2000

Conceptual Architecture of Data Warehouses - A Transformation-Oriented View

Hans-Georg Kemper
University of Cologne, kemper@wi-im.uni-koeln.de

Conceptual Architecture of Data Warehouses – A Transformation-oriented View

Hans-Georg Kemper, Department of Information Systems and Information Management,
kemper@wi-im.uni-koeln.de, University of Cologne, Albertus-Magnus-Platz,
D-50923 Cologne, Germany

About the author

Hans-Georg Kemper is an assistant professor at the University of Cologne and works at the Department of Information Systems and Information Management. He holds a Ph.D. in MIS from University of Essen and passed his university lecturing qualification for business informatics at the University of Cologne in 1999 qualifying him for tenure positions. His research interests comprise the field of management support systems and related areas such as data warehousing, workgroup computing and electronic commerce.

Keywords:

data warehouse, conceptual architecture, transformation, meta data, controller and technician interface

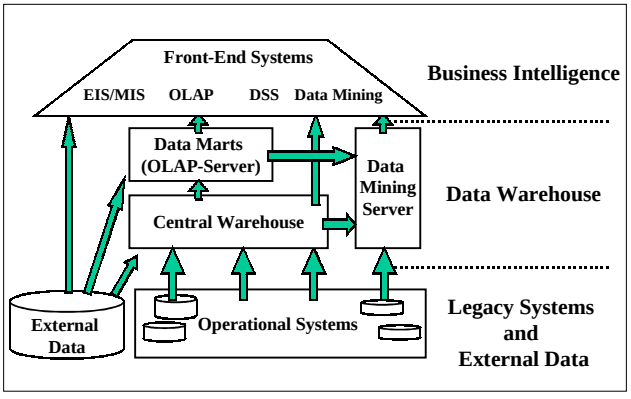
Abstract

A data warehouse (DWH) is an integrated collection of worthwhile data for management support. Unfortunately operational databases – the main source of feeding the DWH with internal data – only provide data with poor management value when no transformation has taken place. The paper deals with this topic. It concentrates on a conceptual architecture of a DWH, in which the *transformation machine* is the main constituent part of the architecture. This machine supports sub-processes of filtering, harmonization, aggregation and enrichment and is maintained by *controller and technician interfaces*. Furthermore an *access, load and meta data administration* handles the secure and documented loading and accessing of relevant data and guarantees technical and business-oriented documentation of all transformation activities.

Introduction

The colloquial meaning of the term *architecture* is related with planning or constructing houses, bridges and other buildings. About 20 years ago IT adopted the term. From this point on the term has been used for almost every aspect dealing with the structure of hard- and software systems. Architecture of processors, networks, databases or protocols are just some examples for this phenomenon (Hammergren 1996). This paper deals with a special kind of architecture as well, namely the architecture of data warehouses. These are systems, which *Inmon* defines as "... a subject-oriented, integrated, time-variant and non-volatile collection of data in support of management´s decision-making processes ..." (Inmon 1996). Usually the architecture of data warehouses is described by specifying the physical components and illustrating the way they are working together. Figure 1 gives an idea of such a technology-oriented view.

Figure 1: The physical architecture of a data warehouse



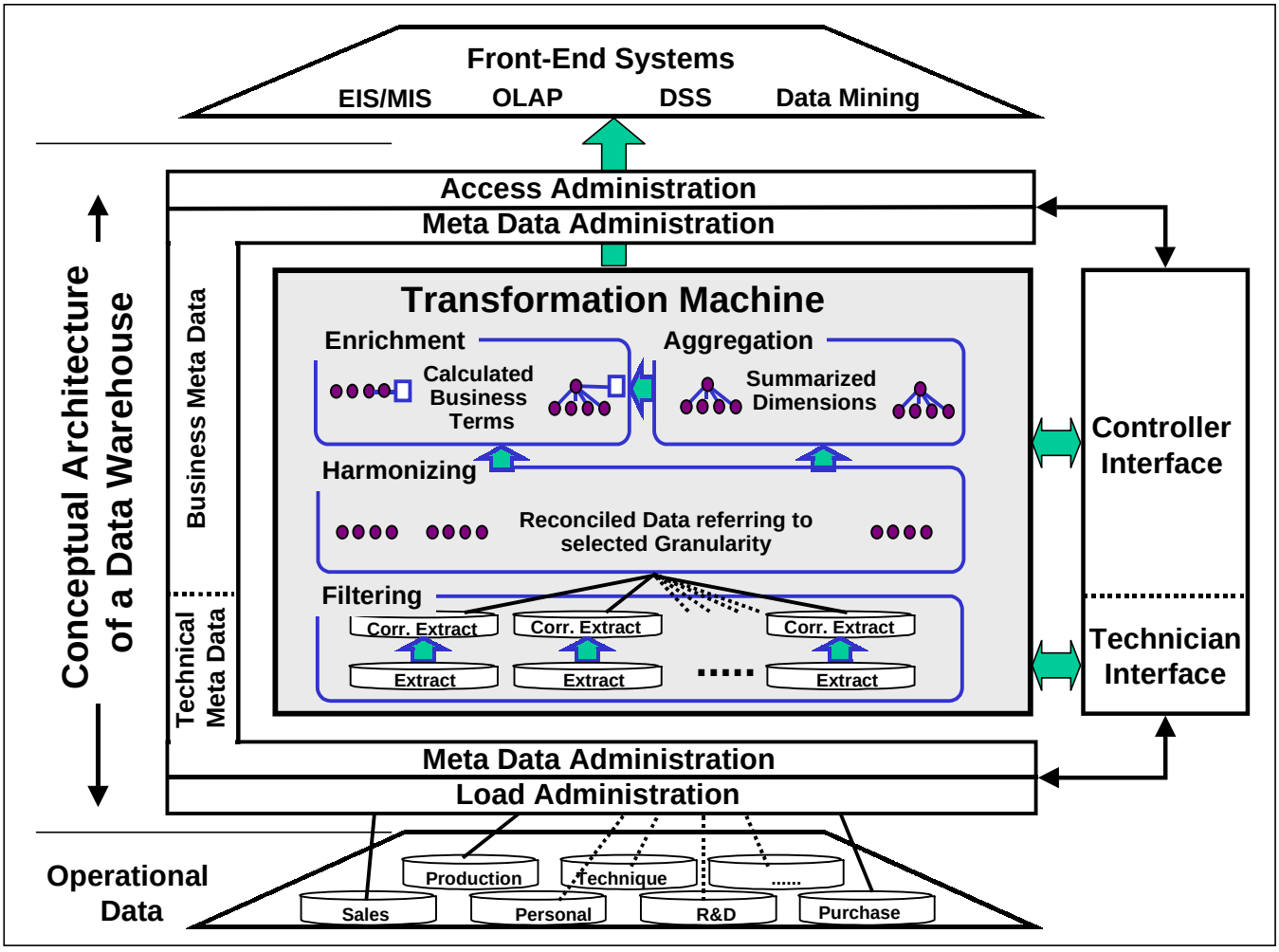
There is no doubt that building a data warehouse and integrating it into an existing infrastructure creates a variety of technical challenges. Therefore creating physically oriented architectures is very worthwhile and necessary for a successful implementation of a data warehouse (Berson, Smith 1997). But empirical research in the field of management support systems shows that other issues cause fundamental problems as well. A German long-term study proved that the majority of the companies had serious problems with transforming operative data into management information (Kemper 1999). In most cases these difficulties were heavily underestimated and often put the whole project at risk. That is why it is necessary to complement the traditional physical view with a conceptual, transformation-oriented approach on how operational data has to be converted to valuable management information. The following paper deals with this topic by creating an adequate conceptual architecture in which all essential transformation steps can be systematized, analyzed, and maintained.

The Conceptual Architecture

Figure 2 illustrates the conceptual architecture. It consists of three main components:

- the ***transformation machine*** as the core of the data warehouse,
- the ***access, load and meta data administration*** which handles a secure and documented access to relevant data for the user, allows the secure loading of operational data and guarantees technical and business-oriented documentation and
- the ***controller and technician interface***, which allows the necessary maintenance of transformation processes.

Figure 2: The conceptual architecture of a data warehouse



In the following paragraphs the three essential components are discussed in detail.

Transformation Machine

Systems supporting operational processes rely on data that normally differ in granularities, definitions, time periods and in the fields of the subjects they describe (Inmon 1996; Kelly 1996; Levin 1997). A large German company in the field of the chemical industry for example concluded that they were not surprised to find different definitions of earnings, costs or other economic references but that they were quite astonished to detect even things like varying ways on how to define the geographical subject *Europe* in the operational data (Kemper 1999). It is clear that merging operational sources and converting

them into information for management purposes has to be done individually in every company. These manipulations are the core tasks to be done in a data warehouse and arranging them is not a technical but a conceptual problem (Berg, Heagele 1997). Furthermore these activities are summarized and assigned to a so-called *transformation machine.* This machine manages 4 sub-processes namely the filtering, the harmonizing, aggregation and the enrichment process (Kemper, Finger 1999).

- *Filtering* groups the activities of data extraction and the correction of defects in syntax and semantics
- *Harmonizing* is the process of granularity adaptation and data merging into subject areas
- *Aggregation* is the summary of data to predetermined levels of detail
- *Enrichment* is the process of adding calculated business terms to the data

Filtering

The filtering process starts with the extraction of data out of operational data stores normally using modern extraction tools. From a conceptual point of view it is very important to check the quality of all operational data that could become reasonable sources of extraction. The main objective in this phase is to gradually gain a stable set of operational sources to feed the data warehouse. The answers to questions like these could help for a first estimation of quality:

- Does the source own attributes with compulsory input?
- Which attributes are input-checked by rules?
- Are the relevant attributes stored in a consistent way over time?
- How long does the source exist at all?
- And finally: Do plans exist to modify the structure of the source in the near future?

After selecting the operational sources the routine pro??cesses of periodical extraction start. During these extraction processes a variety of syntactic and semantic incompatibilities or defects can occur which have to be adapted or corrected (Tanler 1997; Inmon et al. 1999). These activities can be assigned to different classes.

1. class: automatic defect identification with automatic correction during extraction process
2. class: automatic defect identification with manual correction after extraction process
3. class: manual defect identification with manual correction after extraction process

An overview of the different classes and some examples are given in Figure 3.

Figure 3: The cleaning process

cleaning process	1. class automatic detection and automatic correction	2. class automatic detection and manual correction	3. class manual detection and manual correction
syntactic defects	familiar format incompatibilities	detectable format incompatibilities	—
semantic defects	missing of some data	"out of range" data / illogical constellation	unknown semantic defects in operational sources

short-term or medium-term correction needed in operational data sources

The first class of defects can be detected and corrected by implemented routines during the process of extraction. For instance it is possible to match incompatible formats by mapping-tables or to implement rules for replacing missing actual data by budget data. The defects of the second class can only be detected by implemented plausibility rules. A correction has to be made afterwards by either technical or business specialists depending on the nature of the defect. Semantic defects that can only be detected by humans after ending the extraction process belong to the third class. These mistakes and the semantic defects of the second class are always a reliable indication of errors in the operational database. Depending on the seriousness of these defects the operational databases have to be corrected in the short- or medium-term. But no matter what decision has been taken for the adjustment of operational data, it has to be ensured that the extracted data getting into the data warehouse are free of these defects, even by implementing interim methods of adaptation.

Harmonizing

After filtering the data out of the operational database the prerequisites for converting data into the first stage of management information are made (Mattison 1996). At this point the data has to be merged into so-called *subject areas* and adjusted to required granularity to become the most detailed level of data held in the data warehouse. The subject area *product analysis* for instance may need the economic terms (called *facts*) *sales* and *revenue* on the basis of a single *product group* per *month*, *store*, and *customer* (called *dimensions*). Merging the various already filtered operational extracts – here maybe the tables of the single *products* with the respective attributes *sales* and *revenue* on the basis of single invoices – implies that an adaptation in granularity and a harmonization in syntax and semantics has to be carried out (Inmon et al. 1998). In this example the files have to be summarized to a monthly basis and to a product-group oriented basis in order to harmonize the granularity. Furthermore for syntactic adaptation it is necessary to harmonize the keys as well as the coding of attributes. Last not least the homonyms *(attributes which own the same names but have different*

meanings) and the synonyms *(attributes which own different names but have the same meanings)* have to be eliminated in the extracted data files. These problems can be managed by implementing routines. They are mostly not critical.
More serious are the problems related to the activities of semantic harmonization. The main reason for this predicament is a historical one. In most companies a lot of business terms are used, which own the same name but are quite different concerning their definition and interpretation in the various business areas. Time-periods differ e.g. as well as the definitions of key factors like sales, revenue, profit etc. Merging data together in a data warehouse demands the harmonization of these terms across the enterprise or a group of business areas (English 1999). In the mentioned empirical research most of the companies stated that these harmonization activities caused the main political and cultural problems in their data warehouse projects (Kemper 1999).

Aggregation

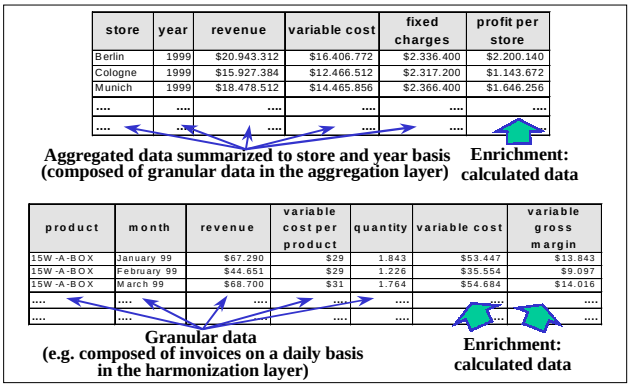
In the third layer of transformation the filtered and harmonized data will be expanded by implemented hierarchy structures. For this reason it is necessary to create special aggregation tables which comprise the ways granular data should be summarized over all involved dimensions. The dimension *store* for instance could be aggregated using the predetermined aggregation levels *district*, *region*, *area*, *country* and finally *total*. It is quite possible that one dimension owns various summary structures, so-called parallel hierarchies. The dimension *product group* for instance could be summarized by using the hierarchy structure *product main groups* or be alternatively summarized over *profit centers* to *total*. Remark: The question if any aggregated data should be stored physically in additional tables or should be calculated "on the fly" in the connection with each future query, is not discussed at this stage. This is rather a topic of tuning the database and depends on the nature of physical realization later on. (Kimball 1996; Kimball et al. 1998; Howard 1995)

Enrichment

The phase enrichment completely violates the paradigm of separation of logic and data. From this stage on data is no longer stored neutrally but it is oriented to the later class of application it supports, because business terms like revenue or profit are calculated and integrated into the stored data. This guarantees the consistence of the business terms on the basis of homogenous definitions used in the entire field of the relevant application class.

Figure 4 gives an example of the enrichment pro??cess in granular and aggregated data.

Figure 4: Enrichment in granular and aggregated data



Access, Load and Meta Data Administration

Access Administration

In contrast to the traditional approach the administration of user access is not a component of the end user systems but a constituent element of the data warehouse architecture itself. This concept enables consistent management of access authorization on the basis of single user rights or prepared profiles without any redundancies. Main issues of this centralized access administration are single users or user groups. Thus the user orientation allows the definition of standard rights for data access without consideration of the analysis systems or tools used for visualization. Furthermore it is naturally possible to restrict or widen standard access rights of the users in dependency of the tools or systems they are using. E.g. it could be possible to give controllers full rights to a data view by allowing them to use any tool they prefer but restrict the same data view to "read-only" for top managers and for exclusive use of an Executive Information System.

Load Administration

The task of the load administration is the secure handling of the periodical data warehouse refreshment due to ongoing changes in the operational environment (Anahory, Murray 1997). For operational data loading processes several techniques exist for incremental data extraction and copying. Normally the refreshment of the data warehouse is being done by using techniques like time-stamp procedures, creating and scanning logs and audit files or making use of snapshot methods. (Kimball et al. 1998)

Meta Data Administration

The meta data administration is closely linked to all procedures and activities concerning the data warehousing process (Devlin 1997; Hufford 1997). It is the main objective of this component to document technical and busi-

ness aspects of the transformation and all details of the user access and the data loaded out of the operational environment (Brackett 1996; Inmon, Hackathorn 1994). For that reason meta data is essential for

- technical staff while creating or editing the extracting and filtering routines,
- controllers while transforming operational data into management information and maintaining existing transformation procedures,
- developers while creating and maintaining information systems for the end users, and
- end users while using analysis tools like OLAP, Data Mining etc.

Controller and Technician Interface

The fact that transformation in data warehouses combines data and logic already by storing them together in a common information base implies that front-end systems need much less functionality than traditional systems. Therefore the role of these systems is confined to viewing and visualizing information only enriched by special functionalities at best. Accepting this new approach implies that most of the logic which was traditionally applied and documented in the coding of the information systems is now created and maintained in the data warehouse itself. Therefore it is essential to offer adequate interfaces to business-oriented and technical staff, here called controller and technician interface.

Technician Interface

The technician interface allows the specialists to create and edit all routines and procedures in the filtering layer of the transformation, whereas every modification of the filtering layer is documented in the meta data (pointed out by the respective arrow in Figure 2).

Controller Interface

The controller interface owns two main tasks. The first task is to offer the controller an adequate access to create new and maintain existing routines and tables for harmonizing, aggregating and enriching the data in the data warehouse. E.g. the controller should be enabled to generate or vary hierarchy trees, construct summarized tables or modify business terms. The second job of the controller interface is to offer the staff a suitable access to add equivalents to the transformed operational data, for example to add planned or budget values to data at any stage of transformation. Naturally every modification in the harmonization, aggregation and enrichment layer is documented in the meta data as well (pointed out by the respective arrow in Figure 2).

Conclusion

The concept of data warehouse is a new approach with considerable potential of improving the field of management support. It differs from traditional solutions by replacing isolated data areas oriented to special information systems with an integrated transformed data collection for a broader set of management applications. Since it merges data and logic by harmonization, aggregation and enrichment the traditional assignment of management support systems gets more and more obsolete. Today information systems rather offer special views on data of the warehouse and mostly need less features for data transforming than in former days. But actually this fact implies that the major part of technical and business knowledge has to be maintained in the data warehouse itself. This requires a high-quality meta data administration with extensive technical and business details, which go far beyond the information held in traditional data dictionaries of operational environments. Furthermore it is necessary to implement interfaces for technicians and controllers which enable them to assemble and edit all rules necessary for transformation and creating management information. Empirical research demonstrates that unfortunately these topics are often neglected in projects and cause a lot of trouble in building and using data warehouses.

References

Anahory, S.; Murray, D.: “Data Warehousing in the Real World”, Harlow 1997.

Barquin, R. C.; Edelstein, H. A.: “Planning and Designing the Data Warehouse”, New Jersey 1997.

Berg, D.; Heagele: “Improving Data Quality: A Management Perspective and Model”, in Barquin, R.; Edelstein, H.: “Building, Using, Managing the Data Warehouse, New Jersey 1997, pp.85-99.

Berson, A.; Smith, S. J.: “Data Warehousing, Data Mining & OLAP”, New York et al. 1997.

Brackett, M. H.: “The Data Warehouse Challenge – Taming Data Chaos”, New York et al. 1996.

Devlin, B.: “Data Warehouse from Architecture to Implementation”, Reading Massachusetts 1997.

English, L. P.: “Improving Data Warehouse and Business Information Quality”, New York et al. 1999.

Hammergren, T.:”Data Warehousing: Building the Corporate Knowledgebase”, Milford 1996.

Howard, P.: “Planning for performance in the data warehouse”, in Capacity Management Review, April 1995, pp. 1-11.

Hufford, D.: “Metadata Repositories: The Key to Unlocking Information in Data Warehouses”, in Barquin, R. C.; Edelstein, H. A.(eds.) “Planning and Designing the Data Warehouse”, New Jersey 1997, pp. 225-283.

Inmon, W. H.: ”Building the Data-Warehouse,” 2. edition, New York et al. 1996.

Inmon, W. H.; Hackathorn, R. D.: “Using the Data Warehouse”, New York et al. 1994.
Inmon, W. H.; Imhoff, C.; Sousa; R.: “Corporate Information Factory”, New York et al. 1998.
Inmon, W. H.; Rudin, K.; Buss, C. K.; Sousa, R.: “Data Warehouse Performance”, New York et al. 1999.
Kelly, S.: “Data Warehousing - the route to mass customization”, Chichester 1996.
Kemper, H.-G.: “Architektur und Gestaltung von Management-Unterstützungs-Systemen,” Stuttgart Leipzig 1999.
Kemper, H.-G.; Finger, R.: “Datentransformation im Data Warehouse”, in Chamoni, P.; Gluchowski, P.: Analytische Informationssysteme, Berlin et al. 1999.
Kimball, R.: “The Data Warehouse Toolkit”, New York et al. 1996
Kimball, R.; Reeves, L.; Ross, M., Thornthwaite, W.: “The Data Warehouse Lifecycle Toolkit”, New York et al. 1998.
Levin, E. J.: “Developing a Data Warehouse Strategy”, in Barquin, R. C.; Edelstein, H. A.(eds.) “Planning and Designing the Data Warehouse”, New Jersey 1997, pp. 53-89.
Mattison, R.: “Data Warehousing - Strategies, Technologies, and Techniques”, New York et al. 1996.
Tanler, R.: “The Intranet Data Warehouse”, New York 1997.