

DATABASE REPORT MODELING

Ladislav Ruttkay

Doctoral Degree Programme (2), FIT BUT

E-mail: iruttkay@fit.vutbr.cz

Supervised by: Tomáš Hruška

E-mail: hruska@fit.vutbr.cz

ABSTRACT

This document describes problems rising in the field of database reporting. It handles about the wide area of so called ad-hoc reports, which are usually used by reporting tools. These reports are so wide, that they are proposed to be designed by database and schema experts and not by casual users. Because of that fact, this document tries to introduce new approach, of so called canned reports, designed by report models. It describes the wide area of advantages of predefined reporting models, and also introduces the object model of the reporting meta-model, which builds the field for further development. The report model, describes possibilities for creating reports even from not normalized databases, and users dependent reports. At the end it describes the further area of possible development and used technologies.

1. INTRODUCTION

Data of all kinds plays significant role in all areas of the society. It becomes subject of handle, workflow, statistics and evaluations, etc. It is not just data by it selves, but the information gained from this data, effectively visualized and understood.

The area of creating databases reports plays a significant role in the visualization. There exist many systems and technologies supporting the database reporting. Basic problem, which comes with this approach, is the wide range of possibilities, provided by such systems. Basic user, without any knowledge about the database, its schema and data has hardly chance to create a meaningful report. This job is usually carried out by business intelligence experts.

An effective solution to this problem is the report modeling approach. The report model contains the whole range of possibilities for creating a report. It recommends appropriate dimensions for being displayed to gain meaningful results. Not only dimension connections are part of the report model, it is able to contain many other recommendations, which will be discussed in this paper later.

2. REPORT MODEL

The report model represents in fact the metadata for the future created report. A report created from a report model is called canned report.

2.1. USED TECHNOLOGIES

The approach of creating a report meta-model is partly used in some common technologies. Nearest is the Reporting Model of Microsoft SQL Server. This model is generated from the database structure and represents the basic concept and connections in the database. The model is represented by XML language, and builds the base for the report builder application. Disadvantages of this model, is its complicated and complex creation, and it does not contain normalization or user based approach.

Other technologies, like Oracle and Cognos, are using XML as the base of a concrete report, but not the meta-model of the report. Cognos is considering this approach, but it is not used in any concrete products so far.

We can conclude used technologies displayed on the Image 1. The widest approach is the ad-hoc reporting, what represents the basic reporting, in the way of having possibility to display different dimensions together, with no restrictions in the meaning of the data. The next approach displayed is the MS reporting model, which is not as wide as ad-hoc reporting, because it is having some report model, but the range of displayed reports for final user is far too wide. The approach of our consideration is the modeled or canned report. This approach recommends and leads the user to create a single report, which may be his goal.

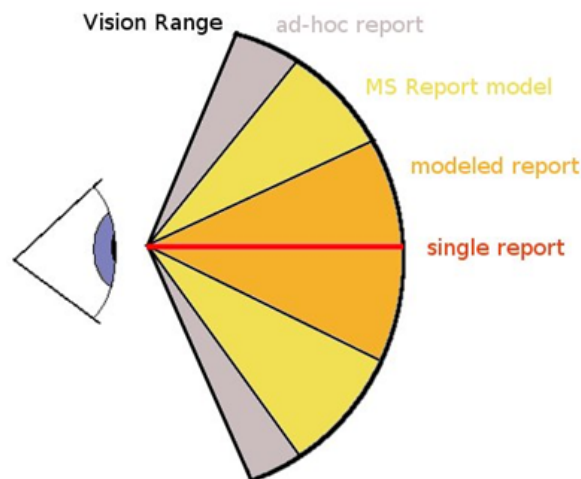


Image 1: Reporting options

2.2. COMPONENT MODEL

How does the final report shall be displayed? The final report usually has to be displayed on the screen or printed and evaluated. Usually it has to hold some corporate identity, and therefore define some header and footers, of the report or page printed. All these aspects have to be in the reporting model considered. Therefore the report model is not just compound of data specifications, but also of the layout specifications, which considers the above described parts.

Basic parts of the component model of the report meta-model, are: report header and footer, page header and footer and the data part. Each of these sections is considered to be in the report model represented as a collection of different recommendations of the header, footer, etc. Each company shall have own corporate identity in the consideration of this created footers, headers, etc. These parts will contain just the data, like company name, report name, or functions like the actual date, or actual user. The style will be assigned by a

different style object which can be associated to this part. The style technology can be different like: CSS styles, XSLT, or other technologies.

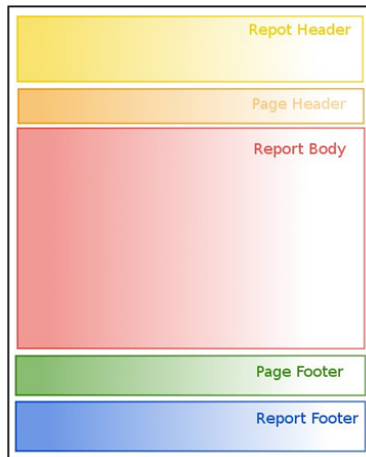


Image 2: Component model

2.3. OBJECT MODEL

The object model represents compound parts, which shall be found in the final reporting meta-model. It describes the structure of the whole model. There can be found parts which are describing and containing data for the Component model. Basically there is described the data component and its different sub-structures, which will be introduced later.

In the given Image 5, there are two basic component kinds. The first represents the single component and the second represents a compound component, or better called: collection of components.



Image 3: Single component

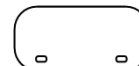


Image 4: Collection component

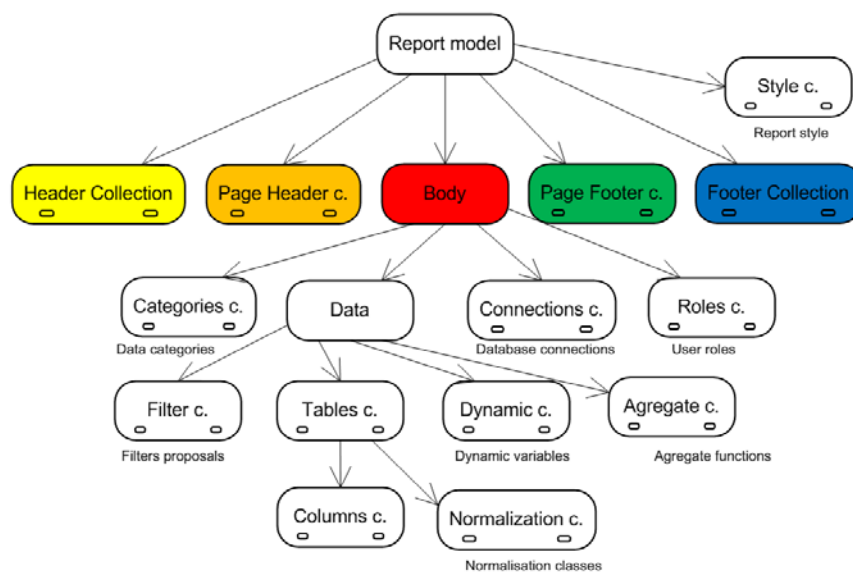


Image 5: Object model

2.4. DATA CATEGORIES

Some tables and filters are often associated in a category, in the mean that the subject handled by them is the same. These subjects are called categories. In some database approaches it is considered to be the Schema of a table, view, user, etc. That means, that if the base model is created of such a database, than the data categories collections and its connections to tables can be created automatically.

Kat. / Tab.	HR	Sales	Marketing
Employee	X		
Customer		X	X
Product		X	X
Department	X		X
Factures		X	

Table 1: Data categories example

2.5. USER ROLES

There is also for consideration a report model, which will introduce report possibilities according to connected user, which is just creating the report. This consideration is introduced in the user roles approach. There are created user roles, which can be associated with the operating system groups or users, and these roles are connected with the data categories, which represent, in fact, the area of interest of the users.

Kat. / Role.	HR	Sales	Marketing
Management	X	X	X
Assistents			X
Production		X	

Table 2: User roles example

2.6. NORMALISATION CLASSES

The next important component of the model is normalization classes. Some databases may not be normalized, in the way, that one table may contain many dimensions from different areas, which could be considered to be placed in different tables. The normalization component enables to categorize such attributes, to be better displayed and interpreted in the final report creation.

Product	
ProductID	
Product Name	
Product Category	Product
Product Price	
Description	
Sales Date	Date
Producer	
City	Producer
Address	

Table 3: Normalisation example

2.7. TABLE ATTRIBUTES

Table attributes define actual connections to User roles and Data categories. They also contain possible alias name for the final table representation. It also contains recommendations

for sorting data in the table, or the recommended dimensions for possible grouping, or aggregation functions.

2.8. COLUMN ATTRIBUTES

Column attributes define connections to other attributes from different tables (foreign key) and options of that connection. Other attributes may define connections with filters, or data changing formats. A very developed possibility is to apply a dynamic variable, which represent a web service function. This can gain actual data from the internet, like conversion curse, and than use it in the column value evaluation. There can be other options in the attributes considered, like: aliases, data formats, constraints, defaults, etc.

3. TOOLS SCHEMA

There shall be applications developed to support this report meta-model. These applications will build the basic model and enable the editing of this model. Final application shall parse the report model and display for the user all proposed options for the final report creation.

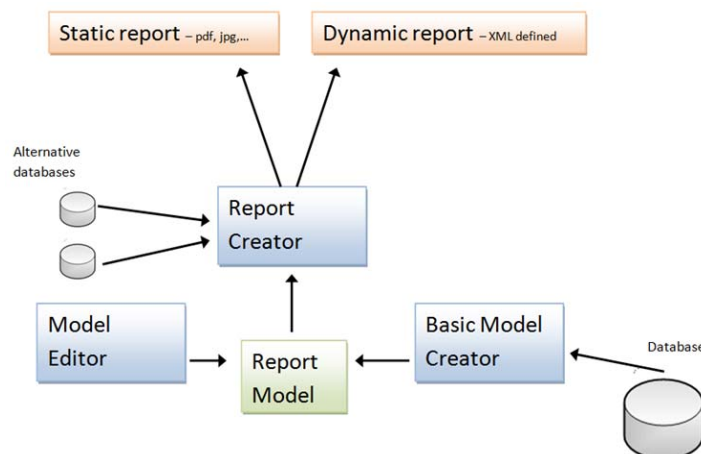


Image 6: Tools schema

4. CONCLUSION

Aim of this whole model is to create the basic for further development of applications. The final report model can be represented in considered open technologies like XML.

The goal of this approach is to make reporting for final users easier and develop simple approach to the reporting problem.

REFERENCES

- [1] HOONG, L.: RepCom: Report Generator Component System using XML-driven, Component-based Development Approach, University of Malaya, 2003
- [2] Col.: Report definition Language Specification, Microsoft, Third version, 2008
- [3] RUTTKAY, L.: Report Model, FIT VUT, 2008