [6]). He states that requirements are the last thing to be considered in the decision support development life cycle, they are understood after the data warehouse has been populated with data and results of queries have been analysed by users. The data warehouse development strategy is based on the analysis of the corporate data model and relevant transactions. The approach ignores the needs of data warehouse users a priori. Company goals and user requirements are not reflected at all. User needs are integrated in the second cycle.

Golfarelli, Maio and Rizzi propose a semi-automated methodology to build a dimensional data warehouse model from the pre-existing E/R schemes that represent operational databases (see [5]).

Goal-Driven Methodologies: Böhnlein and Ulbrich-vom Ende present an approach that is based on the SOM (Semantic Object Model) process modelling technique in order to derive the initial data warehouse structure (see [1]). The first stage of the derivation process determines goals and services the company provides to its customers. Then the business process is analysed by applying the SOM interaction schema that highlights the customers and their transactions with the process under study. In a third

step sequences of transactions are transformed into sequences of existing dependencies that refer to information systems. The last step identifies measures and dimensions: One has to find enforcing (information request) transactions for measures and get dimensions from existing dependencies. In our opinion this highly complex approach works only well when business processes are designed throughout the company and are combined with business goals.

Kimball proposes a four-step approach where he starts to choose a business process, takes the grain of the process, and chooses dimensions and facts (see [7]). He defines a business process as a major operational process in the organisation that is supported by some kind of legacy system (or systems).

User-Driven Methodologies: Westerman describes an approach that was developed at Wal-Mart and has its main focus on implementing business strategy (see [13]). The methodology assumes that the company goal is the same for everyone and the entire company will therefore be pursuing the same direction. It is proposed to set up a first prototype based on the needs of the business. Business people define goals and gather, prioritise as well as define business questions supporting these goals. Afterwards the business questions are prioritised and the most important business questions are defined in terms of data elements, including the definition of hierarchies. Although the Wal-Mart approach focuses on business needs, business goals that are defined by the organisation are not taken into consideration at all.

Poe proposes a catalogue for conducting user interviews in order to collect end user requirements (see [11]). She recommends interviewing different user groups in order to get a complete understanding of the business. The questions cover a very board field and include also topics like job responsibilities.

3   The Process Warehouse

Business process reengineering has turned functional organisations into process organisations (see [3], [4]). In order to gain long-term advantages, it is not sufficient merely to reengineer business processes. It is essential that the newly designed business processes are continuously measured and improved. In order to gauge the performance of business processes, a Performance Measurement System is required According to Kueng it should meet the following requirements: the system must be capable of tracking both financial and non-financial performance indicators, include company-internal and external indicators, store collected data on a non-volatile media so that the data can be analysed over a long period of time, provide a user-friendly interface, which will support, for example, an easy data selection mechanism, consider target values for each performance indicator and disseminate results (see [8]). We address these needs by applying a data warehouse approach, called the *Process Warehouse*, which is defined as follows:

The Process Warehouse (PWH) is a separate read-only analytical database that is used as the foundation of a process-oriented decision support system, which aims to analyse and improve business processes continuously [10].

In the Process Warehouse, basic business process theories are reflected in five perspectives in order to represent fundamental process concepts in an explicit way. The *Business Process Perspective* focuses on the process as a complete entity. The *Organisational Perspective* focuses on the organisational structure of a business process and addresses the fact that business processes, which cross organisational boundaries tend to be inefficient because of changing responsibilities [9]. This perspective supports the detection of delay causing organisational units. The *Product / Service Perspective* focuses on the relationship between product or service and the business process. Its target is to identify the complexity drivers that request a change in the process design, e.g. the segmentation of a process or the special treatment of certain cases.

The *Improvement Support Perspective* is based on the histories of several instances together. The aggregation of instances aims at identifying major performance gaps and deviations, which show that there is need for improvement. As single instances do not have an impact on the aggregated performance, gaps reflect a fundamental performance problem. The *Information Detail Perspective* aims at process information on instance level. It enables the analysis of certain instances and their development over time and helps to determine the cause of performance gaps and deviations.

4 Data-Driven Development Methodology

4.1 Motivation and Application Setting

Current Workflow Management Systems (WfMS) lack comprehensive analysis capabilities. The presented case study can be seen as an extension to a WfMS in order to overcome those limitations. We have applied a data-driven methodology to this case study because commercial Workflow Management Systems have hard coded audit trails in the sense that history data cannot be adjusted to the specific needs of an organisation as well as limited audit trails. Consequently, as much information as possible must be retrieved. Basic components of the WfMS, the meta-model and the audit trail, provide the foundation for identifying information requirements.

The meta-model represents the modelling capabilities of the WfMS. It consists of a process model and an organisation model. The utilised WfMS is IBM MQSeries Workflow™. The organisation model of IBM MQSeries Workflow™ is composed of organisational units, roles and individuals. These entities are assigned as process performers to entities of the process model. The process model of IBM MQSeries Workflow™ is composed of processes, process activities (sub process), block activities (loop) and program activities (execute an application). A process type is defined in the build-time environment.

A process instance is executed in the runtime environment according to its process definition. WfMSs can be seen as state event machines; process and activity in-

stances get through events into certain predefined states. All events in the life cycle of a process or an activity instance are recorded in the audit trail that is either a relational database or a file. Each record in the audit trail contains a number of information including, but not limited to following items: date and time when the event takes place, type of event, process identifier, parent process identifier, activity type, activity state, activity identifier, started program name, role or id associated with the event, etc.

4.2 Data Model

In order to get a data model for this Process Warehouse prototype we analysed the data model of the underlying operational sources and identified the relevant transactions. The process and the organisation meta-model represent the underlying operational sources. The state transition machines for activity and process represent the relevant transactions, which are at runtime tracked in the audit trail. Having business process theory in mind, the entities of the meta-model match perfect with dimensions. We used the audit trail for potential measures.

The audit trail tracks any event change with a timestamp. A process instance may have the following states: *running, suspended, terminated, finished* and *deleted*. The initial process instance state is *created*. The process instance state is *running*, when activities are carried out. A process instance goes into the *suspended* state when a user requests this state explicitly. In this state navigation has stopped and no more activities are carried out. A process instances that has *finished* will be *deleted* later on.

An activity instance is in the state *ready* when a work item is assigned to a role. The state *running* indicates that the item is being processed and the state *terminate* indicates an early termination by an authorised user. In certain situations, the activity implementation cannot be carried out correctly. In this case, the activity is put into the *in error* state. The user can carry out the activity outside of the control of the WfMS, through the support of the checkout facility. The user checks out the activity instance, which is put into the *checked out* state. A user can also *suspend* an activity instance. Activity instances may have further states: *disabled, expired, finished, deleted* and *executed*. These states may not resume the activity instance. We decided for process and activity instances to measure the duration of each state.

The conceptual model represents the ADAPT (Application Design for Analytical Processing Technologies, see [2]) notation. The basic elements of ADAPT are *cubes*, *dimensions*, and *formulas*. Measures represent a particular kind of dimension – a measure dimension. Figure 1 shows the Process Warehouse data model with the cubes *Process* and *Activity*. As process and activity instances have different life cycles, we have separated the cubes for process and activity duration. Each cube has its own measure dimension. The measure dimension for the *Process* cube consists of four single measures: *Ready, Running, Suspended* and *Cycle Time* – this fourth measure is calculated by means of a formula. We have not considered the remaining states, as the instance has already finished and may not resume. The *Process* cube has the dimensions *Organisation*, *Process* and *Time*. The *Process* dimension represents the *Business Process Perspective* and consists of the levels process type and process in-

stance. The *Organisation* dimension represents the *Organisational Perspective* and consists of the levels organisation, role and user. The *Organisation* dimension enables the analysis of the process performance of certain organisations, roles or units. The measure dimension for the *Activity* cube consists of six single measures: *Ready, Running, Suspended, Checked Out, In Error* and *Cycle Time* – this measure is calculated by means of a formula. We have not considered the remaining states, as the instance has already finished and may not resume. The *Process* cube has the dimensions *Organisation*, *Process, Activity* and *Time*.
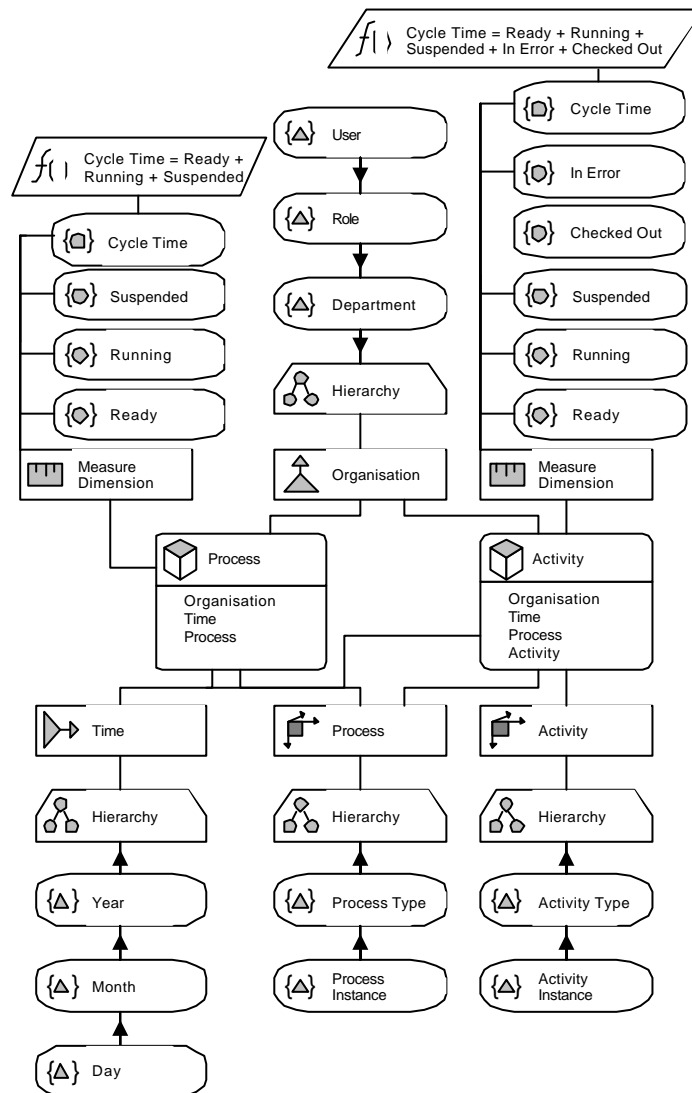

Figure 1: Conceptual design of the Process Warehouse in ADAPT notation

4.3 Evaluation

The data model consists of a few dimensions and a lot of measures. The dimensions represent the basic structure of the WfMS. Measures are time-based and have a very low level of granularity as well as a short-term focus. Basically, measures can be characterised as quantitative, non-financial and unbalanced. Very tight task supervision of individuals can be observed at activity level. In addition to the overall performance of departments, roles and individuals, the development of knowledge can be traced. This approach represents the era of scientific management methods. It supports tayloristic fundamentals of measurement e.g. finding out the best way of doing a job.

Time is not a sufficient key figure to assure long-term process quality. A balance of financial, non-financial, qualitative and quantitative measures, from external and internal sources with the aim of assuring long-term goals is an elementary shortcoming. As the data-driven development methodology allows no scope for long-term strategic visions it is unsuitable for the strategic level of organisation. The tactical level of the organisation or the process owner is only partly supported by the data-driven development methodology. Main shortcomings are the lack of objectives connected to the process definition. Consequently, there is no link to strategic targets. However, the approach is appropriate for managing the process at micro level e.g. management of resources etc. and targets mainly the operational level of the organisational hierarchy.

As the objective of this prototype is to build an extension for a WfMS and the development methodology is based exclusively on the fundamental structure of the system, the integration of end users has not been required. The success of the project fully depends on the skill and experience of the data warehouse designer. No other roles were involved in the project. Due to the lean project structure, the development process was fast and straightforward.

Overall, a data-driven development methodology can generate performance information targeting a non-specified user group at operational level. The information created is suited for tight and narrow monitoring as well as for mining purposes. This approach is especially suited for production workflows, because they generate a high business value, have a high degree of repetition, are customer focused and often time critical. The longevity of the prototype is directly related to the durability of the structure of the WfMS. An integration of external data sources would balance the analysis capabilities of the prototype.

5   User-Driven Development Methodology

5.1   Motivation, Organisational Setting

The goal of the case study that is presented in this chapter was to establish the Process Warehouse in a business environment and to incorporate process performance measurement into the corporate data warehouse. In addition to the advantage of reus-

ing the corporate data warehouse management facilities, it is a first step towards an enterprise-wide performance measurement system. This case study can be also seen as a proof of the Process Warehouse concept in a commercial environment in terms of usability and acceptance. The project was set up as a feasibility study with a strong focus on the production environment. We selected a user-driven development methodology in order to raise the acceptance of the system by constantly interacting with potential users and decision makers. Additionally, it was my concern to convince the company of the merits of this unusual approach for process analysis and to integrate the prototype into the production environment.

The organisation where this case study has been realised is a large insurance company, which is primarily operating in central Europe. The business process is part of the broker business and responsible for proposal processing. IBM MQSeries Workflow™ is utilised to automate the proposal processing business process. In November 2001 a first pilot system went into operation. Today, the workflow engine handles ten users and about 3000 workflow instances per month. The use of this system is limited to two regional offices and a few insurance products. They are planning to launch the system all over Austria and enlarge it on the complete range of product. Up to 150 users will be involved in the processing of some 100.000 instances per year.

5.2 Data Model

We applied the Wal-Mart Model methodology (see [13]), developed by Westerman to this prototype. The insurance company has the significant business need to reduce the process cycle time of the proposal process by more than 50 %. This is the main process goal. Additional goals concerning e.g. quality or cost were not available.

Two data models have been developed. When we finished the first data model, a new person with management experience joined our group and criticised the model as not focusing on top-level management and their decision support requirements. We started again with phase two and developed a second data model.

The second model has had a broader perspective. The focus is on comprehensive business questions e.g. Is the broker responsible for the waiting time?, What insurance product was processed?, etc. and integrated more external data sources. Insurance companies traditionally have a very powerful labour union; therefore we did not analyse the performance of individuals. Compared to the data model in Figure 1, it has got additional dimensions (broker, product, region) that integrate to a larger extent the external environment and fewer measures (working time and waiting time). The *Broker* dimension analyses processes that belong to certain broker and targets the relationship between dealer and cycle time. In this case there was the concern that some brokers deliver incomplete data and increase the cycle time. The *Product* dimension questions whether there is a difference in terms of cycle time between products. Some might be more demanding than others. The *Region* dimension targets different ways of processing the proposal in certain region, e.g. a different skill level. Beside the audit trail, we accessed the proposal database for product, region and broker data. The product, broker and region dimension were reused from the corporate data warehouse.

5.3 Evaluation

The data model contains more dimensions than the model created with data-driven methodology, but has fewer measures, which are time-based and have a low level of granularity as well as a historic and short-term focus. Measures do not represent the state structure of MQ Series Workflow™. Financial measures would have been possible, but were not an issue in the business question definition session. Dimensions represent either the structure of MQ Series Workflow™ or external data sources. The dimension *Product* represents the Product / Service Perspective of the Process Warehouse. The dimension *Broker* partly represents the customer aspect. Basically, measures can be characterised as quantitative, non-financial and unbalanced.

The user-driven development methodology generates performance information that is targeting the organisational level of the people who have been involved in the project. Every person who dominated the data model created a different one. The granularity and the analysis requirements of the models were also different and represented the hierarchical level, the profession, the educational background, the experience and perceptions of the people involved. When the players changed, the requirements changed as well and the complete model might become obsolete. Consequently, the longevity of these models is very short-dated. Overall, the models also represent the company culture, for example the culture of powerful labour union (e.g. very close supervision of individuals has not been a concern).

The fundamental pitfall of the Wal-Mart model is that it is based on the existence of transparent business goals throughout an organisation. At the insurance company, the corporate goals are not communicated to its employees; therefore they are not in a position to support their firm's business strategy. Selected measures, dimensions, the level of granularity and the targeting level of the organisational hierarchy depend on the people involved. The user-driven development methodology requires a project manager with very strong moderation and leadership skills as well as a skilled data warehouse designer. If the business goals are fuzzy, the project manager must act with even more determination. While a user-driven development methodology supports user acceptance, it does not guarantee decision support. It is useful for prototypes and user-sensitive domains, but a monopolisation of this approach is risky and must be avoided.

6 Goal-Driven Development Methodology

6.1 Motivation, Organisational Setting and Data Model

The project (for details see [8]) was in cooperation between one of the authors and the performance measurement group (http://www.measure.ch/) at the University of Fribourg, Switzerland. The motivation for the project was to create a holistic Process Performance Measurement System. Two business processes have been analysed.

The development methodology utilised is a stakeholder driven approach that has been developed by Kueng (see [8]). He suggests starting with identifying stakeholders (e.g. customer, employee, innovation, investor, etc.) for each process. For each stakeholder, process-relevant goals are identified. From the company goals, business process-specific goals are selected. Each goal is measured by at least one indicator. The indicators aim at achieving long-term goals. This leads to a mix of financial, non-financial, qualitative and quantitative measures, from external and internal sources which take various perspectives into consideration.

Five aspects that represent stakeholders were identified for the service process: finance, customer, employee, environment and development. The customer aspect's goal is high service quality and low product price. Customer satisfaction (as revealed in customer surveys), customer complaints and problem solving time are the performance indicators that measure the goal *high service quality*. The goal of a low product price is measured with the ratio to competitors. The finance (investor) aspect has the goal *high profitability* and measures the performance indicators service process margin and operating assets. The employee's goal *good working conditions* and this aspect is measured by the employee satisfaction rate in a survey. The development aspect has the goal *investment in professional training* and measures training days per employee. The environment aspect has the goal *optimise travels* and measures kilometres per visit.

The indicator Customer satisfaction is represented with the *Customer Satisfaction Cube*. The measure dimension for *Customer Satisfaction* consists of three single measures: *Expectation, Perception* and *Performance Gap.* The *Customer Satisfaction Cube* has the dimensions Questionnaire, Process and Time. Any question of a certain questionnaire can be measured for a process type. The indicator problem solving time is represented with the *Duration Cube*, which measures the cycle time of a process. The *Duration Cube* has the dimensions Organisation, Customer, Process and Time.

6.2 Evaluation

The data model contains a few measures, which are balanced and represent a mix of financial, non-financial, qualitative and quantitative measures with a strategic focus. Facts have a moderate level of granularity and do not support a tayloristic monitoring approach. Cubes have few dimensions that support the corporate goals. The prototype integrates a lot of external source systems, in order to achieve a balanced system.

The data model supports users at all organisational levels. The success of the project depends greatly on the support of top management, as the entire organisation is affected. In order to quantify the strategy of the company and to transfer the strategy into key performance indicators, top management, moderators, economists and data warehouse designers are required. End user involvement is exceptional, although this would increase the acceptance of the system. A goal-driven development methodology supports an organisation in turning strategy into action and guides that organisation towards the future desired. This top-down approach provides assistance for modern management methods. On the whole decision support is based upon it.

7   Summary and Conclusion

In this paper we have evaluated data warehouse development methodologies. The empirical background of this work is based on three case studies of the Process Warehouse. This section compares all methodologies and ends up establishing a link between methodology and requirement domain. Results are compared in Table 1.

Basically, a monopolisation of this user-driven development methodology is risky and must be avoided, as it generates performance information that reflects the organisational level of the people involved. Therefore, selected measures, dimensions, the level of granularity and the targeting level of the organisational hierarchy are very unstable. The methodology has a bottom-up tendency, because most employees do not see the organisation from a broad angle, theirs is a narrow-minded, egocentric point of view. The project duration may be long-winded and very costly, as project participants request long discussions on a lot of unnecessary measures and dimensions. Hence, analysing the criteria of the user-driven methodology does not make sense, because results change with the people involved. This development methodology may well raise acceptance of a system, but must be combined with the data-driven or goal-driven development methodology in order to improve the longevity of the system. The more a system suffers rejection, the more user involvement is required beside a focus on organisational strategies or the corporate data model.

The goal-driven development methodology supports modern management methods and is a foundation for decision support at all organisational levels. The level of granularity is much higher compared to that of the data-driven approach. While the Process Warehouse based on the goal-driven development methodology measures only the cycle time for business processes and has only one cube, the Process Warehouse based on the data-driven development methodology measures the duration of all process and activity states as well as the workload of departments and roles. The development duration of the project tends to be very extensive and costly, as a lot of highly qualified professionals and managers take part in numerous workshops and derive performance indicators from strategy. End-users are rarely involved. They are only required when operational detail matters. As the model is aligned with the corporate strategy, it is very stable. Measures and dimensions are balanced: financial, non-financial, qualitative and quantitative aspects are considered. A lot of data sources are integrated, because a holistic approach is based on all aspects of an organisation.

The data-driven development methodology is recommended for data mining and data exploration purposes. The bottom-up approach exploits the database and is suited for tayloristic measurement. The data-driven development methodology is particularly suited for production workflows. These workflows generate a high business value, have a high degree of repetition, are customer focused, often time critical and therefore require tight and close monitoring. All development methodologies have been applied to IBM MQSeries Workflow™ and measure the process cycle time. The goal-driven development methodology measures exclusively the process cycle time. The user-driven development methodology differentiates between working time and waiting time, while the data-driven development methodology measures three states: ready, suspended and running. The working time is equal to the running state.

Methodology Criteria	Data-Driven	User-Driven	Goal-Driven
Basic Approach	Bottom-up	Bottom-up	Top-Down
Supported Management Method	Taylorism Classical School of Management	None Company is culture reflected	Management by Objectives
Project Support	None	Department	Top Management
Application Area / Requirement Domain	Data Exploration and Data Mining	Raise the Acceptance of a System	Foundation for Decision Support
Targeting Organisational Level	Operational Partly Tactical	Depends on the Group of Interview Partners	Strategic Tactical Operational
Focus	Short-Term Focus	Short-Term Focus	Long-Term Focus
Extent of End User Involvement	None	High	Moderate
Project Duration	Low	Very High	High
Skills of Project Members	Data Warehouse Designer	Moderator Data Warehouse Designer	Moderator Economist Data Warehouse Designer
Number of Measures	Many	Many	Few
Type of Measures	Non-Financial and Quantitative Time-Based and Frequency-Based	Non-Financial and Quantitative Time-Based and Frequency-Based	Balanced: Financial and Non-Financial as well as Qualitative and Quantitative
Level of Granularity	Low	Low	High
Number of Dimensions	Few	Many	Few
Type of Dimensions	Represents the Basic Structure of the Application	Represents the Basic Structure of the Application and external Sources	Represents the Strategic Building Blocks of the Organisation
Number of Source Systems	Low	Moderate	High
Longevity / Stability of Data Model	Long	Short	Long
Cost	Low	High	High

Table 1: Comparison of Data Warehouse Development Methodologies

The ready and the suspended states describe the waiting time in more detail. The ready state represents the duration a work item is assigned to a user and has not been accessed before. The suspended state represents the duration a work item is moved off the work list because it cannot be processed because information required is not available. The breakdown of the waiting time into different states enables the detection of work overload, missing resources or lazy employees. Therefore, the insurance company is going to re-engineer the first prototype and apply the data-driven development methodology. The longevity of the data model is directly related to the durability of the structure of the underlying system. Compared to the goal-driven approach, the project duration was very short, as no end users and no other source systems were involved. Due to the limitation of the audit trail, measures and dimensions are time-based. Their main target is the operational level of the organisation.

The data-driven and the goal-driven development methodology do not stress mutual exclusion. As they pursue different purposes they may exist in parallel. The data-driven development methodology can even be seen as a lower level of detail, as the drill-down path of the goal-driven development methodology. These methodologies are complementary and when used in parallel, the benefit is even higher.

8   References

1. Boehnlein, M., Ulbrich vom Ende, A.: Business Process Oriented Development of Data Warehouse Structures. In: Proceedings of Data Warehousing 2000, Physica Verlag (2000)
2. Bulos, D.: OLAP Database Design: A New Dimension, Database Programming & Design, Vol. 9, No. 6, (1996)
3. Davenport, T. H.: Process Innovation – Reengineering Work through Information Technology, Harvard Business School Press, Boston (1993)
4. Hammer, M.: Beyond Reengineering, Harper Collins Publishers (1996)
5. Golfarelli, M., Maio, D., Rizzi, S.: Conceptual Design of Data Warehouses from E/R Schemes. In: Proceedings of the 31st HICSS, IEEE Press (1998)
6. Inmon, W. H.: Building the Data Warehouse. Wiley & Sons (1996)
7. Kimball, R.: The Data Warehouse Toolkit: Practical Techniques For Building Dimensional Data Warehouse. John Wiley & Sons (1996)
8. Kueng, P., Wettstein, Th., List, B.: A Holistic Process Performance Analysis through a Process Data Warehouse. In: Proceedings of the American Conference on Information Systems (2001)
9. Leymann, F., Roller, D.: Production Workflow. Prentice Hall (2000)
10. List, B., Schiefer, J., Tjoa A M., Quirchmayr, G.: Multidimensional Business Process Analysis with the Process Warehouse. In: W. Abramowicz and J. Zurada (eds.)**:** Knowledge Discovery for Business Information Systems, Kluwer Academic Publishers (2000)
11. Poe, V.: Building a Data Warehouse for Decision Support. Prentice Hall (1996)
12. Watson, H., Haley, B.: Managerial Considerations. In Communications of the ACM, Vol. 41, No. 9 (1998)
13. Westerman, P.: Data Warehousing using the Wal-Mart Model, Morgan Kaufmann (2001)