PREFACE

The role of the various information systems is growing in every area of management. Nowadays information systems are extremely important in decision making, strategy planning, forecasting, market research, as well as in risk assessment, modeling, and analysis. All managers of organizations and enterprises to meet challenges and competition have to utilize information technologies in their work.

In this monograph we present results of various research based on different methodologies and utilizing many interesting cases, often directly related to business practice and real-life management problems, showing theoretical and practical importance of the subject.

Current volume consists of 10 papers written by authors coming from different institutions. It provides a rich source of ideas, solutions, and perspectives. I do believe that presented results will be useful to all parties dealing with information systems in management: researchers, experts, and business practitioners, including managers themselves. The chapters are ordered alphabetically, according to the surnames of the first-named authors.

The review of methodology and comparison of available CASE tools (Computer-Aided Software Engineering) that use UML 2.0 notation in the process of object-oriented modelling of information systems is performed (J. Becker). IT solutions for supporting strategic controlling which are parts of packages described as Business Performance Management (BPM) or Strategy Management solutions are presented; especially their features, functions and architecture are described and illustrated with examples (A. Januszewski). The explanation why logistics concepts are so important is made and the description of ERP systems that provide logistics solutions to a company described (K. Karpio, A. Orlowski, M. Przybyć). A process of designing a data warehouse creation is presented and business and logical models developed in this process are discussed in detail (W. Karwowski, W. Kwasiowiec). The key aspects to take into account, when considering implementation of business intelligence software by Small to Medium sized companies to lead to cheaper and reliable solution at the same time are presented and discussed. (R. Nafkha, M. Olejniczak). The modeling of business processes with the established graphic notations, i.e. Unified Modeling Language UML and Business Process Modeling Notation BPMN are presented and the possibilities of the both notations is compared with respect to methodologies supporting their usage and the notation elements together with supporting tools (S. Robak). Some aspects of a decision-support system, which takes into account selected considerations that are necessary
in a choice of a trade customer are presented and descriptions on the basis of the fuzzy logic some of the characteristics of the system which under investigation are imprecise or unpredictable are outlined (S. Robak, A. Pieczyński). A processing native XML data in relational database systems; especially SQL Server 2008 and Oracle 11g is described and a study and comparison on efficiency of XQuery queries performed on the test relational database containing weather information provided by National Oceanic and Atmospheric Administration are presented (M. Śmietanański, M. Drozd). Problems related to the capabilities of the integration of XML data with relational data in databases management systems were discussed, particularly issues, which give developers of database software the possibility to adapt the mechanisms of storage and processing simultaneously native XML and relational data are described and compared on Oracle 11g and Microsoft SQL Server 2008 (M. Śmietanański, M. Drozd). Eventually, fundamental definitions related to data and information quality in the context of data completeness are described and application of artificial neural network for the purpose of practical experiment that aimed to estimate data completeness in corporate Management Information System is presented (T. Ząbkowski, F. Pudło).

Arkadiusz Orłowski
TABLE OF CONTENTS

BECKER J.
COMPARATIVE ANALYSIS OF CASE TOOLS FUNCTIONALITY
COMPATIBLE WITH UML 2.0 NOTATION ................................................................. 7

JANUSZEWSKI A.
IT SOLUTIONS FOR SUPPORTING STRATEGIC CONTROLLING ............................ 17

KARPIO K., ORŁOWSKI A., PRZYBYĆ M.
DATA LOGISTICS IN INFORMATION SYSTEMS ...................................................... 28

Kwasowiec W., Karowski W.
DATA WAREHOUSE DESIGN ................................................................................ 46

NAFKHA R., OLEJNICZAK M.
BUSINESS INTELLIGENCE SYSTEMS – IMPLEMENTING OPPORTUNITIES
IN THE WORLD OF SMALL TO MEDIUM BUSINESS ........................................... 56

ROBAK S.
BUSINESS PROCESS MODELING – METHODS AND TOOLS .................................. 64

ROBAK S., Pieczyński A.
THE SELECTION PHASE IN DECISION MAKING AIDED WITH
A DECISION SUPPORT SYSTEM ........................................................................... 72

ŚMIEĆAŃSKI M.J., Drozd M.
EFFICIENCY OF XQUERY IN PROCESSING NATIVE XML DATA STORED IN
RELATIONAL DATABASE ......................................................................................... 81

ŚMIEĆAŃSKI M.J., Drozd M.
SELECTED ASPECTS OF THE INTEGRATION OF XML DATA
WITH RELATIONAL DATA IN DATABASE SYSTEM .................................................. 91

ZABKOWSKI T., PUDLO F.
DATA COMPLETNESS ESTIMATION IN MANAGEMENT INFORMATION SYSTEMS ...... 100
COMPARATIVE ANALYSIS OF CASE TOOLS
FUNCTIONALITY COMPATIBLE WITH UML 2.0 NOTATION

Jarosław Becker
Higher Vocational State School in Gorzów Wlkp
Technical Institute

Abstract: In the article was conducted the review of methodology and comparison of available CASE tools (Computer-Aided Software Engineering) that use UML 2.0 notation in the process of object-oriented modelling of information systems. The result of conducted researches includes the ranking of CASE tools functionality that allows distinguishing leaders in a group of commercial (chargeable) and open source (free of charge) products. In the researches was used a simple additive method of assigning suitable weight with SPMC model (Single Participant Multiple Criteria).

Key words: formal methodology, agile methodology, CASE tools, UML 2.0, object-oriented modelling of information technology systems.

1. INTRODUCTION

Since the half of the 1990s, just with the increasing popularity of the object-orientated programming, there has been observed a major development of methodology and modelling tools of information technology systems (CASE) compatible with the object-orientated paradigm. Since then the structural modelling – has been more and more often driven out by UML (Unified Modelling Language). Technological development directed towards the object-oriented approach contributed to the popularity of real time systems and Internet systems. The development included such categories as: e-business, e-learning or e-health. Huge influence on the stimulation of object-oriented technology had integration processes of management information technology systems used in practice for such areas as trade, banking, logistics and education. [Wrycza S. and others]

The aim of the article is the comparative analysis of CASE tools functionality that support modelling of information technology systems in UML 2.0 notation. Research results include the ranking of CASE tools and it was prepared from the view of DSS generator engineering needs (decision support systems). It is to distinguish solutions which in the best way support the adaptation process of generator construction and allow precisely coping and documenting its architecture with the use of the round-trip engineering.
2. THE METHODOLOGY OF THE OBJECT ORIENTED MODELLING OF INFORMATION TECHNOLOGY SYSTEMS

The object-oriented attitude towards designing in comparison with the structural methodology enables creating large and complex information technology systems by numerous teams. According to authors of works: [Wrycza S. and others.], [Subieta K.], [Dolińska M.] and [Ksielnicki J., Sroka H.] it is popular and it dominates in practice. In 1989-1994 the object-oriented attitude was characterized by numerous methodologies and notations. The cooperation of three authors: J. Rumbaugh, G., Booch (1995) and I. Jacobson (1996) resulted in the construction of UML prototype, the unified modelling language of technology information systems, which was at first marked with UM (Unified Method).

In practice, UML language has adopted the form of graphic representation of constructed system. It consists of connected diagrams in a logic way which enables system description from general models to very detailed models. Works of the tram created of three scientists [Booch G., Jacobson I., Rumbaugh J.] on UML language, were directed towards the standardisation of the object-oriented system designing and resulted in USDP generic methodology (Unified Software Development Process) in 1998. The term generic means in this case a possibility of forming its various configurations and implementation. [Wrycza S. and others, p. 319] This methodology is a set of terms and directives and it is directed towards modelling cases of use and towards concentration on system architecture as the main issue in the process of software designing. The main process assumption is its iterative and incremental character.

USDP implementation is RUP methodology (Rational Unified Process) [Kruchten P.] [IBM, Rational Unified...] that was designed by Rational Software. It dominates in the area of formal, object-oriented methodology of creating systems compliant with the notation of UML language and it is a competitive solution for OPEN methodology (Object-oriented Process, Environment, and Notation) which was created as a result of combining over twenty international methodologies and their ideas (e.g.: MOSES, SOMA, Firesmith and Synthesis, BON, OOram, UML). [Henderson-Sellers B.]

Among other object-oriented methods, which are currently being developed and which are also based on the assumptions of USDP generic methodology, we may distinguish: XP (eXtreme Programming) [Jeffries R.] [Kaczmarski K.], AM (Agile Modeling) [Ambler S.], DSDM (Dynamic Systems Development Method) [Davies R.], FDD (Feature Driven Development) [Rybus R.], Scrum [Scrum Alliance, What...], Crystal [Cocburn Alistair A. R.]. They are assigned to the group of light methodology and are called agile methodology because of the time limit for modelling and documenting analytical-design works in order to lead quick creation of the system source code. First of all, USDP methodology was mainly used by the iterative and incremental system lifecycle as well as by model-
ling categories of UML language. The deft reaction to changes is their feature which results from good communication in the team and the whole process adaptation to changeable conditions. Their most important product is working and tested system code.

The agile methodology must be reinforced in the sphere of practice. Large-scale undertakings of information technology, which are connected with large complex systems (developed and modified for many years), require such methodology of project running which accurately reflects and defines every process element of software creation (such as RUP methodology). Using formal (hard) methodology in this case is necessary just because of the scale and potential duration of the project. It is to provide security for the undertaking and achieve intended aims. It can be thought with a real change that in the near future the professional level will increase among proponents of light methodology. [Bartyzel M.] It will handle great undertakings provided that they provide a higher control level over work on the project in every moment of its duration without losing its agile activity.

3. FUNCTIONALITY COMPARISON OF CHOSEN CASE TOOLS

UML language has got considerable support in a form of CASE tools [Malina W., Szwoch M.] which provide help in modelling and creating suitable project documentation. Among the most important functions that prove the advanced position of a particular CASE tool and provide a suitable control level which meets user’s requirements in the project we may distinguish: data dictionary (repository of metadata with the management system), graphic notations editor (diagrams editor), correctness control module (searching mistakes in diagrams and repositories), project quality evaluation module (e.g. complexity scale evaluation and evaluation of connections between project elements), generators of reports and technical documentation, source code generators, user’s interface designing module, reverse engineering module (possibility of displaying data dictionary or diagrams on the basis of the source code or data base structure), import and export module of data as well as group work management module. [Chmielarz W.] [Bednarczyk M.]

Introducing 2.0 version of UML language caused the necessity of developing CASE tools. Currently many programs support this version, however, there is a wide range of solutions, especially non-commercial, which stayed in the standard 1.4 UML.

In the comparison there were qualified solutions available and spread in the local (academic) environment. Among commercial CASE tools there were chosen the following: Enterprise Architect (version 7.1) enterprises Sparx System [Sparx System, Enterprise…], Poseidon for UML (version 6.0.2) enterprises

Table 1. Comparison of CASE tools – part 1.

<table>
<thead>
<tr>
<th>Comparative criterion</th>
<th>CASE tools</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Enterprise Architect version 7.1</td>
</tr>
<tr>
<td>Version UML</td>
<td>2.0</td>
</tr>
<tr>
<td>sdl</td>
<td>+</td>
</tr>
<tr>
<td>cdi</td>
<td>+</td>
</tr>
<tr>
<td>pdo</td>
<td>+</td>
</tr>
<tr>
<td>cdi</td>
<td>+</td>
</tr>
<tr>
<td>cdi</td>
<td>+</td>
</tr>
<tr>
<td>cdi</td>
<td>+</td>
</tr>
<tr>
<td>pdo</td>
<td>+</td>
</tr>
<tr>
<td>sim</td>
<td>+</td>
</tr>
<tr>
<td>sdi</td>
<td>+</td>
</tr>
<tr>
<td>cdi</td>
<td>+</td>
</tr>
<tr>
<td>cdi</td>
<td>+</td>
</tr>
<tr>
<td>cdi</td>
<td>+</td>
</tr>
<tr>
<td>SUM</td>
<td>13</td>
</tr>
<tr>
<td>Platform</td>
<td></td>
</tr>
<tr>
<td>MS Windows</td>
<td>+</td>
</tr>
<tr>
<td>Linux</td>
<td>–</td>
</tr>
<tr>
<td>Mac OS</td>
<td>–</td>
</tr>
<tr>
<td>SUM</td>
<td>1</td>
</tr>
<tr>
<td>Source code generation</td>
<td></td>
</tr>
<tr>
<td>Cobra IDL</td>
<td>–</td>
</tr>
<tr>
<td>+C++</td>
<td>+</td>
</tr>
<tr>
<td>C#</td>
<td>–</td>
</tr>
<tr>
<td>Delphi</td>
<td>+</td>
</tr>
<tr>
<td>Java</td>
<td>+</td>
</tr>
<tr>
<td>F#</td>
<td>+</td>
</tr>
<tr>
<td>Visual Basic (VB)</td>
<td>+</td>
</tr>
<tr>
<td>VB.NET</td>
<td>+</td>
</tr>
<tr>
<td>SQL</td>
<td>+</td>
</tr>
<tr>
<td>Number of other languages</td>
<td>3</td>
</tr>
<tr>
<td>SUM</td>
<td>10</td>
</tr>
<tr>
<td>Reverse engineering</td>
<td></td>
</tr>
<tr>
<td>Cobra IDL</td>
<td>–</td>
</tr>
<tr>
<td>+C++</td>
<td>+</td>
</tr>
<tr>
<td>C#</td>
<td>–</td>
</tr>
<tr>
<td>Delphi</td>
<td>–</td>
</tr>
<tr>
<td>Java</td>
<td>+</td>
</tr>
<tr>
<td>F#</td>
<td>+</td>
</tr>
<tr>
<td>Visual Basic (VB)</td>
<td>+</td>
</tr>
<tr>
<td>VB.NET</td>
<td>+</td>
</tr>
<tr>
<td>Number of other languages</td>
<td>–</td>
</tr>
<tr>
<td>SUM</td>
<td>7</td>
</tr>
</tbody>
</table>

Source: Own preparation.
Table 2. Comparison of CASE tools – part 2.

<table>
<thead>
<tr>
<th>Comparative criterion</th>
<th>CASE tools</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>XMI / XML</td>
<td>+</td>
<td>–</td>
<td>–</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>–</td>
<td>–</td>
<td>5</td>
</tr>
<tr>
<td>Rational Rose - MDL</td>
<td>–</td>
<td>+</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>+</td>
<td>–</td>
<td>–</td>
<td>3</td>
</tr>
<tr>
<td>Java - JAR files</td>
<td>+</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>+</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>3</td>
</tr>
<tr>
<td>Number of other formats</td>
<td>–</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td></td>
</tr>
<tr>
<td>SUM</td>
<td>3</td>
<td>3</td>
<td>0</td>
<td>3</td>
<td>5</td>
<td>5</td>
<td>2</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>XMI / XML</td>
<td>+</td>
<td>–</td>
<td>–</td>
<td>+</td>
<td>+</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>2</td>
</tr>
<tr>
<td>POF</td>
<td>–</td>
<td>+</td>
<td>–</td>
<td>+</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>1</td>
</tr>
<tr>
<td>CSV</td>
<td>+</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>1</td>
</tr>
<tr>
<td>Number of other formats</td>
<td>–</td>
<td>–</td>
<td>1</td>
<td>2</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td></td>
</tr>
<tr>
<td>SUM</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>EPS</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>3</td>
</tr>
<tr>
<td>GIF</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>4</td>
</tr>
<tr>
<td>JPG</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>6</td>
</tr>
<tr>
<td>PNG</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>5</td>
</tr>
<tr>
<td>SVG</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>5</td>
</tr>
<tr>
<td>WMF</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>3</td>
</tr>
<tr>
<td>BMP</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>4</td>
</tr>
<tr>
<td>EMP</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>3</td>
</tr>
<tr>
<td>POF</td>
<td>+</td>
<td>–</td>
<td>–</td>
<td>+</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>+</td>
<td>9</td>
</tr>
<tr>
<td>Number of other formats</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>1</td>
<td>1</td>
<td>–</td>
<td>–</td>
<td></td>
</tr>
<tr>
<td>SUM</td>
<td>6</td>
<td>8</td>
<td>6</td>
<td>2</td>
<td>5</td>
<td>5</td>
<td>4</td>
<td>4</td>
<td></td>
</tr>
</tbody>
</table>

Source: Own preparation.

In the comparative analysis there was used a simple, additive method of assigning suitable weight with SPMC (Single Participant Multiple Criteria) model. The model, a single decision-maker, consists of a set of possible variants (or actions) \{A1, A2, ..., Am\} as well as a set of criteria \{C1, C2, ..., Cn\} among which every criterion is connected with choice preferences of defined variant. The aim of analysis is to receive a ranking of variants on the basis of given criteria.

Results of the detailed comparison of CASE tools were collected in table 1 and 2. They are a self-study and were prepared on the basis of information from producers’ websites and electronic documentation attached to available software versions (full, test and demonstration versions). In order to provide complete expression of partial criteria in a form of numerical values, their choice was determined by data availability. Systems evaluation which sums up every category (main criterion) is showed with the use of a five-score scale (0-5). Criteria were assigned suitable values of preferences (weight). In the summary there is calculated the sum and the weighted average of evaluation by scores whose values reflect taken place in the ranking (tab. 3).
The first category of the comparative analysis of chosen CASE tools is a possibility of reflecting particular types of UML diagrams. From the comparison (tab. 1) results that full support for UML 2.0 is provided only by three programs: Enterprise Architect, Poseidon for UML and Visual Paradigm. While designing small information technology systems there is needed very often only a minimum number of diagrams, available in UMLet program. From the point of view of the needs of scientific approach towards documenting the project and research works on DSS generator – whose outputs are specialized information technology systems with different information structures and decision models – providing UML 2.0 standard is justified (the value of weighted rate = 1; tab. 3).

Table 3. Evaluation of CASE tools.

<table>
<thead>
<tr>
<th>CASE tools</th>
<th>Range of diagrams supported</th>
<th>Compatibility of system platforms</th>
<th>Source code generation</th>
<th>Reverse engineering</th>
<th>Data import</th>
<th>Data export</th>
<th>Diagrams export</th>
<th>Weighted average of scores (0-5)</th>
<th>WEIGHT (preference for a criterion)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enterprise Architect version 7.1</td>
<td>13</td>
<td>3</td>
<td>7</td>
<td>3</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>21.85</td>
<td>4.37</td>
</tr>
<tr>
<td>Poseidon for UML version 6.0.2</td>
<td>13</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>20.63</td>
<td>4.12</td>
</tr>
<tr>
<td>Rational Software Architect version 7.5.1</td>
<td>9</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>18.35</td>
<td>3.67</td>
</tr>
<tr>
<td>Microsoft Visio version 2007</td>
<td>7</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>17.2</td>
<td>3.64</td>
</tr>
<tr>
<td>Visual Paradigm for UML version 6.3</td>
<td>7</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>16.85</td>
<td>3.64</td>
</tr>
<tr>
<td>Rational Software Architect version 7.5.1</td>
<td>9</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>15.36</td>
<td>3.64</td>
</tr>
<tr>
<td>Rational Software Architect version 7.5.1</td>
<td>7</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>14.78</td>
<td>3.64</td>
</tr>
<tr>
<td>Rational Software Architect version 7.5.1</td>
<td>6</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>13.2</td>
<td>3.64</td>
</tr>
</tbody>
</table>

*Source: Own preparation.*

For another decision criterion was chosen the availability of software version oriented to various system platforms (operating systems). In case of all analyzed solutions, the support for Windows systems can be taken as a common practice (tab. 1). Linux is also a quite popular platform. Because of long standing domination of Windows systems that are widely spread in the decision maker’s environment, this requirement was highly evaluated and it received 3 scores (other systems 1 score per each). On the other hand, in order to lower the importance of the availability of other versions (Linux, Mac OS) in the ranking, the criterion received a low value of weighted rate which is equal to 0.5 (tab. 3).
Another very important aspect of supporting the information technology engineering is as first a possibility of generating the stub code and as second the ability to create or modify a system model on the basis of the application source code (reverse engineering). The combination of these two functions gained the term round-trip engineering and it enables maintaining the cohesion of the system code for its model. In the ranking these properties are highly preferred (the value of weighted rate = 1; tab. 3) because they streamline the middle-out adaptation designing that is used by the decision maker of the ranking. The main issue of such an approach, opposed to the idea of top-down and bottom-up, is to focus attention on the quick preparation a system prototype and next on its further development, adding new functions or procedures till the moment of forming the shape of a stable system which is able to be moved. The function of the reverse engineering in such approach exempts us from making alternations in the project documentation after frequent changes in the system code. On the stage of applying the generator of information technology systems, especially valuable seems to be a possibility of changing the source code of generated systems which differ between each other into the model form. The ability to analyze CASE solutions in the discussed rage is showed by table 1. It is clear that the number of supported standards (programming languages and others) for a particular tool within the generated source code is a bit higher than in case of the reverse engineering.

Another function of CASE tools, which was taken into consideration in the ranking, is possibilities of importing and exporting data files which were assigned an average preference value (weighted value = 0,6; tab. 3) because of their lower usefulness in comparison with above described criteria. A dominating format of data exchange is XMI/XML, except UMLet and partially Rational Software Architect which are deprived of a function of files import. A leader in terms of bidirectional data exchange is Visual Paradigm. ArgoUML also has got a similarly developed import function. Interesting is the fact of having MDL files reading interface from the professional package of IBM (family of Rational Rose programs) by three programs what was distinguished in table 2.

The last decisive criterion in the comparative analysis is a function of generating diagrams in a graphic format (JPG, PNG, SVG etc.; tab. 2). The weighted value for every category is very low (equals 0,3) because it does not have any influence on the modelling quality. Apart from that, there are many programs in the market for graphic files conversion and in operating systems there is the ability to transfer graphics by a system buffer (clipboard), however, with a loss of quality in printing. Leaders in this category are Poseidon for UML, Enterprise Architect and Rational Software Architect, however, the last position, what can be surprising, is held by Microsoft Visio 2007 (tab. 3).
5. CONCLUSIONS

Calculated weighted sums of score evaluations that are placed in table 3 reflect the place taken in the ranking. Results were rearranged in a form of decreasing sequence and presented in graph 1. On the podium (three first positions) are commercial solutions. A very high position of ArgoUML tool (fourth place) which belongs to the open source category what is connected amongst others with a full lack of license fees. UMLet took the last position, gaining 5,6 scores what is almost four times less than the favourite. It can be regarded as a simple graphic program for drawing UML diagrams.

Figure 1. Results of CASE tools ranking.

The analysis of ranking results in a form of weighted average evaluations that are within taken score scale from 0 to 5 scores, there can be made an attempt to divide tools and distinguish their few classes which differ from each other by a level of functional advance. Near 4,5 scores there are gathered three programs that hold leading places in the ranking. It is a class of over-average solutions directed towards professionals. Systems that hold following three position in the ranking (ArgoUML, Rational Software Architect and Microsoft Visio 2007) are around 3,5 scores and they can be called average solutions which meet general modelling standards. Below the average evaluation (about 2,5 scores) there is StarUML which is assigned to good solutions of the open source class. The last and separate group
is a very simple graphic program UMLet, whose weighted average evaluation is close to one.

Received ranking results are rather subjective, however, they may be a point of reference for similar analyses. A final choice can be made after testing all of chosen CASE tools. Moreover, all producers of programs placed in the ranking made their versions available to be checked.

REFERENCES


18. Malina W., Szwoch M. (2008), Metodologia i techniki programowania, PWN/MIKOM.


22. Scrum Alliance, What is Scrum? (http://www.scrumalliance.org/pages/what_is_scrum)


IT SOLUTIONS FOR SUPPORTING STRATEGIC CONTROLLING

Arkadiusz Januszewski

University of Technology and Life Science in Bydgoszcz, Management Faculty, Department of Management Engineering

Abstract: The article presents IT solutions for supporting strategic controlling. They are parts of packages described as Business Performance Management (BPM) or Strategy Management solutions. BPM suites complete integrated transactional ERP systems. They include financial planning, budgeting and forecasting module, cost and profitability module, strategy management module and consolidation and statutory reporting module. The author describes features, functions and architecture of BPM solution and gives an example of SAS Strategy Performance Management package, which belongs to the most advanced BPM systems on the market.

Key words: Strategic Controlling, Business Performance Management, Corporate Performance Management, Strategy Management, Balanced Scorecard.

1. INTRODUCTION

Controlling is a objective-oriented process of enterprise steering. It focuses on coordinating processes of planning, managing, executing control and delivering information. From the viewpoints of organizational structure and time horizon one can split controlling into:

- Strategic controlling that supports strategic management and tackles a system of plans and targets;
- Operational controlling which deals with problems of short-term planning and is often associated with budgeting and a responsibility centers concept.

Over many years IT solutions that aided operational controlling have profited from advantages of a spreadsheet which first of all was used as a tools for preparing and managing operational budgets. Nowadays companies more often apply solutions of Business Intelligence class, which are based on data warehousing and On-Line Analytical Processing (fig. 1).
For several years development of IT tools supporting strategic controlling, especially the concept of Balanced Scorecard (BSC) have been observed. An implementation of BSC concentrates on inventing a system of performance measurement in many perspectives (fig. 2). The system enables managers to integrate operational activities of all units and all employees with strategic goals of company.

BSC makes it possible to monitor the process of executing company’s strategy with the use of inter-linked strategic and operational performance indicators. In the BSC concept one assumes that improving performance in the objectives found in the Learning & Growth perspective enables the organization to improve its Internal Process perspective Objectives, which in turn enables the organization to create desirable results in the Customer and Financial perspectives.
IT tools that support BSC are the basic component of information systems, understood as Strategic Performance Management, Corporate Performance Management (CPM), Enterprise Performance Management or Business Performance Management (BPM) solution [Januszewski 2008, p.145, 2009, p.39-45]. A CPM suite consists of a set of integrated tools for both strategic and operational controlling.
2. THE CHARACTERISTIC OF CORPORATE PERFORMANCE MANAGEMENT SOLUTIONS

According to Gartner Group the main application components of a BPM suite are as follows [Chandler et al. 2009, p.5-7):

- Budgeting, Planning and Forecasting,
- Profitability Modeling and Optimization,
- Strategy Management,
- Financial Consolidation,
- Financial, Statutory and Management Reporting.

The general architecture of a BPM solution is presented on the scheme (fig. 3).

**Budgeting, Planning and Forecasting** (BP&F) support the creation of financially focused budgets and plans, and should support the complete budget creation and approval process with appropriate workflow that enables users to define and control the flow of budgets, plans and forecasts for review and approval. BP&F applications provide links to strategy maps in scorecard applications which are used for strategic initiatives and goals monitoring. This is due to internal integration of all CPM applications which are based on a common database (usually data warehouse).

**Profitability modeling and optimization** modules support implementation of detailed costing and profitability models and analysis, what is necessary for margins improvement. The applications often use activity-based costing (ABC) and activity-based management (ABM) as the methodology to develop sophisticated cost and profitability analytical models. The ABM applications provide various measures, which are used for performance evaluation in Balanced Scorecard.

**Financial Consolidation** applications let organizations reconcile, consolidate, summarize and aggregate financial data based on different accounting standards and federal regulations.

**Financial, Statutory and Management Reporting** module supports both reporting of statutory financial information and management reporting. Statutory financial reporting must meet specific disclosure rules, such as U.S. GAAP (generally accepted accounting principles) or International Financial Reporting Standards (IFRS). The management reports are used by executives at corporate and business unit levels to manage and explain financial performance. The reporting applications are strongly integrated with dashboards and scorecard applications, and use specific visualization techniques, such as decision trees, heat maps, and hyperbolic trees.
Strategy management applications are essential for managing business strategies. They provide a platform for definition and implementation of strategies at any level, from departments to business lines and throughout the company. The applications support strategic planning, modeling and monitoring to improve corporate performance, accelerate management decision making and facilitate collaboration.
They support formal strategy management philosophies (e.g. the Balanced Scorecard) but might be adapted to business specific needs. Strategy management comprises [Chandler et al. 2009, p.6]:

- Strategic planning application, which includes long-term financial planning, which creates a high-level perspective of revenue, expenses, balance sheet items and cash flows to show the financial impact of different strategic alternatives.
- Initiative/goal management application, which includes project-management-like tools to enable responsible managers to execute specific tasks related to a strategy.
- Scorecards and strategy maps, that help organizations measure and align the strategic and tactical aspects of businesses, processes and individuals with goals and targets. They are used to record strategies, objectives and tasks, measure performance, and provide a collaborative environment for effective, enterprise-wide communication. Strategy maps show a logical, step-by-step connection between strategic objectives in the form of a cause-and-effect chain.
- Dashboards (or cockpits), which are a reporting mechanism that aggregates and displays metrics and key performance indicators (KPIs), enabling them to be examined at a glance before further exploration via additional BI tools, make it possible for the users to quickly monitor and track performance via an aesthetic user interface, which employs visualization components, such as gauges, thermometers, dials and traffic lights (fig. 4).

3. SAS STRATEGIC PERFORMANCE MANAGEMENT AS AN EXAMPLE OF STRATEGY PERFORMANCE SOLUTIONS

The market of Corporate Business Performance solutions is still growing [Rayner et al.2007, p.4]. Consultant companies usually mention 4 to 6 main vendors of CPM solutions: Oracle (with Hyperion), SAP (with Business Objects), IBM (with Cognos), SAS Institute, Microsoft and Infor [Hamerman 2009, p.9], [Vesset & et al. 2007, p.4], [Chandler et al. 2009, p.2]. According to Gartner Group, SAS Institute has: “the strongest combination of profitability modeling and strategy management of any CPM suite vendors, which makes it suitable for organizations looking for a strategic approach to CPM” [Rayner et al.2007, p.19]. Also French analytical company Yphise confirms the leading position of SAS. It considers SAS Strategy Performance Management the best strategy management package, with better functionality than Cognos Metric Studio (IBM), Microsoft Office PerformancePoint Server (Microsoft) and SAP Strategy Management (SAP)

1 Dashboards are a common part of all CPM applications.
The same opinion shares Simon who considers that [Simon 2007, p.4]: „Competent strategy solutions are still few and far between, but SAS Strategic Performance Management represents a notable exception, providing support for the whole strategy process”.

Figure 4. Dashboard – an example.

SAS Strategic Performance Management helps align, monitor and measure the execution of key initiatives that serve strategies and performance goals. It is the part of SAS Financial Intelligence platform. SAS Financial Intelligence includes three solutions: Strategic Performance Management (strategy management, improvement, monitoring), Financial Management (planning and financial reporting) and Activity-Based Management. All the solutions are independent but integrated and complementary. The Strategic Performance Management (SPM) solution relies on the Enterprise Intelligence Platform. It includes three levels and tools: monitoring (Information Delivery Portal), modeling (SPM) and population (Data Integration Server) [Yphise 2009, p.12].
SAS Strategy Performance Management is a Web-based application for designing, building and managing scorecards, dashboards and diagrams, including strategy maps. The most important components of SAS SPM solution are as follows [Simon 2007, p.2-4):

- strategy maps,
- performance dashboards,
- scorecards,
- performance portal.

Strategy maps are highly visual tools that allow relationships between objectives, projects and KPIs to be illustrated and supports collaborative working as management strive to confirm the integrity of the inter-relationships between different measures in different areas. The components of the strategy map are completely user definable and can therefore support any reasonable strategy description regardless of the methodology employed. Users have virtually unlimited flexibility to describe objectives and projects and performance measures using terms that fit the organization’s culture and language.

In contrary to strategy maps, which provide a window on performance suitable for analysts, strategy setters and senior managers, performance dashboards have become accepted as a much more suitable way of communicating strategy ‘down the line’. Like the strategy map, performance dashboards use traffic lighting to denote good (green), amber (acceptable) and red (bad) performance.

Scorecards are the more ‘traditional’ way of depicting performance, with performance set out in grid style and icons, such as colored arrows, or other user selected icons to denote the status of performance. Whereas the emphasis of a dashboard is to instantly communicate performance for a limited number of measures in an eye catching way, the scorecard is designed to promote deeper enquiry and analysis. The exact layout of the scorecard is totally user definable. It is possible to look at different performance measures, in different time periods and for different slices of the organizational hierarchy and other dimensions. The drill-down capability of the SAS scorecards allows the user to follow KPIs down the hierarchical organizational structure, revealing the performance contribution made by business units lower down the tree. The system provides visual cues, flagging where acceptable performance at one level may be masking poor performance lower down the structure. During drill-down the system shows the interrelationship between the performance measures in a tree-like structure. The user can see the associations between KPIs and other measures such as projects, goals and objectives and start to form a view about how performance in one area is affecting performance in another.

The SAS Portal is the place where different views on performance can be brought together (fig. 5).
The portal can present a highly personalized view of performance that allows the users to focus on the measures that are most relevant to their role. As other above described tools the portal is totally user definable and if desired, can combine dashboards, scorecards and strategy maps depending on the user’s skills and role within the organization.

SAS SPM includes several out-of-box templates with predefined contents and indicators for operational directions (IT scorecard, HR scorecard, marketing, finance) and sectors (banking, insurance, telco, retail, hospital, health care).
4. CONCLUSION

Despite the fact that a lot of companies have introduced transactional ERP systems and Business Intelligence tools for operational controlling, only few of them have a complex solution which integrates strategic and operational management. And for the purpose of integration itself CPM suites are suggested. CPM packages are IT complex solutions that integrate strategic and operational controlling (management). Embedding CPM packages in technologies that integrate sets of information (data warehouses, OLAP tools) and communication and reporting platforms (corporate portals), along with implementing methods combining strategic and operational management (Balanced Scorecard) make it strongly possible to execute the fundamental function of controlling – parallel and horizontal coordination of company’s operations.

The CPM market is rapidly growing. Analysts expect CPM software revenues to grow by 12.7% to 2012, growing from its current size of $2 billion to $3.2 billion during this timeframe [Hamerman 2009, p.1]. This forecast makes us consider that CPM solutions are becoming more and more important for many managers and companies.

REFERENCES

LOGISTICS IN INFORMATION SYSTEMS

Krzysztof Karpio
Department of Informatics
Warsaw University of Life Sciences (SGGW)

Arkadiusz Orłowski
Department of Informatics
Warsaw University of Life Sciences (SGGW)
&
Institute of Physics
Polish Academy of Science

Mateusz Przybyć
Department of Informatics
Warsaw University of Life Sciences (SGGW)

Summary: The future competitive advantage of a company will come from responding to customers needs at the end of the supply chain in a better way than competitors. Logistics plays a key role in this process. The purpose of this paper is to explain why logistics concepts are so important and to provide the description of ERP systems that provide logistics solutions to a company.

Key words: Logistics, ERP Systems.

1. INTRODUCTION

Nowadays world economy is in a radical shift. We are moving away from a world where national economies are isolated from each other by barriers that hinder cross-border trading and investments. Such barriers as: distance, time zones, languages, national differences in government regulation, culture and business systems do not cause major problems in today’s trading market. The globalization of markets is merging national markets into one huge global marketplace, where companies have great opportunities to earn money. However, the emergence of globalization has made the business world tougher. Competition between companies is fiercer than ever and staying profitable is not as obvious as in earlier days. The main drivers of globalization are the changes in customer preferences, technological improvements and the decline in barriers to the free flow of goods, services and capital [7, pp. 5-7].

Customer preferences have changed in many ways. The increasing variety of available services and products on the market imply that customers can be more demanding in their selection. Customers nowadays expect goods to be available at
all times, in the right quality and to the right price. As late as the early 1990’s the average time that a company needed to process and deliver goods to a customer from warehouse inventory ranged from 15 to 30 days [7 p.232]. Today, however, as customers are more demanding, firms must be able to manufacture to exact specifications and rapidly deliver to customers anywhere around the globe. Customer order and delivery of goods can be made in hours. Service failures that often occurred in the past are no longer acceptable. Instead, perfect orders i.e. delivering the correct assortment and quantity of products to the right location on time, damage free with correct invoice is what customers expect [2 p.3].

In order to cope with these challenges in the new business environment it is essential that companies have, among many other things, a well-functioning material management, i.e. an efficient physical supply of materials throughout the supply chain. An efficient materials flow is crucial in order for firms to meet customer expectations and thereby deliver the products at the right time, in the right quality and quantity, at the right price and at the right place.

As globalization of markets and of production is constantly growing, more and more goods will be traded across the world, thus more goods need to be stored. This fact further strengthens the importance of conducting studies in the areas presented in this paper.

Logistics and Supply Chain Management have during the last centuries grown in popularity because of the firms’ needs to become more customer-oriented and cost effective. One of the factor that influence firm costs is holding inventory balanced between supply and demand. By holding inventory, cost reductions in other business areas can be made. For example, in purchasing where ordering large quantities will lead to lower per unit price but also increased inventory. This is a very delicate balance that companies must consider, as holding inventory is very expensive.

To achieve a well functioning materials flow companies must put much emphasis on the different warehouse operations and inventory management, including the stratification of articles, product flow patterns and the layout of the storage. Today control and communication in materials flow demands computerized systems. Information systems refer not only to computerized solutions but also to all communication and data processing connected to the materials flow. Thereby, the overall control of the entire materials flow can be viewed as one process, in order to see how the different parts are integrated and influence each other.

2. LOGISTICS

Logistics has gone from being a military matter to being one of the key business issues today. The underlying factor to this situation is, in fact, that logistics is a cross-functional subject, cutting across functional boundaries. Logistics is the
work required to move and position inventory throughout a chain of participants. From initial purchase of material or component, the system of logistics adds value by moving inventory when it is needed and where it is needed. Materials and components gain value at each step of their change into finished inventory [2 p.44]. A basic definition of logistics is as follows:

“Logistics is the process of strategically managing the procurement, movement and storage of materials, parts and finished inventory (and related information flows) through the organization and its marketing channels in such a way that current and future profitability are maximized through the cost-effective fulfillment of orders.” [5 p.4

There are a number of theories and methods that are described in the business literature about how to succeed today and stay competitive in the future. Companies must recognize that the competition today is through their capabilities and competencies. By managing their core processes better than competitors manage theirs, organizations can create superior value for customers and consumers. The core processes include such activities as new product development, supplier development, order fulfillment, and customer management. If an organization can perform these activities in a more cost-effective way than the competitors, the organization will gain the advantage at the marketplace [5 p.28].

With the use of logistics management, the goal is to link the marketplace and the operating activity business in such a way that customers are served at higher levels and at a lower cost. A broader definition of logistics management that is widely used comes from The Council of Logistics Management (CLM) and is as follows:

“Logistics management is that part of the supply chain process that plans, implements and controls the efficient, effective flow and storage of goods, services, and related information from the point-of-origin to the point-of-consumption in order to meet customers’ requirements.” [13 p.3]

In this definition, compared to the first definition, flows of goods, services, and information are included in the sectors of manufacturing and service. During the last decades important aspects such as quality and information have arisen within this field, since these aspects have a direct influence on companies’ flows efficiency.

With the use of a total system viewpoint of logistics management, the needs of customers could be satisfied through the coordination of the materials and information flows that extend from the marketplace, through the operation environment of the firm and to the suppliers. Materials can be raw materials, components, parts, tools, consumables, services or any other types of item. The material flow represents the supply of product through the network in response to demand from the succeeding organization. Often it is difficult to see where the flow starts in the
chain and where it ends. The negative effect of this is the build-ups of inventory and slow response to demands of the end customer [6 p.11].

Logistics is a cross-functional subject cutting across functional boundaries of the organization in focus into the supply chain. This implies the complexities of synchronizing the movement of materials and information between business processes. The system’s nature of logistics has proved a particularly difficult lesson to learn, and individual organizations still often think that they can optimize profit conditions for themselves by exploiting others in the supply chain. The emergence of logistics has therefore been dependent on the development of a cross functional model of the organization and there has to be an understanding of the need to integrate business processes across the supply chain, both internally and externally. The future competitive advantage will come from responding to customers at the end of the supply chain better than competitors do, and in this response, logistics play a key role [6 p.11].

FLOWS IN LOGISTICS
This part of the work provides a short description of logistics flows that exist within the supply chain. These flows are:

- monetary that is concerned with invoicing and payment;
- information that is the initiator for both monetary flow and material flow;
- material that represents the goods to be moved;
- recourse that represents the resources that are needed to move the material within the chain.

**Material flow**

The material flow represents the supply of product through the network in response to demand from the next organization. The issue here is how long does it take to get the product through the various stages from one end of the chain to the other. The focusing on time is important here because it measures how quickly a given network can respond to demand from the end customer. Often it is difficult to see where the flow starts in the chain and where it ends. The negative effect of this is the build-up of inventory and the slow response to the demand of end customer [6 p.11]. Therefore there are different strategies for managing inventory and the material administration.

**Information flow**

Information is a collection of facts that is organized in such way that they have additional value beyond the value of the facts themselves. Data on the other hand consist of raw facts, such as an employee’s name and number of hours worked in a week, inventory part numbers or sales order. The value of information is directly linked to how it helps decision makers achieve their organization’s goals [12 p.7].
Today the information flow within the logistics has become vital as, this flow enables chains to respond in real time with accurate data. Companies today look at information flow as an asset, since it is not possible to have efficient and reliable material flow without it. The ability to change data into useful information is essential because otherwise it will only be data. The material flow cannot be isolated from information flow. There are strong links between the physical flow and the information that flows both upstream and downstream. To manage and communicate a material flow today, IT is necessary. But an information system is not only IT solutions but also other communication and data processing that are linked to material flow.

Figure 1. The logistics information flow.

Source: Own preparation based on [13 p.176].
Information flow can be defined as the stream of data in different directions with variable contents between various databases (departments) within a company. Data for a logistics management information system can come from many sources and the most important sources of data is common database as presented at the figure 1.

SUPPLY CHAIN
A supply chain consists of all parties involved, directly or indirectly, in fulfilling a customer request. The supply chain not only includes the manufacturer and suppliers, but also transporters, warehouses, retailers, and customers themselves. Within each organization, such as manufacturer, the supply chain includes all functions involved in receiving and filling a customer request. These functions include, but are not limited to, new product development, marketing, operations, distribution, finance, and customer service [4].

Let’s consider a customer walking into a ABC store to purchase a product. The supply chain begins with the customer and their need for detergent. The next stage of this supply chain is the ABC retail store that the customer visits. ABC stocks its shelves using inventory that may have been supplied from a finished-goods warehouse that this store manages or from a distributor using trucks supplied by a third party. The distributor in turn is stocked by the manufacturer that receives raw material from a variety of suppliers.

A supply chain is dynamic and involves the constant flow of information, product, and funds between different stages. In our example, ABC store provides the product, as well as pricing and availability information to the customer. The customer transfers funds to ABC store. ABC conveys point-of-sales data as well as replenishment order via trucks back to the store. ABC transfers funds to the distributor after the replenishment. The distributor also provides pricing information and sends delivery schedules to ABC. Similar information, material, and fund flows take place across the entire supply chain.

This example illustrates that the customer is an integral part of the supply chain and that the primary purpose from the existence of any supply chain is to satisfy customer needs. As it was described in previous example, supply chain activities begin with a customer order and end when a satisfied customer has paid for his or her purchase. Presented example involves only one player at each stage, but in reality, a manufacturer may receive material from several suppliers and then supply several distributors.

The objective of every supply chain is to maximize the overall value generated. The value a supply chain generates is the difference between what the final product is worth to the customer and the effort the supply chain expends in filling the customer’s request. For example, a customer purchasing a product from ABC store pays $1,000, which represents the revenue the supply chain receives. ABC and other stages of the supply chain incur costs to convey information, produce com-
ponents, store them, transport them, transfer funds, and so on. The difference between the $1,000 that the customer paid and the sum of all costs incurred by the supply chain to produce and distribute this product represents the supply chain profitability. Supply chain profitability is the total profit to be shared across all supply chain stages. The higher the supply chain profitability, the more successful the supply chain. Supply chain success should be measured in terms of supply chain profitability and not in terms of the profits at an individual stage [4].

INVENTORY MANAGEMENT
The Institute of Logistics and Transport has the following definition of inventory management:
“The effective management of stock, materials, parts and finished products, including additions and deletions (i.e. control of movements in and out). Essential for determining capital investment returns and viability of stock levels and for the avoidance of opportunity cost (i.e. money tied up in stock that could be better used).” [10]

In theory, the best solution for a company in servicing its customers would be to locate an inventory in each facility that is closest to the customer. There are not many companies that could afford such a luxurious inventory commitment because the costs are discouraging. Therefore, according to the definition, the objective of the effective inventory management is to accomplish desired customer service with the minimal inventory level followed by lowest tied up capital in inventory.

*Figure 2. The Japanese Sea: Inventory hides the problems.*

The issue of coping with uncertainty is important as it exists in a realistic operating environment. Uncertainty exists in the production system in many different ways.
The demand for the product and replacement parts is typically forecast, and as a result has a certain degree of uncertainty with respect to both the quantity demanded in a particular time period, and the timing of the quantity demanded. Hence, the existence of an inventory can be considered based on the fact that the logistics process is not stable enough. Inventories function can be illustrated with help of Japanese manufacturing philosophy, “The Japanese Sea”, which was developed from the concept of removing all inventories in order to identify problems in the company’s logistics processes. In other words, lowering the inventory levels discloses problems that were previously hidden [5].

The inventories are still used among the companies and, of course, the companies do not have to lower or eliminate its inventories. It is almost impossible as well as dangerous for a company to make an effort to solve all previously hidden problems immediately. Obviously, the cautious step-by-step process of lowering the inventory levels in combination with the total logistics cost analysis seems to be the best solution.

It is not a mistake to use inventories. However, the utilization of inventory demands effective inventory management with defined inventory policy, order quantities, and inventory service levels. The objective of inventory management is to improve the corporate profitability and increase customer service level with minimal tied up capital in inventory. However, the inventory management strategy has to be seen in the greater perspective of the total logistics cost concept with possible negative consequences for other logistics costs. Furthermore, in the longer perspective, a company has to try to remove insecurities in the supply chain by establishing collaboration with strategic suppliers.

The main purpose of inventories is to act as a buffer between supply and demand. They allow operations to continue smoothly and avoid disruptions. There are of course other reasons for holding inventories [16]:

- to act as a buffer between different operations;
- to allow for demands that are larger than expected, or at unexpected times;
- to allow for deliveries that are delayed or are too small;
- to take advantage of price discounts on large orders;
- to buy items when price is low and expected to rise;
- to buy items that are going out of production or are difficult to find;
- to make full loads and reduce transport costs;
- to give cover for emergencies.

Problems of inventory management have been around for a very long time and people are still developing new ideas. A huge amount of work has been done to develop the principles of effective inventory control, but it is still difficult to identify the best policies. There is no ideal way of organizing stocks, and the best op-
tions depend on the type of operations, constraints, objectives, and a whole range of subjective factors. In spite of this, there is a clear trend towards lower stocks. New methods allow organizations to move goods quickly through the supply chain, working with far less stock to achieve the same levels of customer service [3 p.195].

Inventory management is a major investment, which managers want to organize as efficiently as possible. Unfortunately, this is more difficult than it seems. The amount of stock varies enormously, but a typical organization holds about 20% of its annual turnover. To be able measure inventory management performance, the inventory turnover rate and inventory service level were applied. The turnover rate of the storage is defined as the number of times per year storage is turned over. There are different ways of calculating inventory turnover rate that depend on institutional to company dependent relations. One of the ideas covers tied up capital and can be calculated as the following:

\[
\text{Inventory}_\text{turnover} = \frac{\text{Turnover}}{\text{Tied}_\text{up}_\text{capital}}
\]

Today, it is generally accepted that the need to understand and meet customer requirements is a condition for survival. For a company to meet customers’ requirements it is not possible for functions within a company to act independently of each other. The cost of purchased materials and supplies is a significant part of a total costs in most organizations, but there is also a major opportunity for leveraging the capabilities and competencies of suppliers through closer integration of the buyers’ and suppliers’ logistics processes.

WAYS OF MAPPING PROCESSES

Process mapping is the visualizing of the activities and objects of a process, and how their relationships are carried out. To map a process is to create a model of a process.

Business process is defined as a coordinated sequence of activities that has its purpose to transform some kind of input to output. The input could be in form of information, material or payments. The most typical characteristic for business processes, especially core processes that add value to the products and services, is that they reach across functional boundaries. Process perspective means organizing resources and responsibilities around core business processes instead of around tasks and functions [9].

Processes often include customers and suppliers, although many companies have not reached the level to include external participants, but this does not mean that a process should not be mapped. To make supply chains more effective, there is a need to improve the conditions regarding the exchanges and flows between com-
panies. Due to the conditions that companies today meet on the market, a process oriented view has influence in the way they organize and conduct their business. Mapping a supply chain is the actual process that requires focus, not result. The key is to track one order, one product, or one person through the process with the respect to time. A map is a mirror of what takes place during a given time period and during this time period the actual time that it is observed is recorded. By mapping the process, key operations are still visible, but the subprocesses that often consume time and generate the greatest inefficiencies are revealed at the same time. This causes solutions to problems to be generated and thus the supply chain is improved [6 p.122].

Create task weakness: Before the mapping process, the supply chain processes that cross all functions of the organization need to be identified. This stresses the importance of the key functions to be represented.

Select the process to map: To make the mapping process feasible, identify the core processes within the organization and the time they take before deciding on the priorities for detailed mapping. When selecting the process, make sure that there is a generic customer or group of customers that the process serves.

Collect data: The most effective way to collect data is simply to follow the item through the process, also referred to as walking the process. An actual component or order will be followed through all stages that are included in the process. It is important to identify those individuals who are actively involved in the process and really knows what is happening. Each movement of the item should be described with respect to time [6 p.123].

Distinguish between value-adding and non-value-adding time: Value-adding time is time when something takes place on the item that the end customer is willing to pay for. It is important that the definition of value-adding takes place within the organization and the definition should be associated with the overall business strategy. When there is an understanding of the value-adding criteria at the strategic level, these criteria can be translated into value-adding criteria at an operational level. There are three criteria that are characterized for value-adding time:

• if the process or elements physically change the nature of consumable item;
• if a change to the consumable item produces something that the customer values or cares about and also is willing to pay for;
• if the process is right the first time, and will not have to be prepared in order to produce the desired result, that is valued by the customer.

The non-value adding activity can be split into four categories: delay, transport, storage and inspection.

Construct the time-based process map: The overall purpose of the time-based process map is to represent the data that is collected clearly and concisely. By doing this the critical aspects of the supply network can be communicated in an easily accessible way. If the process can be represented on a single piece of paper the
members of the project can easily see the issue. To extract the relevant data, it is useful to sketch a flow diagram so that the linkages and dependencies between steps can be clarified before constructing the map. The flow diagram can be used to estimate the total time that the business process consumes. Sample process map is presented on the following figure:

![Figure 3. Example of process map.](image)

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
<th>Symbol</th>
<th>Time</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Machine complete</td>
<td>O</td>
<td>1:37</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Inspect</td>
<td>□</td>
<td>0:45</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Wait transport</td>
<td>D</td>
<td>5:53</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Transport to heat treat</td>
<td>→</td>
<td>0:08</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Wait heat treat</td>
<td>D</td>
<td>3:34</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Heat treat</td>
<td>O</td>
<td>4:15</td>
<td></td>
</tr>
</tbody>
</table>

→ - transport  ▼ - store  O - operation  □ - inspect  D - delay

Source: [6 p.124].

Solution generation: When the time-based process map has been produced, the opportunities for improvement are generally obvious. The next step is to collect ideas and categorize causes of non-value-adding activity by using problem-solving approaches.

3. ENTERPRISE RESOURCE PLANNING

“The enterprise resource planning (ERP) system is an integrated set of programs that provides support for core organizational activities such as manufacturing and logistics, finance and accounting, sales and marketing and human resources. An ERP system helps the different parts of the organization share data and knowledge, reduce costs, and improve management of business processes” [14].

Enterprise Resource Planning system is a tightly integrated software package that consists of modules that helps the company to automate and integrate its business processes and provides an easily accessible environment to share common data. ERP system provides many benefits to companies and gives a chance to enable organizational standardization. They integrate departments like sales, logistics, production etc. in such a way that data produced in a department becomes immediately available to other departments. It enables a company to have a single image to its customers and vendors. Documents and data produced in different departments, plants and locations can have a common structure and view. ERP systems come
with a number of best practices that can help companies in improving the way of doing business.

THE EVOLUTION OF ERP

ERP began life in the 1960s as Material Requirements Planning (MRP), an outgrowth of early efforts in bill of material processing. MRP’s inventors were looking for a better method of ordering material and components, and they found it in this technique. The logic of material requirements planning is pretty simple and is based on asking the following questions [15 p.6]:

- What are we going to make?
- What does it take to make it?
- What do we have?
- What do we have to get?

This is called the universal manufacturing equation. Its logic applies to every type of things and wherever they are being produced. Material Requirements Planning simulates the universal manufacturing equation. It uses the master schedule (What are we going to make?), the bill of material (What does it take to make it?), and inventory records (What do we have?) to determine future requirements (What do we have to get?). The main objectives of MRP system are:

- To ensure that materials and products are available for production and delivery to customers,
- To maintain the lowest possible level of inventory,
- To plan manufacturing activities, delivery schedules and purchasing activities.

MRP systems are not common nowadays as they have a lot of limitations. Those limitations describes H. Klaus in one of his lectures:

“MRP were the first off-the-shelf business application to support the creation and maintenance of material master data and bill-of-materials (demand-based planning) across all products and ports in one or more plants. These early packages were able to process mass data but only with limited processing depth”[8 p.141].

Visual depiction of this and the subsequent evolutionary steps presents Figure 4. MRP quickly evolved, however, into something more than merely a better way to order. Early users soon found that Material Requirements Planning contained capabilities far greater than merely giving better signals for reordering. They learned this technique could help to keep order due dates valid after the orders had been released to production or to suppliers. MRP could detect when the due date of an order (when it’s scheduled to arrive) was out of phase with its need date (when it’s required). This was a breakthrough. For the first time ever in manufacturing, there was a formal mechanism for keeping priorities valid in a constantly changing environment. The function of keeping order due dates valid and synchronized with these changes is known as priority planning.
The issue of priority was not the only problem—equally challenging problem was capacity. Techniques for helping plan capacity requirements were tied in with Material Requirements Planning. Further, tools were developed to support the planning of aggregate sales and production levels (Sales & Operations Planning); the development of the specific build schedule (master scheduling); forecasting, sales planning, and customer-order promising (demand management); and high-level resource analysis (Rough-Cut Capacity Planning). Systems to aid in executing the plan were tied in: various plant scheduling techniques for the inside factory and supplier scheduling for the outside factory—the suppliers. These developments resulted in the second step in this evolution: closed-loop MRP [15 p.8] (see Figure 5).

The most important benefit of closed-loop MRP is that it contains tools to address both priority and capacity and that it supports both planning and execution. Moreover, it has provisions for feedback from the execution functions back to the planning functions. Plans can then be altered when necessary, thereby keeping priorities valid as conditions change.
The next step in this evolution is called Manufacturing Resource Planning or MRP II (to distinguish it from Material Requirements Planning, MRP). A direct outgrowth and extension of closed-loop MRP, it involves three additional elements:

- Sales & Operations Planning – a powerful process to balance demand and supply at the volume level, thereby providing top management with far greater control over operational aspects of the business,
- financial interface – the ability to translate the operating plan (in pieces, pounds, gallons, or other units) into financial terms (dollars).
- Simulation – the ability to ask “what-if” questions and to obtain actionable answers – in both units and dollars.

Definition of Manufacturing Resource Planning come from APICS – The Educational Society for Resource Management:

“MANUFACTURING RESOURCE PLANNING (MRP II) – A method for the effective planning of all resources of a manufacturing company. Ideally, it addresses operational planning in units, financial planning in dollars, and has a simulation capability to answer “what-if” questions. It is made up of a variety of functions, each linked together: business planning, sales and operations planning, production planning, master scheduling, material requirements planning, capacity require-
ments planning, and the execution support systems for capacity and material. Output from these systems is integrated with financial reports such as the business plan, purchase commitment report, shipping budget, and inventory projections in dollars. Manufacturing resource planning is a direct outgrowth and extension of closed-loop MRP” [1].

The latest step in this evolution is Enterprise Resource Planning (ERP). The fundamentals of ERP are the same as with MRP II. However, thanks in large measure to enterprise software, ERP as a set of business processes is broader in scope, and more effective in dealing with multiple business units. Financial integration is even stronger. Supply chain tools, supporting business across company boundaries, are more robust.

Figure 6. Enterprise Resource Planning

Source: [15 p.11]
Enterprise resource planning (ERP) predicts and balances demand and supply. It is an enterprise-wide set of forecasting, planning, and scheduling tools, which:
- links customers and suppliers into a complete supply chain,
- employs proven processes for decision-making, and
- coordinates sales, marketing, operations, logistics, purchasing, finance, product development, and human resources.

Its goals include high levels of customer service, productivity, cost reduction, and inventory turnover, and it provides the foundation for effective supply chain management and e-commerce. It does this by developing plans and schedules so that the right resources—manpower, materials, machinery, and money—are available in the right amount when needed. Figure 6 shows a graphical view of ERP.

ADVANTAGES AND DISADVANTAGES OF ERP SYSTEMS

The main reason for using ERP systems is to make possible online information sharing and flow efficiently throughout all departments, plants and other units of a company. The practical benefits are divided into five aspects [11]:

1. Operational – can offer benefits in terms of cost reduction, cycle term reduction, productivity improvement, quality improvement, and improved customer service.
2. Managerial – can help an organization achieve better resource management, improved decision making and planning, and performance improvement.
3. Strategic – can assist in business growth, alliance, innovation, cost, differentiation, and external linkages.
4. IT infrastructure – supports business flexibility, reduced IT cost and marginal cost of business units’ IT, and increased capability for quick implementation of new applications.
5. Organizational – supports organization structure change, facilitating employee learning, empowering workers, and building common visions.

One of the main advantages of ERP is that it provides an integrated environment and frees the company from using software “islands” that do not effectively interface with each other. Without ERP, a company must use isolated software for handling customer orders, one for managing product engineering and bill of materials, one for inventory tracking, one for finance management etc. Another advantage is that an event is entered only once and is available to all departments by related modules and transactions immediately. It eliminates the data redundancy problem. Management of data can be done more easily.

Additionally, ERP provides a chance for a company to renew and develop its business processes. From customer interaction point of view, ERP includes the concepts of front office (how the company interacts with customers), which include Customer Relationship Management (CRM), back end (internal workings of the
company to fulfill customer needs), which includes quality control, to make sure there are no problems not fixed, in the end products, supply chain management (interacting with suppliers and transportation infrastructure), all of which are generally integrated in an ERP. Without an ERP that integrates this stuff, it can be a nightmare for a manufacturer to manage [17].

However, with all the advantages that ERP offers, there are a number of disadvantages as well. Perhaps one of the biggest disadvantages to this technology is the cost. At this time, only large corporations can truly take advantage of the benefits that are offered by this technology. A number of studies have shown that the biggest challenges companies will face when trying to implement ERP deals with investment. In addition to initial costs, support costs can also be very high. The employees must be continually trained on how to use it, and it is also important for companies to make sure the integrity of the data is protected.

One of the biggest problems with ERP is that it is hard to customize. Very few companies can effectively use ERP right out of the box. It must be modified to suit their needs, and this process can be both expensive and tedious. Even when a company does begin changing the system, they are limited in what they can do. Most ERP vendors will not allow the structure of the software to be altered. One advantage to ERP is that making the necessary changes to use it may actually make a company less competitive in the market. In addition to the costs involved with implemented ERP and training workers to use it, the ERP vendors may charge additional license fees, putting a strain on companies that do not have enough resources to pay for them.

5. FINAL REMARKS

The world economy is, and has been for some time, in a radical shift and that we are moving away from a world where national economies are isolated from each other. The globalization of markets is merging national markets into one huge global marketplace, where companies have large opportunities to earn money. Competition between companies is fiercer than ever and there is great pressure from various stakeholders to be as profitable as possible. In order to cope with these challenges it is essential that companies have, among many other things, a well functioning material management, i.e. an efficient physical supply of materials throughout the supply chain. An efficient materials flow is crucial in order for firms to achieve high customer service levels and meet customer expectations thus deliver the products at the right time, in the right quality and quantity, at the right price and at the right place.

The purpose of the paper was to show and emphasis how important computerized logistics systems are in nowadays economy. This work presented the impor-
tant facilities of such integrated systems as well as their major advantages and dis-
advantages.

REFERENCES

source Management, Falls Church, VA
2. Bowersox, D., Closs, D. and Cooper, M., Supply Chain Logistics Management, Inter-
Management, Elsevier Science Ltd, 2001
Times Prentice Hall, 1998
6. Harrison, A. and van Hoek, R., Logistics Management and Strategy, Pearson Educa-
tion Limited, 2002
7. Hill, C., International Business–Competing in the Global marketplace, 3rd edition, 
8. Helmut Klaus, Michael Rosemann and Guy G. Gable, Information Systems Frontiers, 
2000, Information Systems Management Research Center, Queensland University of 
Technology, Brisbane, Australia
10. Lowe, D., The Dictionary of Transport and Logistics, The Institute of Logistics and 
Transport, Kogan Page, London, 2002
Course Technology, 2001
14. J. Stratman, A. Roth, Enterprise resource planning (ERP) competence constructs: two-
stage multi-item scale development and validation, Decision Sciences 33 (4) (2002) 
601.
15. Thomas F. Wallace, Michael H. Kremzar, Mike Kremzar “ERP: Making It Happen” 
Publishing Company, 1996
17. Website: http://library.ieu.edu.tr
DATA WAREHOUSE DESIGN

Włodzimierz Kwasowiec  
Department of Informatics  
Warsaw University of Life Sciences (SGGW) &  
Faculty of Mathematics and Natural Sciences  
Cardinal Stefan Wyszyński University

Waldemar Karwowski  
Department of Informatics  
Warsaw University of Life Sciences (SGGW)

Summary: The present paper shows the designing process in a data warehouse creation. It starts from a general description of data warehouses. It discusses in detail business and logical models developed in this process. Finally the most important tools provided by software vendors are mentioned.

Key words: Data Warehouse, Data Marts, Business Intelligence.

1. INTRODUCTION

Traditional online transaction processing systems (OLTP) are not most efficient tools in decision-making processes, especially if we need to have an efficient access to various information. If we want to get better strategic information and to aid better corporate manager decisions (see Person, Smith, Thearling [9] and Shaw [10]), we may use data warehouses (DW). DW will integrate data of various sources and allow online analytic processing (OLAP). The basic aim of data warehouse is to deliver proper information in a reasonable time and cost to enable obtaining required information and to be able to make best decisions (see Karwowski, Kwasowiec, Orłowski [8]).

According to W.H. Inmon [6], data warehouse is a centralized repository of integrated, and non-volatile, and directed databases designed as a basis for decision-support systems (DSS), where each information (either atomic or processed data) is connected with a given time. Data warehouse is a subject-oriented integrated database where multi-dimensional analytical data are usually stored for a long time.

2. DATA WAREHOUSE STRUCTURE

In data warehouse structure we may distinguish three layers which are shown on the Figure 1 (see Jarke, Lenzerini, Vassiliou, Vassiliadis [7]), and we shall pre-
sent them chronologically, i.e. bottom-up. In the lowest level of the Figure 1 there exist operational databases that were designed and created for various applications. These databases are usually internal in an enterprise but the data may be also imported from outside or even from usual text files. Various data schemes may be used for storing this information. Data of a consecutive layer are obtained from the data of the lower layer.

**Figure 1. Architecture of a data warehouse.**

Proper parts of data warehouse are situated in the middle and upper layers of the Figure 1. Basic data warehouse, called global or corporational, is in the middle layer and contains results of processing of data coming from the lowest layer. Together with an information its history is stored here. After each update a data warehouse is completed by new properly processed information coming from an operational database of the lowest layer. Indeed, creation of a data warehouse might remind an adding of consecutive materialized relational views over some databases.
In the upper layer local data warehouses intended for special purposes may exist. They may contain highly aggregated information, e.g. some sums of global data. Data gathered in data warehouses might have been initiated either several minutes or few hours or many years ago. They may be used in various analyses (either integrated historic or concerning future planning and so on), in current enterprise management. For example, a data warehouse may contain data of all persons engaged in a given enterprise and data of all products and of all its parties.

However sometimes we may need more efficient processing. We may then design local data warehouses (called departmental data warehouses or data marts). Data marts are small repositories connected with sections of global data warehouses, containing data of some departments. Data marts might be implemented as relational databases or special multidimensional structures.

Data mart, for example concerning the marketing department, should be restricted to information on clients and transactions and products that are being sold (or bought) by this enterprise. Data marts are more efficient because they are small. Usually, data marts are implemented as relational databases or as special multidimensional structures.

There are two important differences between global data warehouses and data marts. First, global data warehouses are results of a composed process of extraction, integration, and transformation of all data but data marts result from an integration and transformation of some data (a section) taken from a global data warehouse. Second, data warehouses contain particular, large-scale and a little aggregated data. On the contrary, data marts contain strongly aggregated and small data.

Often we can distinguish an operational data store (ODS) which is situated between source data layer and the global data warehouse (see the Figure 1). ODS contains current and integrated and particular subject data and results of some transformations, integrations and aggregations of data coming from various sources. Granularity (a degree of aggregation) of data in ODS should be small and data might be summed partially. ODS may be considered as a set of updated materialized relational views. ODS should be very useful for decision support and for making various quick analysis needed in an enterprise management.

Very important component of a data warehouse is metadata repository, which contains information on the structure and placement of source data, and of data warehouses, and information on transformations realized when accessing data, and on user’s authorizations and privileges.

3. DATA WAREHOUSE GENERAL DESIGN

If we are going to design a data warehouse for a given enterprise, first we should find out and conform information coming from various sources and try to union them together in a model of the data warehouse for this enterprise (see
Gorawski [3], [4], [5]). Next we should design a logical structure of relations supporting the data warehouse as a non-volatile source of data used for processing queries. Moreover we should decide how to represent: tables, processes, indexes, and logical scheme of the data warehouse and how to ensure efficiency of the whole system. We should also choose hardware and software for the given data warehouse. Database schemes composing data warehouse may change because of the requirement changing possibility and therefore designing is a dynamic process in the life cycle of the system.

There are two main approaches to storing data in a data warehouse: the dimensional and the normalized approach.

In the normalized approach data in the data warehouse are stored following database normalization rules. Tables are grouped together by subject areas that correspond to some data categories (e.g., data on customers, products, finance, etc.). Then we can get some difficulties with taking data from different sources and with joining them into meaningful information and also with accessing information without a precise understanding of the sources of data and of the data structure in the data warehouse (because of the big number of tables involved). However this approach enables convenient adding information into the database.

In the dimensional approach, transaction data are partitioned either into facts, which are usually numeric transaction data, or into dimensions, which are the reference information that gives context to the facts. An example of facts is a sales transaction which can be broken up into such facts as the number of ordered products and the price of these products. By dimension we mean a group of logically connected attributes, e.g. day, month and year in the dimension date. As examples of dimensions, we can distinguish customer name, product number, order date, order ship-to and bill-to locations, and name and address of salesperson responsible for receiving the order. A key advantage of a dimensional approach is that the data warehouse is easier to understand and to use for users. The integrity of facts and dimensions are complicated and modelled enterprise should be stable because of difficulty to modify the data warehouse structure. However the retrieval of data from the data warehouse tends to operate very quickly. These approaches can be mixed and dimensional approaches can involve normalizing data to a degree.

Another important question in designing a data warehouse is which data to conform and how to do this. For example, one operational system feeding data into the data warehouse may use "M" and "F" to denote sex of an employee while another operational system may use "Male" and "Female". During implementing a data warehouse we should make consistent similar meaning data when they are stored in the data warehouse. Usually extract, transform, load tools are used in such situations.

There are two main data warehouse architectures: top-down (centralized) and bottom-up (distributed).
In the top-down design data warehouse is defined as a centralized and normalized repository for the entire enterprise. Data at the lowest level of detail (atomic data) are stored in the data warehouse and this provides good logical framework for delivering business intelligence and business management capabilities. Dimensional data marts containing data needed for specific business processes or for specific departments may be created from the data warehouse. In this architecture the data warehouse is subject-oriented (all the data elements relating to the same real-world event or object are linked together), and integrated (the data warehouse contains data from most or all of an organization's operational systems and these data are consistent), and time-variant (all changes to the data in the data warehouse are recorded and reports can show the changes), and non-volatile (data in the data warehouse is never over-written or deleted - once committed, the data is static).

The top-down design methodology generates very consistent dimensional views of data across data marts since all data marts are loaded from the centralized repository. Top-down design is also robust against business changes. Generating new dimensional data marts against the data stored in the data warehouse is a relatively simple task. A disadvantage to this methodology is that it represents a large project with a broad scope. The cost and the execution time of project are significant. In addition, the top-down methodology can be inflexible and unresponsive to changing some needs during the implementation.

In the bottom-up design first data marts are created to provide reporting and analytical capabilities for specific business processes. Data marts might contain atomic and summarized data. These data marts can eventually be unioned together to create a comprehensive data warehouse. The combination of data marts is managed through the implementation of a data warehouse architecture. Business value can be returned when the first data marts are created. The most important management task is making sure that dimensions among data marts are consistent (dimensions conform). Moreover hybrid methodologies are possible and have evolved to take advantage of the time of bottom-up design and the enterprise-wide data consistency of top-down design.

Operational systems are optimized for preservation of data integrity and speed of recording of business transactions through use of database normalization and an entity-relationship model. Codd defined five rules of normalization and operational system designers generally follow these rules in order to ensure data integrity. Fully normalized database designs often result in information from a business transaction being stored in many tables. Relational databases are efficient and have very fast performance because only a small amount of data in those tables is affected during processing a transaction. To improve performance, older data are periodically purged from operational systems. Data warehouses are optimized for speed of data retrieval. Frequently data in data warehouses are denormalised via a
A dimension-based model. Also, to speed data retrieval, these data are often stored multiple times - in their most granular form and in summarized forms called aggregates. Data of data warehouse are gathered from the operational systems and held in the data warehouse even after the data has been purged from the operational systems.

4. BUSINESS AND LOGICAL MODELING

To build a conceptual model of a data warehouse we should define a business model of an undertaking (of an enterprise) and thus we should analyze:

- Query requirements
- Data structure changes
- Time of answering
- Resource management
- Interface and verification tools
- System of making backups and of data recovery

In addition we can distinguish some detailed issues: time of answering in queries, data resource management, using operational costs and so on. By business process we mean an essential process of a given business (of an enterprise) which is engaged in storing and managing useful data. Business process examples are orders, expedition, marketing and so on.

Business modeling consists of three tasks: strategic analysis, business model creation, metadata creation. At the beginning strategic analysis is made for a given enterprise to identify and define most important business processes and their priorities. Next the business model is developed – detailed business requirements are given for all business processes and source data are verified to get proper requirements. Metadata are created at the beginning of the design process and they are results of the business process analysis. This information will be very useful in consecutive phases, when we create the conceptual model.

During strategic analysis we try to obtain a simple and approximate model of a data warehouse. We must identify and try to understand main (key) business processes of the considered enterprise and analyze their hierarchy and select them properly for implementation. Moreover, we should understand and apply general rules of the considered business to achieve the best results. Then we should have an access to all data, aid customers of this business, observe competitors and try to find the advantage of their activity. Moreover we should be ready for various changes in business and quickly react to such changes.

During strategic interviews we identify and analyze key business processes and we might check hierarchy of them with respect to the Return Of Investment size and with respect to their strategic value (concerning their meaning in making decisions) and to the implementation ease and time.
Processes should form a hierarchy according to the following rules:

- We should analyze which process causes the largest Return Of Investment
- What strategic values might be results of the implementation of a given process in the data warehouse?
- What important decisions might be undertaken due to results of a process?
- What efforts should be made in implementation and how long it will take?

The strategic analysis consists of:

- business process analysis and process identification with respect to making decision requirements,
- business process understanding through the analysis of its parameters (business dimensions) which characterize given process,
- setting priorities in processes and selecting ones with such efficient implementation that we get the greatest coefficient Return Of Investment.

For strategic analysis we should select most important persons involved in considered business. We should make interviews with heads of departments (of production, marketing, logistics and so on). We want to gain good picture of the enterprise activity and of its aims. It would be required to use a strategic questionnaire to better understand the activity and to identify main business processes. We ask questions like the following: What general problem are we dealing with? What aims do we want to reach? When (in what situations) shall we need the data warehouse? Then we identify generally key business processes and we define their detailed business requirements. Next we should analyze the complexity of these processes, find metadata and verify sources of information.

Business model is created with making business analysis for each key process. We need to define and document business measurement methods. We examine and work out a parameter list for each measure and adopt granularity for analyses. We verify business definitions and document them too. We should define dimensions, and specify their attributes. Next we check whether data required for business are accessible.

To identify business requirements we use source resources of the system (very important are end users since we make interviews with them), because they are responsible for analyzing and measuring and maintaining processes of the system. We can use also other resources:

- earlier metadata,
- entity diagrams of existing OLTP systems,
- results of earlier analyses made for this DW or for other applications.

After strategic interviews we should conduct more detailed, tactical ones concerning business requirements of processes. Example questions are: How efficiency is measured? How the production is going on? What are results of sale? What are units of sale?
We could ask also about dimensions. How often efficiency should be measured? What are features describing given product? The same about clients, contractors, stores. We should collect also information concerning IT. So we might ask: Where are data, and of what format they are, and what is the access to them? How often they should be refreshed? What is their quality?

Next we identify business processes. Here are examples of business processes concerning marketing: regional sale, Web sale, promotions, storing data, product expeditions. Often a process model may be triggered by some events and might describe a business process as a program which gets something in an input and produces some output.

Now we shall describe briefly the creation of the logical model in which we are using all products obtained in the business model. In fact, we should work out such data scheme which will implement given business model very faithfully.

During data warehouse logical model creation we should specify the structure of all tables. To this end we should adopt names of tables and their columns (taken from business analysis phase), we should take correspondent data types and assign them properly to these columns and define keys (main and foreign). We should also unify naming of tables, and of attributes, and of data types. We should select the way of implementation of relations that hold between attributes. We should also work out means of aggregation and of transformations connected with time.

5. FINAL REMARKS

Process of data warehouse designing, described above, is time consuming and without proper tools it would be very long. Main database suppliers like Oracle, IBM and Microsoft offer solutions for data warehousing with dedicated tools to warehouse management and design. Of course there are many other software vendors, but mentioned companies are in the leaders square in Gartner’s Magic Quadrant for Data Warehouse Database Management Systems (see figure 2). We shortly describe most popular design solutions.

Microsoft together with SQL Server delivers SQL Server Integration Services (SSIS), it is an ETL () platform introduced in the SQL Server 2005 release (see [11]) . Microsoft SSIS is bundled with SQL Server Professional and is a fully scalable, programmable, embeddable and extensible tool. It provides graphical tools and command prompt utilities to manage the databases which are intuitive and easy to learn and can be used to build a complete data warehouse solution. Programmers can use Business Intelligence Development Studio to create shared solutions with Analysis Services and Reporting Services, including source control, metadata integration, and the design, testing, deployment, and maintenance of end-to-end business intelligence applications.
Oracle Warehouse Builder (OWB) is an ETL tool produced by Oracle that offers a graphical environment to build, manage and maintain data integration processes in business intelligence systems (see [10]). The primary use for OWB is consolidation of heterogeneous data sources in data warehousing and data migration from legacy systems. Customers can use OWB not only for data warehousing but also data quality checking and data profiling, data cleansing and data auditing, centralized metadata management, dimensional and metadata data modeling, complex transformations of data from heterogeneous sources, and data integration.

**Figure 2. Magic Quadrant for Data Warehouse Database Management Systems.**

IBM delivered InfoSphere Warehouse, which is successor of DB2 Data Warehouse Enterprise Edition (see [1]). It includes several BI features such as ETL, data mining, OLAP acceleration, and in-line analytics. Customer can use Warehouse Design Studio which allows integrated physical data modeling, based on InfoSphere Data Architect. Very useful is SQL Warehousing Tool (SQW) for
data flow and control flow design including integration points with InfoSphere and DataStage ETL systems. The SQW is a graphical tool that generates SQL for warehouse maintenance and administration, replacing hand coded SQL.

At the end we want to conclude that there is no a strong leader among the tools to design a data warehouse and the effective use of these tools require high level experience of designers.

REFERENCES

19. D. Feinberg, M. A. Beyer, Magic Quadrant for Data Warehouse Database Management Systems, Gartner RAS Core Research Note G00173535, 28 January 2010
BUSINESS INTELLIGENCE SYSTEMS – IMPLEMENTING OPPORTUNITIES IN THE WORLD OF SMALL TO MEDIUM BUSINESS

Rafik Nafkha  
Department of Informatics  
Warsaw University of Life Sciences (SGGW)

M. Olejniczak  
University of Economics and Computer Science

Abstract: The Business Intelligence solution is not a new conception. It is large used by the IT stuff in large enterprises. Today the new fact is that the analytical capability that was once reserved for large enterprise could be offered and implemented in the world of Small to Medium Business (SMBs) and provided a solid foundation of visibility into what really matters to these companies. For the medium-sized companies the most important factor today is choosing the correct product and reliable supplier or maybe a good idea to adapt to their needs the widely available open source tools on the Internet? This article will focus on the key aspects to take into account, when considering implementation of business intelligence software by Small to Medium sized companies to lead to cheaper and reliable solution at the same time.

Key words: BI, BIS, Business Intelligence.

1. INTRODUCTION

Every day the amount of information generated is increasing. Acting on the market place, companies collect large amounts of data. These data are sometimes scattered, and access to them is often difficult because of the diverse specificities of computer systems from which they are originated.

Traditionally, companies developed month-end processes to generate financial reports or queried data at specific intervals, in order to provide data to decision makers throughout the company [1]. Furthermore, reporting processes were implemented to provide users and decision makers with regular static reports over time. Acting on the market place, the company needs require more powerful reporting tools to capture significantly higher amounts of data more often. Businesses are shifting to accommodate increased data demands, and are attempting to become proactive in their corporate planning. The appearance of Business Intelligence (BI) tools, largely has improved the processing of data which, are used by analysts in the form of reports and analysis [1].

56
Business Intelligence solutions are created with a view to the modern and dynamically developing companies. This concept is used for many years by the IT staff in the very large corporations. Access to such tools requires large financial outlays, and good trained professionals to handle them. The development of new processing technology and more effective systems caused that the access to BI tools is possible also for small and medium-sized enterprises as is clear from the Wayner report [5] in InfoWord.

Some examples of BI application in medium-sized enterprise:

- Identification of the purchasers, who limited their shopping to offer them special conditions that induce them to remain in business clients
- The implementation of dashboards and scorecards, so that managers and auditors quickly identify any deviations in the operational activities carried out and finds a potential overspend
- Comparison of turnover of individual departments to identify potential problems with employee morale Track customer orders and required shipping dates and comparing them with provisions, and then adjust the production and logistics chain in order to reduce storage costs
- Integrate operational data and historical data from a spreadsheet for analysis, while ensuring consistency and facilitating arrangements "one true version"
- Better understanding and an improved analysis of the operational activities and contacts with customers, aimed at increasing the competitiveness.
- Adjust the daily operational activities to strategic objectives and the timely identification of disagreement between them.

2. WHY BUSINESS INTELLIGENCE SOFTWARE IS NECESSARY FOR ANALYSIS AND INSIGHT?

In this age of rapidly expanding electronic data, enterprises have found it increasingly difficult to efficiently or effectively leverage their data to meet business goals. Important information may be found in every operational area of a company: sales, finance, customer service, engineering, marketing and more. Companies have traditionally tried to use their operational software programs to both run operations and analyze the resulting data. Enterprises have tried to use operational software programs designed for a specific functional area (e.g., sales automation, marketing automation, inventory management, ERP, or financial services) or export the data to Excel or a database for manual analysis.

Operational programs, however, were not designed to integrate, analyze or present data in large volumes or from multiple operational sources. Each of these approaches has drawbacks, summarized in the following points:

1. Reporting available within operational software programs (CRM, ERP, HR, accounting, etc.) approach
a. Primary purpose of operational software is for transactions and conducting operations, not analysis of the resulting data. Reporting capability tends to be very limited in terms of the variables and dimensions of analysis, requiring significant manual work (such as exporting data to a spreadsheet) to do analysis that business users really want.

b. Reporting functionality within the individual program is only designed for the data captured by that program, so the user gets just a partial picture of:
   i. How this business function impacts the rest of the business; and
   ii. How what is happening in other operational areas, in turn, impacts them.

c. Analyzing data in an operational system can slow down the capability for which the system was originally designed – running operations. This problem grows worse as data volume increases.
   i. Operational applications are designed for simply queries and to make insert/update operations quickly;
   ii. Business Intelligence requires complex queries that span many tables and operate on large row sets; and
   iii. Making large-scale analysis demands on an operating system can crash the system or slow it significantly.

d. May require expensive and time-consuming integration with other operational groups’ programs in order to achieve desired cross-functional analysis.

2. The Microsoft Excel approach
   a. Highly manual. Extremely cumbersome for data analysis, report presentation, and for ongoing data updating and maintenance. Updating an analysis with new data often requires re-running the analysis from scratch. The spreadsheet is often only easily understood by its original owner.

   b. Smaller scale data sets only. Does not handle the larger data volumes which BI solutions can easily handle.

   c. Insufficient analytic capability:
      i. Does not provide typical BI capabilities like slice & dice, drill down capabilities, pixel-perfect report layout;
      ii. Requires an analyst mindset and advanced Excel proficiency; and
      iii. Not usable for advanced analysis by the typical business user.

   d. Integrating data from multiple operational sources requires significant time and skill.

   e. Prone to error when updated.
3. Databases approach
   a. Databases such as Oracle or SQL are good places for storing data, but do not provide the breadth of analysis and presentation tools required for business analytics.
   b. Presentation tools are lacking:
      i. Lack charting, pivot table, dashboard functionality;
      ii. Requires technical knowledge to craft SQL statements and present the results as a plain table; and
      iii. Doesn't address how insights are shared in the organization.
   c. Integration of data from multiple systems may require custom work.
   d. Significantly slowed down by Business Intelligence software's large cross-functional data sets:
      i. Can crash the system;
      ii. As with operational applications, OLTP-type databases are designed for simply queries and to make insert/update operations quickly; and
      iii. Not well suited for BI type queries.

3. THE FIVE STAGES OF BUSINESS INTELLIGENCE

Traditional enterprise BI software involves five stages of taking raw data and presenting it as relevant, actionable data to users.

The five stages are:
1. The Data – The first step in the BI software process is to define which data will be loaded into the system and analyzed. The overall data structure must be defined up front, requiring companies to fully define which data they want to use before they can get started with the rest of the backend process.
2. The ETL (Extract, Transform, and Load) Engine - The ETL (Extract, Transform, and Load) Engine is the stage of moving the source data to the Data Warehouse. If this doesn’t work properly, the BI solution simply cannot be effective. Technical teams manually map source data and create definitions, map to tables, and then map to a data warehouse production environment. This process is repeated for each BI initiative within the organization.
3. The Data Warehouse - Using conventional tools, the design and creation of a data warehouse is mainly a manual process. The dimensional model and database schema for a data warehouse has to be specified and created manually based on an understanding of the business requirements. Traditionally, companies are forced to choose between two sub-optimal approaches:
a. Multiple data marts and warehouses fed by multiple ETL processes, limiting the scope of any one application OR
b. One Enterprise Data Warehouse (EDW) that provides scale but severely limits the flexibility of delivering new analytical applications.

4. Analytic Engine - Once a data warehouse is populated with data, the next step is to conduct analysis on it. Traditionally, this stage has required significant engineering resources to make a data warehouse ready for analysis. Traditional tools for end user analysis and reporting require additional metadata. Customers develop another set of metadata to navigate universes, subject areas, and cubes or to build custom cubes for analysis.

5. Presentation Layer - Reporting packages need to be implemented from multiple solutions, since only some of them provide pixel perfect reporting, while another provides banded reporting, and so on. IT needs to build and publish unique dashboards and reports for each different

4. HOW TO CHOOSE THE RIGHT BUSINESS INTELLIGENCE?

The information provided by managers in the company or the enterprise, must be obtained in the appropriate form and timely current data in order to make accurate decisions.

Any information is useful only at a specified time, and over time its value decreases until it reaches the total uselessness. Only fast-supplied information may be used to take appropriate action when appropriate situations arise in a competitive market.

The basis for decision-making are data from existing sources on various topics. Any, even the smallest company uses in its business various applications that enable more efficient data collection and aggregation, which in the general concept of utility, do not provide a clear picture of the company. The amount of data from year to year, continues to grow, and using them without appropriate business solutions is nearly useless.

Company, deciding to implement BI system should answer the following questions:

- Is the implementation of Business Intelligence really necessary?
- What kind of information is important for the business users?
- How deeply details information must be delivered?
- How often the data will be generated?
- What is the best method for data acquisition?

Often not the amount of supplied reports and their quality is crucial for the effectiveness of managers. BI should provide only such information which at the requested moment is unique and necessary in the decision making process.
Table 1. Popular BI open source systems functionality vs Oracle Business Intelligence Applications.

<table>
<thead>
<tr>
<th>feature</th>
<th>JasperSoft</th>
<th>BIRT</th>
<th>Pentaho</th>
</tr>
</thead>
<tbody>
<tr>
<td>The data definition separated from the</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>definition of the report</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Multiplicity of predefined types of data</td>
<td>Yes – in</td>
<td>Yes – in</td>
<td>Yes – in</td>
</tr>
<tr>
<td>sources</td>
<td>addition</td>
<td>addition</td>
<td>addition</td>
</tr>
<tr>
<td></td>
<td>Java</td>
<td>Java</td>
<td>Java</td>
</tr>
<tr>
<td></td>
<td>objects</td>
<td>objects</td>
<td>objects</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>The presence of a convenient Client tools</td>
<td>iReport -</td>
<td>BIRT Report</td>
<td>Pentaho Report</td>
</tr>
<tr>
<td>assist in the formatting Templates</td>
<td>the most</td>
<td>Designer -- plug-in for Eclipse. Good concept, allows also in parallel work in Java. For developers</td>
<td></td>
</tr>
<tr>
<td></td>
<td>developed</td>
<td></td>
<td>Designer - Requires refinement [2].</td>
</tr>
<tr>
<td></td>
<td>tool;</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Programming</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>experience</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>is required.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ensuring corporate Reporting Platform</td>
<td>JasperServer. - limited functionality. lack for community version</td>
<td>Available Products Commercial.</td>
<td>Pentaho Server</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ready functionality delivering reports</td>
<td>Only via email</td>
<td>No</td>
<td>Printing, supply via email</td>
</tr>
<tr>
<td>frequently used channels.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>multilinguage support</td>
<td>Only through the Java mechanisms. Template must be created with using these mechanisms.</td>
<td>Only through the Java mechanisms. Template must be created with using these mechanisms.</td>
<td>Only through the Java mechanisms. Template must be created with using these mechanisms.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>The bursting possibility, ie the division</td>
<td>No</td>
<td>No</td>
<td>available</td>
</tr>
<tr>
<td>single file in the resulting report for</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>distribution to multiple recipients.</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

5. THE BI SYSTEM COST

In this chapter, I would like to present the best known, available for free tools for reporting, using Java technology, versus the commercial tool of Oracle. Because of the technology so they can be considered as products rival to Oracle BI Publisher. Some of the presented products can also boast a number of implementations not only in the small and medium-sized companies, but in large corporations. Users will have the opportunity to compare the individual license cost [5] of reporting tools available under:

- Pentaho Open BI Suite [2]
- JasperSoft Business Intelligence Suite [3]
- Business Intelligence and Reporting Tool (BIRT) [6]

These products will be compared with Oracle BI Publisher [4]

Anyone probably knows from numerous experiences, comparing the cost of individual license is not a simple matter. Some give money per processor, others on the number of users and still others on the number of years. However, it is so important criterion that is useful to know at least an order of magnitude required to bear the cost.

Table 2. Popular open source BI systems license cost (community edition) for small-medium sized businesses.

<table>
<thead>
<tr>
<th>Producer</th>
<th>System</th>
<th>Destiny</th>
<th>License cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>JasperSoft</td>
<td>JasperReports Community Edition</td>
<td>Small to Medium business</td>
<td>free</td>
</tr>
<tr>
<td>Pentaho Corporation</td>
<td>Pentaho BI Suite Community Edition</td>
<td>Smal to Medium business</td>
<td>free²</td>
</tr>
<tr>
<td>BIRT RD – BIRT Report Designer</td>
<td>BIRT</td>
<td>Smal to Medium business</td>
<td>free</td>
</tr>
</tbody>
</table>

*Source: PLOUG Conference – Szczyrk 10. 2008*

If we assume, as a basis for comparison, the cost of fees for the 2 processor platform in the first year, we can see that fees for Oracle BI Publisher is shaped more or less at 7 to 10 times greater than other products. These are just about the fees. Additionally to this must be added the costs of infrastructure, other necessary software and implementation costs.
### Table 3. Popular BI open source systems license cost (enterprise edition) vs Oracle Business Intelligence Applications.

<table>
<thead>
<tr>
<th></th>
<th>Oracle BI Publisher</th>
<th>JasperSoft Server Professional</th>
<th>BIRT + Actuate iServer Express</th>
<th>Pentaho Reporting</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>46 000 $ per CPU, Support for additional approximately $11 000 per year or $1,000 for the user as part of Oracle BI Standard Edition One (up to 50 users)</td>
<td>Subscription approximately 8 000 $ to 2 CPUs per year (based on the [Way08])</td>
<td>9 000 $ for 25 users</td>
<td>Subscribe for $12 000 4 CPU per year (data under [Way08])</td>
</tr>
</tbody>
</table>

*Source: PLOUG Conference – Szczyrk 10. 2008*

### 5. CONCLUSIONS

Quite careful look at open source products showed that they are so mature and proven that they can even be taken into account when making decisions. We should know that not all functionality is being implemented, so depending on the needs of the project the key to the proper choice might be just the functional requirements.

Another crucial element is such that small and medium companies usually do not have the proper specialist during the implementation, in such a situation, a small company leading the implementation based on its own staff is very risky.

### REFERENCES

2. [BID06] Pentaho BI Server Documentation, 2006

---

2. System Pentaho BI Suite Enterprise Edition is commercial product. Cost of license starts from $300.00;
Abstract: In the paper the modeling of business processes with the established graphic notations, i.e. Unified Modeling Language UML and Business Process Modeling Notation BPMN will be presented. The possibilities of the both notations will be compared with respect to methodologies supporting their usage and the notation elements.

The UML as the standard notation of the methodology Unified Process comes from the domain of the object-oriented design of information systems and is currently regarded as a world quasi-standard modeling notation promoted by the OMG consortium. On the other hand, the BPMN promoted by the Business Process Management Initiative associated with some traditional methodologies in the domain of the management information systems and it promises a better support of the SOA concepts.

The tools supporting the modeling of business processes in both regarded notations will be also considered in the paper.


1. INTRODUCTION

In the contemporary world information systems have gained increased contribution in usefulness for business organizations. The modeling of the business processes in the whole enterprise and business process reengineering focus nowadays the attention, replacing the previous emphasis on modeling merely the customer-dedicated processes.

According to [1] a business process is a collection consistent of some activities, which have a goal to reach a value in a form of a business product. For this aim required are the resources, other products intermediate goods, resources and the clear rules for the creation of the product. The business process should be definable and measurable. All business processes have some common characteristics like: an environment, the clients, the information processing, a communication, the queues and resources and the final product.

There are some methodologies, initiatives or reference models that support the business process modeling, such as:

- RUP – Rational Unified Process;
- Business Process Management Initiative BPMI;
- UMM – UN/CEFACT’s Modeling Methodology;
- ARIS – Architecture of Integrated Information Systems;
- SCOR – Supply Chain Operation Reference-Model.

Each of them supports a specific graphic notation integrated into the process. In the domain of the object-oriented analysis and design (OOA/OOD) for IT projects the UML [2] within a (Rational) Unified Process UP [3] is currently regarded as a world quasi-standard modeling notation promoted by the world consortium Object Management Group OMG [4]. The Business Process Modeling Notation BPMN [5] promoted by the Business Process Management Initiative BPMI originated from the domain of management information systems. The UMM methodology is supported by the organization United Nations Centre for Trade Facilitation and Electronic Business [6]. The Architecture of Integrated Information Systems ARIS, a methodology represented by the IDS Scheer group [7], and the Supply Chain Operation Reference-Model SCOR reference model [8] are the instances of other methodologies and reference models also offering some possibilities for the aims of the business process modeling.

In this paper the usefulness of two graphic notations - UML and BPMN for business process modeling, with respect to methodologies supporting their usage and the available tools will be considered.

The further content of this paper is structured as follows. In the Section 2 the object-oriented (R)UP methodology, and the suitability of its notation the UML for business process modeling aims will be considered. The Section 3 will include some considerations about BPMN and the modeling tools available for this graphic notation. Section 4 contains a résumé of the remaining methodologies, their notations and the comparison of the visual modeling tools. Section 5 contains a conclusion of the paper.

2. OBJECT-ORIENTED SOFTWARE DEVELOPMENT – UP METHODOLOGY AND THE UML

Current accepted object-oriented software development process models are no longer like the waterfall model, but are more spiral, and iterative-incremental process based models. A prominent example of such a process is The Unified Software Development Process UP originated in the best practices of OMT, method of Grady Booch and Ivar Jacobson’s Objectory [9]. The Unified Process can be considered as de facto standard, UML-based, object-oriented software engineering process. The industrial variant of UP is the Rational Unified Process RUP, a process product (offered by IBM [10]), which is a process framework that can be adapted and extended to suit the needs of the organization and has comprehensive tool support for the development process of the software-intensive systems.
The (Rational) Unified Process is organized within two dimensions: the time and process components. In the time dimension (horizontal) there are four main phases named Inception, Elaboration, Construction and Transition. Each of them consists of a sequence of iterations separated by the milestones.

The process components, i.e. Core Workflows on the vertical axis, are analogous to the steps in the classical waterfall-based development approaches. The workflows (disciplines), which are organizing the activities, the produced artifacts and the workers, are the key notions in the description of the process. The Rational Unified Process RUP also contains in the “Core Supporting Workflows” - the means for the configuration and change management, which were lacking in the origin Unified Process (see Figure 1).

**Figure 1. The RUP process - phases and workflows**

The Rational Unified Process is method that has been based on the industrial “best practices” for the object-oriented software development. The process has a comprehensive tool support and it may be regarded as an object-oriented industrial quasi-standard. Nevertheless, as an object-oriented process it is concentrated on the business process modeling only in the first process component, i.e. Business
Modeling Discipline, and further mainly in the Inception and Elaboration phases - see Figure 1.

As a methodology the RUP promotes the usage of the “4+1 views” for the aims of the modeling of the software-intensive systems, and within these views there are several kinds of the different UML diagrams available (thirteen in the UML 2.0 version [4]). For the purpose of the business process modeling of information systems, the two kinds of the UML diagrams – the activity diagram and the use case diagram could be applicable.

In the next Sections some considerations about Business Process Modeling Notation BPMN, the comparison of the BPMN with the UML and an overview of the visual modeling tools will follow.

3. BUSINESS PROCESS MODELING NOTATION AND THE VISUAL MODELING TOOLS

The Business Process Modeling Notation (BPMN) emerged in 2004 defined by the Business Process Management Initiative, which is now a part of the Object Management Group consortium. The BPM Notation has derived from diverse previous notations like, for example the formal notation Petri Nets and Extended Event-driven Process Chains (EPC) [11].

In the BPMN method at the first the message flow and process control is modeled. This graphic notation provides only one kind of diagram, the so-called Business Process Diagram (BPD). It is a kind of a flowchart with the following basic elements [5] like:

- Flow objects;
- Connecting objects;
- Swimlanes;
- Artifacts.

The Flow Objects have their subcategories, i.e. the Events, Activities and Gateways. The Connecting Objects encompass symbols to represent the Sequence Flow, Message Flow and the Association. The processes can proceed in Pools and Swimmlines. The Artifacts (non-active elements) enclose Data Objects, Annotations and Groups.

A usage of just Flow Objects and the Connecting Objects allows designing models with low level of details included, but which are still sufficient for the aims of a documentation and basic communication between the designers groups and the stakeholders.

There are three categories of business processes in BPMN: a private process, a public process and a cooperation process. One of the assumptions of the BPMN
was to be designed user-friendly for the users in the e-business domain. The numerous graphical language elements are grouped into four above categories. The graphical language elements embrace many usable types and forms.

There several types of events are available, i.e. – initial, intermediate and terminal events, with some usable types like: undefined, message, time, rule, connection, and multiple. Also the foreseen possibilities for modeling activities in BPMN include different forms, such as the embedded process, the expanded process, the repeatable and concurrent activity.

There are also several different types of decision gates offered, like XOR controlled by data or events, AND, OR and composed. Moreover, several process types are distinguished in BPMN, like: normal, not controlled, conditional, default, exception flow, message and compensation/association. Additionally, there also are rules given for the allowed connections between the diagram objects. Each of the BPMN graphic elements in the diagram should be also documented in a table containing the description of the possible element’s attributes. The developers of BPM Notation have specified the amount and format of these attributes in the according BPMI documents.

There are several tools available for the aims of business process modeling in the Business Process Modeling Notation. Additionally some of the tools also enable to design with the object-oriented UML models and the source code generation.

Several tools have been installed at the University of Zielona Góra at the Faculty of Mathematics, Computer Science and Econometrics, some of them only for a limited period of time as the trial versions. The following tools have been tested for their appropriateness for the aims of the business process modeling for information systems:

- iGraphx, a professional tool developed by Corel Corporation [12];
- EclipseUML, an open source program Eclipse offered by IBM with a plug-in Omondo [13];
- BP Visual Architect developed by Visual Paradigm [14];
- Enterprise Architect developed by Sparx Systems [15].

The first two tools and the last, i.e. the iGraphx, the BP Visual Architect and the Enterprise Architect, offer the possibility of visual modeling of diagrams in both regarded notations - the BPM Notation, as well as in the UML.

An open source program the EclipseUML supports the visual models with the object-oriented UML diagrams only. The three first tools – the iGraphx, the EclipseUML and the BP Visual Architect include additionally the possibility of a generation of the BPEL code. The Enterprise Architect has not such possibilities available, but it supports the model driven-architecture MDA concepts.

The first mentioned tool, i.e. the iGraphx offered by Corel Corporation, has been destined for the professional usage. Therefore this product is very costly in
comparison with the other tools like BP Visual Architect and Enterprise Architect, which both tools are approximately at the same cost level. On the other hand, the significant advantage of the EclipseUML over the other regarded tools is that it is free available, as it is offered an open-source program.

The serious constraint of the Corel’s iGraphx and Sparx Systems’s Enterprise Architect it that both tools are destined for the Windows platforms only. On the other hand the IBM’s EclipseUML and the Visual Paradigm’s BP Visual Architect, according to theirs developers, are tools that are suitable to work with any Java virtual machine.

4. OTHER METHODOLOGIES, INITIATIVES OR REFERENCE MODELS SUPPORTING THE BUSINESS PROCESS MODELING

The UMM methodology [6] is a formal method for description of process scenarios compatible with Open-edi Reference Model for business transactions published in the ISO/IEC 14662:2004 norm. It describes the structure of modeling, the description techniques, protocols and e-business services. The model uses the Business Operational View BOV to describe the business aspects of business transactions and the Functional Service View FSV for the Information technology aspects of business transactions. The UMM methodology uses the BOV view of the Open-edi Reference Model that has been subdivided into four to layers: the Business Domain View, the Business Requirements View, the Business Transaction View, and the Business Service View. In each development iteration the designer should analyze all of the views in a sequential way and construct the following appropriate diagrams: the value chain models, then the UML interaction and further the activity diagrams. The UMM methodology offers an enhancement of UML and contains a business process and business information meta-model, its business orientation supports the reuse of the modeling artifacts.

Together with BPMN the notation for description of the interaction between the XML-based solutions the Business Process Execution Language BPEL have been introduced. Due to the mutual dependency of both technologies, the BPMN notation is suitable for supporting the concepts of the Service-oriented architecture SOA [16], [17], [18]. The conjunction of graphical BPMN models with BPEL code enables an easy and fast integration of needed changes into the process.

5. CONCLUSION

In the standard object-oriented notation UML there are several kinds of diagrams available (eight in UML 1.4 and thirteen in the UML 2.0). They are intended for the documentation of the static application structure, its dynamic behavior and also management and organization of a software system. The UML diagrams are means, based on diverse “best-practices”, destined for the software
architects and software engineers to visualize and model the developed software systems.

The quasi-standard standard object-oriented process RUP is an architecture-centric and use-case driven process and therefore an object-oriented development of a software-intensive system concentrates itself in the beginning on the use case diagrams (use case-view) and the static aspects of the system in form of the class diagrams.

The Business Process Modeling Notation as a process-centric notation follows a different approach to business process modeling, than the object-oriented paradigm. The BPMN, unlike UML, has strong mathematical foundation and is targeted at the business process analysts and business process developers. It disposes of only one kind of a diagram - the Business Process Diagram and that enabling multiple views of the modeled process.

In the e-business domain the modeling control and message flow of processes comes to the foreground. For the aims of the business process modeling in a comparison of the UML with the Business Process Modeling Notation BPMN only two of the UML’s behavior diagrams could be taken into account: the use case diagram and the activity diagram. The use case model, which identifies what system is supported to do and the system environment, is too general to be taken into consideration. For that reason we can only compare the UML’s activity diagram with the BPMN’s Business Process Diagram. Both kinds of diagrams have possibilities of the modeling the flow control of the business logic, but the BPMN has possibilities for more detailed, in-depth description. Moreover, the Business Process Modeling Notation as based on the process execution metamodel can be conversed to the fully executable BPEL process.

Therefore, BPMN offers a more natural and user-friendly design process practice in the e-business domain, than the object-oriented paradigm in such a way as the RUP does. In the BPMN method preferential the message flow and process control should be modeled. The graphic BPM Notation provides one wide-ranging type of diagram, the so-called Business Process Diagram (BPD). Its aims may be compared with those of the UML’s activity diagram. The BPMN’s Business Process Diagram offers the much more comprehensible means to precisely modeling of the business processes than the activity diagram in the UML. Moreover, the Business Process Modeling Notation BPMN is more suitable for supporting the concepts of the Service-oriented architecture SOA than UML.

At the market there are several tools offered for the aims of the visual modeling, like iGraphx, a full professional tool developed by Corel Corporation, BP Visual Architect developed by Visual Paradigm and Enterprise Architect developed by Sparx Systems, which offer the possibility of modeling in the BPM Notation as well as in the UML language. In the polish market there are two other enterprises offering BPMN modeling tools – the polish firms Rodan Systems with
the product OfficeObjects®WorkFlow and DYSANT Software with DYSANT nWorkflow. The Rodan Systems offers in the provided tool the possibility of generation of the BPEL on the base of the BPMN description.

REFERENCES

5. BPMN 1.0 http://www.bpmn.org/Documents/ OMG Final Adopted BPMN 1-0 Spec06-02-01.pdf
8. SCOR 8.0 Overview Booklet http://www.supply-chain.org/
10. IBM online: http://www-306.ibm.com/software/rational
THE SELECTION PHASE IN DECISION MAKING AIDED WITH A DECISION SUPPORT SYSTEM

Silva Robak
Faculty of Mathematics, Computer Science and Econometrics
University of Zielona Góra

Andrzej Pieczyński
Faculty of Electrical Engineering, Computer Sciences & Telecommunication
University of Zielona Góra

Summary: The paper presents some aspects of a decision-support system, which takes into account selected considerations that are necessary in a choice of a trade customer. Some of the characteristics under investigation are imprecise or unpredictable, therefore the authors suggested their descriptions on the basis of the fuzzy logic and probability methods for these aims. The considered decision-support system has been implemented and validated in the environment of the Exsys™ expert system.

Key words: business intelligence, decision-support systems, heuristic decision tree, fuzzy logic, expert system, Exsys.

1. INTRODUCTION

In the contemporary world information systems have gained increased contribution in usefulness for business organizations. More specifically, the change comes from extended opportunities they give in management decision-making process.

Decision-support Systems (DSS), so called “business intelligence” -systems, are of great value in improved decision-making process for senior and middle management, operational management, and also individual employees and project teams. The decision characteristic depends on the decision-making level. High quality decision-making is especially important in unstructured and semi-structured decisions. There are certain stages in decision-making such as an intelligent decision design, a choice between the solution’s alternatives and an implementation (monitoring of the results). Moreover, there are different types of DSS, basically there are model-driven or data-driven systems. The model-driven systems, based on theory and analytical models, have good analytical capabilities and usually include user-friendly graphical user interfaces. The data-driven systems are capable of analysis of large pools of data, often collected from transactions processing systems in data warehouses or data marts. The DSSs enable carrying out analyses by
application of the libraries of diverse analytical models like: the statistical models, the optimization models and the forecasting models. They also provide means for accomplishment of the “what-if”-sensitivity analysis. In case of the problems with poor quality of analytical models, we may complement them with the heuristic data mining methods.

In this paper we consider the selection phase in decision making for interactive, focused and specified decisions, as it functions in the domain of agriculture transport. Selecting from diverse solution alternatives for a voyage-estimating program for a bulk transport of sugar-beets with trucks, will be supported aided with fuzzy logic applied in the modules of the Exsys expert system.

The expert system, developed and applied at the University of Zielona Góra will be presented in the paper. In Section 2 we present the considerations about a voyage-estimating program for a bulk transport with trucks, and also a choice of the stable and variable features in the voyage planning. In Section 3 there are some considerations about the fuzzy set theory. In Section 4 we present our fuzzy expert system application for the decision support in truck transport intended for the sugar factory. In Section 5 we conclude our work.

2. A VOYAGE ESTIMATING PROGRAM FOR A BULK TRANSPORT WITH TRUCKS

In our paper we analyze the instance of a sugar company. The company needs to transport the contracted white beets from the farmers to the own base in the factory. The company uses its own trucks, but when needed, the firm may also rent some external means of the transportation. A DSS may support the managers with its analysis of financial and technical aspects of the voyage contracts. Technical aspects in the voyage estimation include such concerns as: distances between the contracted farmers and the factory, the cargo capacity of the available trucks, and further parameters such as trucks speed, the distances between the farmers, a fuel consumption and also the load patterns of the trucks. The technical details of the trucks and the place distances have a stable character. The financial aspects of the transport include the truck and time costs, fright rates for a cargo, and other expenses. The first mentioned financial factor, which includes the costs of fuel and labor, might vary because of some dependencies on the weather conditions, the type of the road and the time spent on a delivery, and for loading und unloading the sugar-beets, etc.

The considered DSS will operate on a computer system connected to the analytical models that are suitable to analyze data taken from the diverse data files with needed details, like: trucks file, place distances, fuel consumption of the trucks, trucks charter hire history and the cost file (see Figure 1).
Figure 1. Decision support system for the truck transportation for a sugar factory

In the next Section some basic considerations about the fuzzy sets will be presented, that will be applied for modeling some of the imprecise features of the transportation.

3. BASIC NOTIONS – FUZZY SETS THEORY

A The concept of fuzzy sets as a collection of objects that might belong to a certain degree - from 1 (full membership) to 0 (full non-membership) through all intermediate values, was introduced by L.A. Zadeh [8]. The intensity of belongingness assigning each element a number from the unit interval, was dependent on the membership function.

The definition of a fuzzy set is as follows:

Supposing that $X$ is a universe of discourse i.e. the set of all possible elements with respect to a fuzzy property, then a fuzzy set $A$ in $X$ is defined as a set of ordered pairs $\{x, \mu_A(x)\}$ , where $x \in X$ and $\mu_A : X \rightarrow [0,1]$ is the membership function of $A$.

Further considerations based on of the fuzzy representation are presented among others in [2,4,8]. In our paper the fuzzy representation will be used to describe some non-technical aspects of voyage estimation for bulk transportation. Based on the file records and stored knowledge, fuzzy logic may be applied to support to acquire a rule-based description of a decision tree, generated from the shell expert system.
4. FUZZY EXPERT SYSTEM APPLICATION FOR THE DECISION SUPPORT IN TRUCK TRANSPORTATION

A decision support system for sugar factory will generate a suggestion on how to make a decision for the choice of a client for the transportation of the contracted white beets between the client and the factory. The profit of the truck owner is dependent on some factors like:

- The transport cost defined as aggregation of distance between the base and a client, the degree of the filled space in a truck volume and the average fuel consumption of a truck. The last parameter is additionally dependent on the type of road, and further taking into consideration the influence of the weather conditions on the road;
- The client profile defined as aggregation of bulk cargo capacity of the white beets contracted between the client and the sugar factory, and also a kind of the client and his importance for the factory.

The expert system will deduct, having analyzed the above voyage features.

Figure 2. The decision tree used in the fuzzy expert system

Source: own.

4.1 A KNOWLEDGE BASE OF THE FUZZY EXPERT SYSTEM

The knowledge base use in our expert system is shown in Figure 2. This knowledge was designed as a decision tree representation and it contains three
decision levels.
Most of the transportation features are continuous variables. A few of the above mentioned features will be defined with the description based on fuzzy sets [6, 7].

*Figure 3. The membership functions used for modeling customer profile - CP (a), quality of the road - KW (b), and truck owner profit - TOP (c)*

The kind of customer (his importance for the sugar factory) and the quality of the road have been described by means of three fuzzy sets; i.e. good, bad, very bad road - see Fig. 3b. The customer profile variable has been described also by means of three fuzzy sets namely: new, stable and VIP (see Fig. 3a). The company profit has been defined with five fuzzy sets: very low, low, medium, high and very high (see Fig. 3c).
The aggregation mechanism is presented below as formulas (1)-(3). The customer bulk resource may be described in the following way [5]:

$$CBR(BV, PC, W_{PC}) = \left[ \sum_{i=1}^{V} \mu_i(PC) \cdot w_{PC_i} \right] \cdot BV$$

(1)

where: $BV$ is the bulk volume, $PC$ is the profile of customer, and $W_{PC}$ is the customer importance factor.

The embarkation rate of truck, distance between base of a sugar factory and customer place and fuel consumption of trucks are described in the following way:

$$TC(ERT, w_{ERT}, D, w_D, FC, w_{FC}) = ERT \cdot w_{ERT} \cdot D \cdot w_D \cdot FC \cdot w_{FC}$$

(2)

where: $TC$ is the transport cost, $ERT$ is a embarkation rate of truck, $w_{ERT}$ is the embarkation rate of truck influence on the transport cost, $D$ is a distance between base of sugar factory and customer place, $FC$ is a truck fuel consumption, $w_D$ and $w_{FC}$ are the distance and fuel consumption influence on the transport cost.

The trucks fuel consumption is described in the following way:

$$FC(AFC, KW, W_{KW}) = \left[ 1 + \left[ \sum_{i=1}^{V} \mu_i(KW_i) \cdot w_{KW_i} \right] \right] \cdot AFC$$

(3)

where: $FC$ is the fuel consumption of truck, $AFC$ is a average fuel consumption given by the car producer, $KW$ – is a kind of way, $W_{KW}$ are fuzzy value importance vectors of the average fuel consumption and kind of way.

4.2. IMPLEMENTATION OF THE FUZZY EXPERT SYSTEM

The proposed solution was implemented in the expert shell system EXSYS Developer. The knowledge base for this system shown in Fig. 4 is the basis for automated generation of knowledge rules. Such kind of knowledge representation is advantageous because of many possible hierarchies of conclusions with diverse certainty levels. The conception of a rule established knowledge base, allows for acquisition of conclusion sets with assigned certainty factors.

The applied shell expert system provides consistency checks on the knowledge base. In this version, the tree representation uses 216 nodes. The user interface construction was based on special script language, implemented in the EXSYS shell program (see Fig. 4).

Therefore, for small expert systems development, application of a dedicated system may be a better solution.
Figure 4. The fuzzy expert system answer – a screen dump.

a) starting screen,

b) definition of a kind of road variable,
c) screen of activated rules

**Rule 2 - Decision tree fuzzy:2 : TRUE**

**IF:**
- Distance between sugar factory base and customer place is < 5 km
- and Transport capacity order is <10 Ton
- and Customer best resource is < 3 Ton
- and Road is bad

**THEN:**
- Truck owner profit is medium - Confidence= 85

**Rule 3 - Decision tree fuzzy:3 : TRUE**

**IF:**
- Distance between sugar factory base and customer place is < 5 km
- and Transport capacity order is <10 Ton
- and Customer best resource is < 3 Ton
- and Road is very bad

**THEN:**
- Truck owner profit is low - Confidence= 83

d) answer of expert system

<table>
<thead>
<tr>
<th>Confidence</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>496</td>
<td>Truck owner profit is low</td>
</tr>
<tr>
<td>34</td>
<td>Truck owner profit is medium</td>
</tr>
<tr>
<td></td>
<td>Distance between sugar factory base and customer place is &lt; 5 km (Conf=1)</td>
</tr>
<tr>
<td></td>
<td>Transport capacity order is &lt;10 Ton (Conf=1)</td>
</tr>
<tr>
<td></td>
<td>Customer best resources are &lt; 3 Ton (Conf=1)</td>
</tr>
<tr>
<td></td>
<td>Road is good (Conf=0.035) AND bad (Conf= 4) AND very bad (Conf= 6)</td>
</tr>
</tbody>
</table>

*Source: own.*
5. CONCLUSIONS

The proposed system supports decision-making process facilitating the selection of appropriate transport offers. In the system, in the fuzzy representation, such aspects as: a customer profile, a kind of a way, and a truck owner profit have been taken into account.

The fuzzy representation of knowledge made it available to construct a transparent knowledge base and a flexible inference system.

The implemented solution allowed us to validate the anticipated thesis, as shown in our paper on a simple example. The used shell expert system Exsys facilitates rapid development of an expert system [9]. Nevertheless, one serious disadvantage of the implemented system could be the high cost of Exsys, thus not applicable for diminutive expert systems.

REFERENCES

9. url: http://www.exsys.com
EFFICIENCY OF XQUERY IN PROCESSING NATIVE XML DATA STORED IN RELATIONAL DATABASE

Marek J. Śmiański
Marcin Drozd

Faculty of Mathematics and Computer Science,
University of Lodz

Abstract: This paper is focused to the selected issues related to use of XML Query Language (XQuery) in processing XML data stored in relational database based on SQL Server 2008 and Oracle 11g. The first section introduces generally the topics of processing native data in database systems. Next, it was briefly shown the state of the implementation of XQuery in two most popular commercial database management systems. The fundamental features of language were discussed and indicated most significant differences in relation to the recommendation of W3C. The last section establishes study on efficiency of XQuery queries performed on the test relational database containing weather information provided by National Oceanic and Atmospheric Administration.

Key words: databases, queries, XML, XQuery, SQL Server, Oracle.

1. INTRODUCTION

Storing and processing native XML data in database applications is no longer anything new, but the capabilities offered by various relational database management systems vary in successive versions and are miscellaneous both in terms built-in functionality and efficiency. Advanced standardization carried out by the World Wide Web Consortium should lead to ensure compatibility and integration of various technologies, especially those related to management of migrating data in the network. However, it is easy to note that software developers tend to comply with standards only in limited way, often introducing own additional solutions aimed at increasing competitiveness. On the other hand it should be emphasized that the implementation of XML data type, XML Schema, XQuery language, etc. is only part of the activity of DBMS vendors in developing XML technologies. Both Microsoft and Oracle are the active members of the XML Query Working Group, whose aim is to provide flexible query facilities to extract data from real XML documents, databases or other persistent storage that are viewed as XML via a mapping mechanism [8].

Microsoft introduced the basic capabilities of handling native XML data for the first time in SQL Server 2000. In the version 2008 are supported such standards as
XML namespaces, XML Schema, XQuery, XSLT, SQL/XML and SOAP. It is also possible to index of the XML data (one primary and three secondary indexes), full-text search XML data, modify XML data using XML DML language and use the FOR XML clause in SQL queries. Additionally, an ability of representation of spatial data in Geography ML was introduced in accordance with guidelines of Open Geospatial Consortium [4]. In turn, Oracle 9i R2 (2001) offered Oracle XML DB technology, which allows advanced processing XML data. Currently, in version 11g R2, support for native data includes: XML namespaces, XML Schema, XQuery, SQL/XML, XSLT and SOAP/WSDL. Moreover, in databases containing XML data it is also possible to index of XML data (XML Index), full-text search XML data, compare two versions of XML documents (SQL-function XMLDiff) and use of XLink and XInclude to represent document relationships (as a support of XML Repository particularly in the case of fine-grained manipulation of documents) [1][2]. Therefore, it is noticeable a trend of companies to implement the obvious functionalities while offering support of unnecessary commonly used standards to significantly diversify the software.

2. IMPLEMENTATION OF XQUERY IN SQL SERVER AND ORACLE

XML Query Language is typed, declarative language for querying XML data sources which is similar to SQL from some point of view. But unlike him, XQuery defines a data model based on sequences of nodes (according to XPath 2.0, because technically XQuery is considered as its extension) and allows to construct a new data (XML). XQuery is also referentially transparent, in the sense that „the same expression evaluated in the same context returns the same value” [1]. The most important expression of language is expression FLOWR which both the construction and operation are similar to SELECT statement in SQL.

Relational query language used in SQL Server 2008 (Transact-SQL) contains XQuery as a subset to manipulate of XML data through XML data type methods (query, value, nodes, exist, modify). In turn, Oracle 11g R2 provides support for XQuery through a native implementation of SQL/XML functions (XMLQuery, XMLTable, XMLExists and XMLCast) in Oracle XML DB. Moreover, for the convenience of users Oracle added to SQL*Plus language XQUERY command, which allows to enter XQuery expressions directly (so that XPath expressions as well). In this way, SQL*Plus has become a kind of XQuery command-line interpreter, which is not offered in SQL Server.

Two discussed implementations of the XQuery language are obviously based on the W3C recommendation, so both their capabilities and usage are quite similar. Both versions manipulate sequences of nodes, support namespaces, have built-in FLOWR expression, conditional and quantifier expression, support operators and functions provided in W3C standard (although Oracle – almost all of them, while
MS SQL – only basic), among which there are the constructors of XML data and aggregate functions. A very important functionality offered by both DBMS is the ability to integrate XML and relational data by creating the requested XML views, on the fly. This is possible by the extension functions of XQuery: sql:column in SQL Server and ora:view in Oracle. However, they are used quite differently, because the first one returns an atomic value from relational column as an XML value, and the second one returns a sequence of nodes, one for each row (so, if atomic value is requested, an additional iteration using FLOWR is necessary). These features allow to select XML data based on condition imposed on relational data or aggregate data of both types from any tables.

The following features specified by the XQuery standard can be regarded as the most important among the basic defects of both original implementations [1][4]:

- it is impossible to create user-defined functions (and therefore modules),
- the aggregate functions does not work on sequences that mix data types,
- there are difficulties in typecasting,
- there are no sort modifiers which control how order by clause of FLOWR expression should sort the empty sequences and NaN.

Moreover, SQL Server does not allow access to external XML data sources via URIs by function fn:doc and does not have many basic functions (e.g. absolute value fn:abs), which sometimes could be a significant impediment. In turn, Oracle does not support XML identifier xml:id (no function fn:id and fn:idref) and datatype xs:duration (user must choose between year-MonthDuration and dayTimeDuration). Most of the remaining lacks either does not limit capabilities of data processing in a significant way or concerns the functionalities provided in XQuery recommendation as optional.

A few words should be devoted to general interface between SQL and XQuery, without which the integration of XML and relational data would not be possible. It allows to take advantage of both languages, offering a flexible tool that significantly enhances the processing any type of data stored in database. In SQL Server, two of the previously mentioned methods of the XML data type are particularly important because of the processing XML instance stored in XML column (or variable). Namely, the methods query and value allow to perform an XQuery, returning the result as untyped XML and a scalar value, respectively. Oracle offers the functions of SQL/XML, inter alia XMLQuery and XMLTable, which return aggregating items as a single XML document (or fragment) and relational table whose rows contain items from XQuery sequence, respectively. It should be noted that the writing XQuery queries is much easier in SQL Server (but it is the only method, in contrast to Oracle), as it is shown below:

SELECT XML_col.query('XQuery') FROM Table (: SQL Server :)
It seems that Oracle greatly extended support for XQuery, while Microsoft has done little in comparison with the previous versions (although missing LET clause to FLOWR expression was finally added). On the other hand, SQL Server allows to use of more advanced features in a simple way.

4. STUDY OF EFFICIENCY OF XQUERY

In order to study the efficiency of XQuery queries we made tests on the same relational databases in SQL Server 2008 and Oracle 11g. Tests were carried out on the Dell PowerEdge 1800 with two 3GHz Intel Xeon processors and 4GB of RAM under Windows Server 2008.

The processed XML data come from National Digital Forecast Database provided by National Oceanic and Atmospheric Administration [7]. Database consists of three tables, containing basic information about weather stations, U.S. states and territories, in which the stations are located, and detailed information about weather conditions from September 2009. All data is stored in relational columns, except of measurements, which have been placed in XML column in the form of original XML documents.

Results of selected tests are presented below in the form of a short description of the query, XQuery code and the following statistics: CPU execution time of query in seconds (illustrated in charts) and buffer hit ratio ($bhr$), calculated by the formula $bhr=(lr-pr)/lr*100\%$, where $lr$ and $pr$ denote a number of logical and physical reads, respectively. $bhr$ can be considered as one of the most important metrics, since it indicates the efficient performance of I/O operations in data processing. Oracle suggests such tuning of database to maintain the level of $bhr$ at least 95%. To increase the utility of our study, we used the tables of measurements of three different sizes (with 400, 800 and almost 1200 thousands rows).

Example 1: Select all measurements of temperature in °F and wind speed from the station identified by 'KSAC'

```sql
query(''<pomiar>
  <temp>{data(/observation/temp_f)}</temp>
  <wiatr>{data(/observation/wind_mph)}</wiatr>'')
```

To filter the data we used two methods: (a) SQL clause WHERE and relational column (WHERE STATION='KSAC') and (b) XQuery conditional expression and XML column (if (/observation/station_id="KSAC") then...else ...). Note that in the first case we obtain only measurements from a given station, while in the latter one – table containing all rows with value Null corresponding measurement from other station.
Table 1. Results of Example 1.

<table>
<thead>
<tr>
<th></th>
<th>400000 records</th>
<th>800000 records</th>
<th>1172370 records</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Oracle 1</strong></td>
<td>$t=1.06$</td>
<td>$bhr=1.82%$</td>
<td>$t=3.41$</td>
</tr>
<tr>
<td><strong>MS SQL 1</strong></td>
<td>$t=1.36$</td>
<td>$bhr=97.63%$</td>
<td>$t=2.33$</td>
</tr>
<tr>
<td><strong>Oracle 2</strong></td>
<td>$t=45.9$</td>
<td>$bhr=29.16%$</td>
<td>$t=93.1$</td>
</tr>
<tr>
<td><strong>MS SQL 2</strong></td>
<td>$t=86.1$</td>
<td>$bhr=100%$</td>
<td>$t=172.4$</td>
</tr>
</tbody>
</table>

Source: own study

Figure 1. Chart for Example 1.

**Example 2:** Select measurements with temperature below 10°C and humidity above 95%. It is worth to note that the data filtering can be done very simple in the SQL clause WHERE using XML data type method exist in SQL Server and SQL/XML function XMLExists in Oracle. Below the whole queries are presented due to significant differences in the SQL code.

**SQL Server:**
```
SELECT DATA.query(''
  <pomiar>
    <temperatura>{data(/observation/temp_c)}</temperatura>
    <wilgoc>{data(/observation/relative_humidity)}</wilgoc>
  </pomiar>'
FROM XML.Measurements
WHERE DATA.exist(''
  /observation[temp_c < 10 and relative_humidity > 95]')=1
                  (: SQL Server :) )
```

**Oracle:**
```
SELECT XMLQuery(''
  <pomiar>
    <temperatura>{data(/observation/temp_c)}</temperatura>
    <wilgoc>{data(/observation/relative_humidity)}</wilgoc>
  </pomiar>'
FROM measurements_bin
WHERE XMLExists(''
  /observation[temp_c < "10" and relative_humidity > "95"]'')=1;
                 (: Oracle :) )
```
Table 2. Results of Example 2.

<table>
<thead>
<tr>
<th></th>
<th>400000 records</th>
<th>800000 records</th>
<th>1172370 records</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle</td>
<td>( t=87 ) ( bhr=0.04% )</td>
<td>( t=175 ) ( bhr=2.06% )</td>
<td>( t=259 ) ( bhr=0.9% )</td>
</tr>
<tr>
<td>MS SQL</td>
<td>( t=139 ) ( bhr=100% )</td>
<td>( t=282 ) ( bhr=100% )</td>
<td>( t=419 ) ( bhr=100% )</td>
</tr>
</tbody>
</table>

Source: own study

Figure 2. Chart for Example 2.

Source: own

Example 3: Convert of wind speed indicated in mph to km/h and m/s, if such information exists as a number:

```sql
query('if (data(/observation/wind_mph)!="" and
data(/observation/wind_mph)!="NA")
then <data>
  <wind v="kmh">{data(/observation/wind_mph)[1]*1.609}</wind>
  <wind v="ms">{data(/observation/wind_mph)[1]*0.447}</wind>
</data> else ()')
```

Table 3. Results of Example 3.

<table>
<thead>
<tr>
<th></th>
<th>400000 records</th>
<th>800000 records</th>
<th>1172370 records</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle</td>
<td>( t=317 ) ( bhr=94.44% )</td>
<td>( t=643 ) ( bhr=95.35% )</td>
<td>( t=948 ) ( bhr=95.37% )</td>
</tr>
<tr>
<td>MS SQL</td>
<td>( t=799 ) ( bhr=100% )</td>
<td>( t=1624 ) ( bhr=100% )</td>
<td>( t=2332 ) ( bhr=100% )</td>
</tr>
</tbody>
</table>

Source: own study

Figure 3. Chart for Example 3.

Source: own

Example 4: Select all measurements (time and location) containing a word 'gusting' in description of wind. Full-text search was performed in SQL clause
WHERE based on XQuery function contains used inside exist (SQL Server) and XMLExists (Oracle):

```sql
SELECT DATA.query('data(/observation/location),
data(/observation/observation_time_rfc822)')
FROM XML.measurements WHERE DATA.exist(''
/observation/wind_string/text()[contains(.,"gusting")])=1
   (: SQL Server :

SELECT XMLQuery('data(/observation/location),
data(/observation/observation_time_rfc822)'
   PASSING DATA RETURNING CONTENT)
FROM measurements_bin WHERE
XMLExists('/observation/wind_string[contains(text(),"gusting")]'PASSING DATA)=1;   (: Oracle :)
```

### Table 4. Results of Example 4.

<table>
<thead>
<tr>
<th></th>
<th>400000 records</th>
<th>800000 records</th>
<th>1172370 records</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle</td>
<td>t=113 bhr=88.48%</td>
<td>t=225 bhr=88.33%</td>
<td>t=338 bhr=89.13%</td>
</tr>
<tr>
<td>MS SQL</td>
<td>t=81 bhr≈100%</td>
<td>t=160 bhr=100%</td>
<td>t=245 bhr=100%</td>
</tr>
</tbody>
</table>

Source: own study

### Figure 4. Chart for Example 4.

Example 5: Integrate XML and relational data by the creation of a short description of the station (attribute with identifier of station from SQL column and location of station from XML document):

```sql
query('<Station StationID="{sql:column("Station")}">
   /observation/location/text()
</Station>')           (: SQL Server :)
```

```sql
XMLQuery('for $m in ora:view("MEASUREMENTS_BIN")/ROW
   return <Station stationID="{$m/STATION/text()}">
   {$m/DATA/observation/location/text()}
</Station>' RETURNING CONTENT)   (: Oracle :)
```
Table 5. Results of Example 5.

<table>
<thead>
<tr>
<th></th>
<th>400000 records</th>
<th>800000 records</th>
<th>1172370 records</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle</td>
<td>( t = 206.6 ) bhr=87.67%</td>
<td>( t = 413.8 ) bhr=87.67%</td>
<td>( t = 611.1 ) bhr=87.73%</td>
</tr>
<tr>
<td>MS SQL</td>
<td>( t = 154.6 ) bhr≈100%</td>
<td>( t = 269.1 ) bhr≈100%</td>
<td>( t = 449.7 ) bhr≈100%</td>
</tr>
</tbody>
</table>

Source: own study

Example 6: Join data from two tables: measurements data from table Measurements and name of station from table Stations.

```
SELECT DATA.query('<Measure>
    <Station>{sql:column("a.station_name")}</Station>
    <Date>{data(/observation/observation_time_rfc822)}</Date>
    <Temp>{data(/observation/temperature_string)}</Temp>
</Measure>')
FROM XML.Stations a, XML.Measurements b
WHERE a.STATION_ID=b.STATION

SELECT XMLQuery(
    'for $m in ora:view("MEASUREMENTS_BIN")/ROW
     for $st in ora:view("STATIONS")/ROW
     where $m/STATION eq $st/STATION_ID
     return <Measure>
     <Station>{$st/STATION_NAME}</Station>
     <Date>{data($m/DATA/observation/observation_time_rfc822)}</Date>
     <Temp>{data($m/DATA/observation/temperature_string)}</Temp>
</Measure>') RETURNING CONTENT
FROM DUAL;

Table 6. Results of Example 6.

<table>
<thead>
<tr>
<th></th>
<th>400000 records</th>
<th>800000 records</th>
<th>1172370 records</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle</td>
<td>( t = 695 ) bhr=97.84%</td>
<td>( t = 1388 ) bhr=97.83%</td>
<td>( t = 2063 ) bhr=97.84%</td>
</tr>
<tr>
<td>MS SQL</td>
<td>( t = 207 ) bhr≈100%</td>
<td>( t = 456 ) bhr≈100%</td>
<td>( t = 676 ) bhr≈100%</td>
</tr>
</tbody>
</table>

Source: own study
5. CONCLUSIONS

We have performed many queries and we note, that CPU time and number of logical reads increase linearly with the number of rows stored in table. Oracle usually operates faster, while SQL Server offers much better \( bhr \) ratios, what is a consequence of significantly fewer physical reads.

XQuery is very solid and simple query language, which allows to create almost any queries to database containing both relational and native XML data in intuitive way. Despite observable differences tested systems offer comparable implementations of language, which unfortunately are not satisfactorily efficient in processing of large XML documents and when the query contains relatively complicated path expressions. In both cases it is possible to improve the efficiency and cope the difficulties or imperfections of environments by suitable modification of XQuery queries and choice of appropriate index (SQL Server) or model of XML data storage (Oracle).

REFERENCES


Abstract: This paper concerns problems related to the wide capabilities of the integration of XML data with relational data in databases management systems. There were discussed issues, which give developers of database software the possibility to adapt the mechanisms of storage and processing simultaneously native and relational data to the specifics of these data. In particular, based on Oracle 11g and Microsoft SQL Server 2008 were shown and compared such features offered for XML data, as XML storage options and possibility of indexing XML data, as well as methods of integration of XML and relational data in queries.

Key words: databases, XML, data integration, index, queries, Oracle, SQL Server.

1. INTRODUCTION

Integration of XML technology with databases emerged from the crawling stage a long time ago and thrives in full, as can be seen among the most well-known both commercial and open source database management systems. Just to mention Oracle, Microsoft SQL Server, IBM DB2 or PostgreSQL, which have a significant built-in support for XML data. It should be noticed that the introduction of advanced handling of native data was not only a consequence of some crucial disadvantages of relational model and the impossibility of efficient mapping of XML data model to the relational one (first of all because of hierarchical structure and implicit document order), but also need to propose some alternative for the object model.
2. STORAGE OF NATIVE XML DATA

There are several reasons, which strongly confirm the sense to use relational or object-relational database system for native handling XML documents. The most important are:

- need to store XML data together with relational data and consequently the necessary SQL support for processing mixed data,
- frequently performed search and modification of the XML data,
- required automatic validity check and optional well-formedness check of the XML data,
- required access to the XML data using ADO.NET library (Microsoft) or OLE DB interface (Oracle),
- the potential need for use of some administrative functionalities such as e.g. backup or replication,
- need to increase efficiency of data processing (e.g. through the use of server clustering technology – Oracle RAC).

The database management systems usually allow several ways of storing XML data in database. They differ in the internal storage, what is important both because of the way of processing and the total volume claimed on tablespace level. Regardless of the work environment, we have the following approaches (specific data types of Oracle and SQL Server were indicated in the brackets):

- dedicated XML data type (XMLType and xml) – the data is stored in the internal representation, what preserves a specific XML content (hierarchy and document order, the values of elements and attributes etc.). However, it may not exist the exact text copy of XML data, because not all information is retained (insignificant white spaces, the order of attributes, namespace prefixes and XML declaration);
- mapping between XML and relational data – the contents of the XML documents should be decomposed into columns of one or more tables. The fidelity of the data at the relational level is then preserved (hierarchical structure) in contrast to the document order. Interesting ways of mapping the XML data model to database schemas can be found in [3] (table-based and object-relational mapping);
- some standard text type ([n]varchar2 and [n]varchar) or binary type ([n]clob and varbinary), which allows to store an exact copy of data. However, not every application requires identical copy and is usually satisfied with incomplete content.

In addition, the various XML data can be stored in a various way, using a combination of the above-mentioned approaches, if necessary.

Choice of the appropriate approach depends on the specific data to be stored in the database. We distinguish:
- document-centric XML data, which are characterized by a complex structure with a noticeable lack of regularity, but with important document order. Such documents contain coarse-grained and often mixed data (e.g. due to the conversion from other formats), such as books, manuals, documentations;
- content-centric XML data, which have a constant and regular structure with the irrelevant document order. Such documents contain fine-grained data, such as personal data, measurements data, stocks.

The table below shows the relevant XML data storage models depending on their type.

<table>
<thead>
<tr>
<th>Table 1. XML data storage models in DBMS</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Unstructured parts</strong></td>
</tr>
<tr>
<td>Content-centric data</td>
</tr>
<tr>
<td>Document-centric data</td>
</tr>
</tbody>
</table>

*Source: own preparation*

A binary XML storage option in Oracle 11g is worth mentioning new. The stored in this way documents are compact, post-parse and can be XML Schema aware. This approach is much more efficient regarding indexing, updating or selecting XML fragments than the previous ones (i.e. as CLOB or data of XML type). Moreover, the data occupy much less disk space, which additionally improves the efficiency of data processing by reducing number of I/O operation performed by database engine.

### 3. INDEXING OF XML DATA

Index is a data structure, that increases the speed of data operations. Appropriate choice of index type in relational database can significantly shorten the access time to the processed information. Due to the large data increases and hardware limitations setting proper indexes can improve the efficiency of data processing. In view of the XML database design, a very important functionality of database servers is the ability to indexing XML data, which is offered both in Oracle and SQL Server. The use of the XML index is particularly desirable, when database contains a relatively large amount of XML data (large XML documents or many rows with XML data) or there is a justified need to query for relatively small data (e.g. using XQuery).

XML Index in Oracle 11g can be used in range queries, aggregations and arithmetic operations performed on content-centric data. It is a logical index, consisting from three components [1]:

- a path index, which indexes the XML tags and identifies its various document fragments,
- a value index, which indexes the XML document values,
- an order index, which indexes the hierarchy of the XML document nodes (parent-child, ancestor-descendant and sibling relations).

XMLIndex can be used in various situations, including:

- to speed access to `SELECT` and `FROM` list data, what is useful for XML fragment extraction,
- in SQL queries containing XQuery functions, but its use is not restricted only to `WHERE` clause,
- when the prior knowledge of the XPath expressions is not required;
- for searches with XPath expressions returning collections (when nodes occur multiple times in results of query).

SQL Server offers two types of indexes, which can speed up queries: one primary and three secondary indexes. A primary index creates B-tree contained all tags, values and path from node to root and primary key of table. The primary index is used during the execution every query with the exception of retrieving of the full XML instance. On a column with an existing primary XML index can be created a secondary XML index, which can improve performance of some queries. Choice of the secondary index type depends on query, which is assisted:

- the `PATH` index is designed to optimize the XQuery queries performed on XML columns and based on paths or on paths with predicates, especially when `exist` method is used in `WHERE` clause;
- the `VALUE` index is useful, when query is value based and/or the path is not precise, e.g. when descending search axis without specifying of full path or with path including a wildcard is performed;
- the `PROPERTY` index facilitates data retrieving from single XML instances based on their some properties, e.g. when one or more scalar values is searched, using the `value` method and the primary key is known.

Comparing with the relational index, the important difference is the necessity to create the XML index on single column and the impossibility of using XML indexes on a computed column or the XML variable.

Both SQL Server and Oracle offer the possibility of creating a full-text indexes on XML type column, which index only the pure data, ignoring the XML tagging. In practice, a combination of XML and full-text index is often applied, which allows to filter of XML data using SQL, especially for document-oriented data.

It is easy to see, that in both considered systems the advantages achieved by use of the XML indexes can be reduced by the weakening performance of DML operations. This means that one of the criteria for applying the XML index is a type of database (indexing is particularly recommended for analytical databases). Equally important drawback is size of the index, which exceeds the size of XML...
data stored in the table in both environments (especially in Oracle, where all values and all XPath locations are indexed). A severity of this phenomenon is highly dependent on the structure of the XML instance. Generally, the relatively small number of tags, even with large amounts of XML data, makes it slightly exceeds the size of data, otherwise the size of index can be even several times larger. Using knowledge of the data to be selected, it is possible to limit size of index. In SQL Server, this is done choosing the appropriate secondary XML index, while in Oracle – by choice of proper subset of XPath expressions (through the exclusion removing those that will never be used or inclusion adding only those paths, which will be used). This procedure is similar as in relational databases, where creation of indexes on all possible combinations of columns of various tables is also deprecated.

4. INTEGRATION OF XML DATA WITH RELATIONAL DATA IN QUERIES

First it is worth to note that regardless of the database management system we have two approaches to the integration of XQuery queries and access relational data:

- generate a static XML document and then include the obtained document in the query as any other document;
- dynamically create an XML document inside the XQuery query.

There are few possibilities of generating XML data in SQL Server:

- construction of the XML instance – both queries embedded in method `FOR XML` offer quite fast operation;
- creation a valid XML document based on many columns (relational and XML) is possible by using `FOR XML` clause; through four modes (RAW, AUTO, EXPLICIT, PATH) and some directives almost any desired structure of the document can be obtained.

Generation relational data based on XML is also possible in two basic ways:

- by `OPENXML` clause in SQL queries – such mechanism requires calling `sp_xml_preparedocument` procedure creating internal pointer to XML document, from which data will be converted to the relational form;
- using two methods of data type `XML`: `value` allows to extract scalar values from the XML instance, while `nodes` allows the mapping of any nodes from the XML instance to rows.

We can also mention possibilities of using XQuery access functions to relational data `sql:column` and `sql:variable` and modifying the XML data by statements of XML DML language (insert, delete and replace value of), which is an extension of XQuery. It follows that integration XML and rela-
tional data is not very difficult in practice, even if there is a need of filtering data of one type by condition based on another type.

**Example 1:** *(Integration of XML and relational data in SQL Server 2008)* The query below shows how to retrieve value from relational column `rel_col` in order to integrate them with XML data obtained from XML column `XML_col`, when both columns are in the same table *(table)*.

```sql
SELECT XML_col.query(''<element attribute="{sql:column("rel_col")}">
  {XPath_expression}
</element>'
) FROM table
```

The function `sql:column` is used to create a value of attribute based on relational value and the `XPath_expression` to construct a structure of created element.

Oracle offers also several interesting possibilities of creation the XML data based on relational data, among the others by:

- SQL/XML publishing functions *(XMLElement, XMLAttributes, XMLForrest, XMLConcat, XMLAgg)*, which allow to construct XML data;
- Oracle-specific SQL functions *(SYS_XMLGEN, SYS_XMLAGG)*, generating documents in canonical XML format;
- `DBMS_XMLGEN` package containing many useful procedures creating XML documents from SQL query results *(as FOR XML clause in SQL Server)*.

A general interface between SQL and XQuery languages was implemented in Oracle through `XMLQUERY` and `XMLTable` functions, coming from SQL/XML standard. These functions allow both to construct XML data from relational form and relational data from XML documents, and also to query relational data as if they were XML documents. `XMLQUERY` is a very powerful function, which enables the processing of data from XML and relational columns simultaneously, but it requires the use of the FLOWR expression. Direct access to XML data is provided through the `ora:view` function, which produces XML views over the relational data, on the fly. In turn, the `XMLTABLE` function gives a possibility to decompose result of XQuery expression into relational form.

**Example 2:** *(Integration of XML and relational data in Oracle 11g)* The query below shows how to retrieve value from relational column `rel_col` in order to integrate them with XML data obtained from XML column `XML_col` when both columns are in the same table *(table)*.
SELECT XMLQuery(''
for $p in ora:view("table")/ROW
return
  <element attribute="{$p/rel_col/text()}">
    {$p/XML_col/XPath_expression}
  </element>
',
RETURNING CONTENT)
FROM table

The function ora:view creates XML view over relational data retrieved from table. Because the function returns an unordered sequence of nodes, we have to iterate over it by the FLOWR expression. In such case, the access to the relational column is possible by her name placed as part of the XPath location path instead of SQL clause PASSING rel_col, what is implied by form of result returned by ora:view. In turn, the XQuery function text() retrieves data as usual string. Obviously, the XPath_expression used inside FLOWR clause return lets to construct a structure of created element based on XML column.

Figure 1. Idea of integration of XML and relational data in XQuery query by using SQL/XML

Source: [2]
One of possibility of integration of XML and relational data (SQL/XML) in database management systems have been illustrated in Figure 1. XQuery allows to join the XML data retrieved from relational column with native XML data. The considerations above imply that Oracle offers many more ways of access to relational data from level of XQuery query than SQL Server.

Important tools from a viewpoint of XML data processing are also XMLExists function (Oracle) and exist XML datatype method (SQL Server). They are used in WHERE clause to limit the query results to data which satisfy some property (function checks if the document contains the node specified by the given XPath expression).

Despite observable differences between implementations of the XML technology in considered environments (first of all the way of use functions to retrieve relational data inside XQuery expression sql:column and ora:view), it seems that the offered capabilities are very similar and the built-in features allow for effective processing and integration of data from various sources.

Figure 2. Architecture of XML-enabled data management system.

Source: [4]
5. CONCLUSIONS

Architecture (Figure 2), which was implemented not only in Oracle and SQL Server, but also among the other in IBM DB2, gives users the choice of the appropriate approach based on the specificity of the stored data and application requirements. Therefore there are no obstacles to efficiently create the valid XML documents based on existing or computed relational data, to load the XML data into database, to shred them into relational tables or to query the XML and relational data, exporting them in almost any form.

The discussed mechanisms offered to the developers of database application enable fast loading and processing of the relational and XML data through traditional database user interface (SQL and other languages), using power of relational processing combined with the possibilities of XQuery language. Functionality offered in the considered database management systems provide the support not only to generate and query XML data (XML documents), semi-structural data (XML fragments), structural (relational) data and non-structural (text) data, but also to integrate these data. Moreover, the integration of data is possible in several ways depending on the method of storage and user needs. At the end, it should be strongly emphasized the fact that achievement of the full functionality of a database application is only possible when the data managing is carried out with use of various languages (XQuery, SQL, XPath) and additional features (e.g. specific functions of DBMS) properly combined.

REFERENCES

DATA COMPLETENESS ESTIMATION IN MANAGEMENT INFORMATION SYSTEMS

Tomasz Ząbkowski
Department of Informatics
Warsaw University of Life Sciences (SGGW)

&

PTC Sp. z o.o.

Filip Pudło
PTC Sp. z o.o.

Abstract: Article presents fundamental definitions related to data and information quality in the context of data completeness. Application of artificial neural network was presented for the purpose of practical experiment that aimed to estimate data completeness in corporate Management Information System.

Key words: data completeness, artificial neural network, information quality, Management Information System.

1. INTRODUCTION

According to multidimensional approach, data quality is described by several criteria like: currency, consistency, completeness, accessibility, timeliness, security, ease of operations, understandability, relevance, reliability and accuracy [Pudło and Ząbkowski 2008]. Such criteria or dimensions meet pragmatic information quality attitude which deals with a “degree of usefulness and value data to support the enterprise processes that enable accomplishing enterprise objectives” [English 1999]. Pragmatic information quality is the level of information utility to describe processes that exist in organization whereas inherent information quality is the level of data precision to describe the facts [English 1999]. For those involved in data quality management it is important to measure above dimensions of quality depending on current objectives, problems and expectation. This article focuses on measuring data completeness dimension. Data completeness can be defined as a level of data range available for users according to their requirements or expectations. Another definition can be found in literature as “percentage of data available in database in reference to expected value” [Jarke et al. 2003]. Expectation becomes a fundamental factor due to practical differentiation of expected data completeness and rather theoretical definition of inherent data com-
pleteness. Definition of data completeness may be considered based on customer related data example. Data completeness could be then taken into consideration in two following categories:

- Quantitative – all customers are available
- Qualitative – all customer attributes are available (fulfillment of dimensions: address, segment, tariff).

Depending on objectives data completeness can be represented by percentage or binary variable.

2. PROBLEM OF DATA COMPLETENESS ESTIMATION IN MIS SYSTEM

Considering Data Warehouse environment data completeness become particular factor. Due to complexity of data loading processes, many heterogenic data sources and complex architecture of adapted technology and also organization itself, data completeness is one of important Data Quality dimensions for Management Information Systems practitioners. Sophisticated ETL (extract - transform - load) architecture and variety of loading levels cause the difficulty in data completeness estimation on end user information level. During ETL process, data is extracted, consolidated, aggregated, and transformed. Data attributes are modified according to reference dictionaries (Master Data Management). That is why data on end user level can not be simply compared with its elementary or original sources. Moreover, the issue is more complex due to accessibility of data sources and too high data quality estimation costs. In practice, there are some simple, alternative data completeness estimation models designed to support estimation mainly based on estimation of data loading processes. Unfortunately such models are not sufficient and clear enough for appropriate estimation. In Data Warehouse environment delay of data loading process is common issue. Reasons like missing or wrong attributes, problems with data source, optimization issues are examples. In most cases delayed data is finally loaded to the system however there is a requirement to estimate level of current data completeness. On the other hand in some cases problems with extracting complete data set are not caused by loading process or technical architecture of data warehouse. Data source problems are also among reasons but unfortunately data consumers responsible for data completeness estimation have minimal perception of those issues.

Models that estimate data completeness are also key matter considering Service Level Agreements (SLA) between information suppliers and their customers. Data completeness is very often one of Key Performance Indicator (KPI) which must be measured to estimate level of service according to Information Technology Infrastructure Library (ITIL).

Management Reporting based on Data Warehouse environment requires data completeness estimation due to time rigor and timeliness of its process organiza-
tion. Completeness level of a given data set for analytical purposes must be estimated to support decision making to start Management Reporting. Such decisions are based on alternative indicators (e.g. ETL process estimation) and experience of information system user. In practice most decision are based on trend analysis as an empirical knowledge. Decision whether particular value is found expected and data set is complete becomes a matter for subjective estimation for Data Steward\(^3\). This specific role has been described in following definitions: “Data Stewardship has, as its main objective, the management of the corporation’s data assets in order to improve their reusability, accessibility and quality” [Imhoff 1997] and “The Data Stewardship function entails the definition of quality data and the enforcement of governance over the business processes to ensure compliance” [Lampa 2002].

As far as quantitative data completeness is relatively easy to estimate (e.g. 30% of customers have ‘UNKNOWN’ segment), qualitative data completeness in particular point in time has usually uncertain value. In practice following questions may occur: was last month sale report based on complete data set and what was the level of its completeness? Therefore, variables suggesting no data loading delays and no technology problems may be helpful but give no certainty. In particular point in time exist considerable uncertainty diminishing in course of time. Pragmatic point of view towards completeness can not be considered without timeliness. Data completeness can be relatively easy estimated in case of previous reporting periods. Real challenge is to estimate completeness of current reporting periods just after their closure. How to use an empirical knowledge is also a problem in case of Data Steward substitution.

There is a need to make a solution proposal which on information end user level will have ability to estimate data completeness in automatic way based on Data Steward experience, fast and without need to compare information with its data sources.

3. DATA CHARACTERISTICS

The data gathered for the purpose of the experiment described the situation in one of the company’s corporate management systems. The MIS system was supported with the data that were imported from the source system. Due to different circumstances (mainly delays) the data in the source system were not always complete and the business owner of the MIS system aimed to deliver daily information whether the data in the system are complete or not. Therefore the problem of data completeness estimation was focused on binary classification. There were 21% of

\(^3\) Data Steward – person responsible for managing the data in a corporation in terms of integrated, consistent definition, structures, calculations, derivations and so on. Claudia Imhoff “Data Stewardship: Process for Achieving Data Integrity”, 1997, TDAN.com
the observations classified as incomplete out of 500. The data represented period of 500 days in time that was investigated to gather the information about the data volume, day of the week and the binary target variable indicating whether the incompleteness was observed. The Data Steward role was crucial for data collection step. To give an idea how the data looked like the data volume observed during 500 days is presented on fig. 1. The data volume variable had minimum value of 17.2 and the maximum value of 30.5. Additional 18 variables were derived from these primary three features and the final table for the analysis counted 21 variables (see tab. 1).

Table 1. Variables taken into analysis.

<table>
<thead>
<tr>
<th>Variable name</th>
<th>Variable description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Incomplete</td>
<td>Binary target variable (1- data incomplete, 0- data complete)</td>
</tr>
<tr>
<td>Data_volume</td>
<td>Observed data volume</td>
</tr>
<tr>
<td>DT</td>
<td>Day of the week</td>
</tr>
<tr>
<td>Lag_inc1</td>
<td>Indicator of data incompleteness observed a day before</td>
</tr>
<tr>
<td>Lag_inc2</td>
<td>Indicator of data incompleteness observed two days before</td>
</tr>
<tr>
<td>Lag_inc3</td>
<td>Indicator of data incompleteness observed three days before</td>
</tr>
<tr>
<td>Lag_inc4</td>
<td>Indicator of data incompleteness observed four days before</td>
</tr>
<tr>
<td>Lag_inc5</td>
<td>Indicator of data incompleteness observed five days before</td>
</tr>
<tr>
<td>Sum_inc2</td>
<td>Counted incompleteness during 2 days time</td>
</tr>
<tr>
<td>Sum_inc3</td>
<td>Counted incompleteness during 3 days time</td>
</tr>
<tr>
<td>Sum_inc5</td>
<td>Counted incompleteness during 5 days time</td>
</tr>
<tr>
<td>Lag_1d</td>
<td>Data volume observed a day before</td>
</tr>
<tr>
<td>Lag_2d</td>
<td>Data volume observed two days before</td>
</tr>
<tr>
<td>Lag_3d</td>
<td>Data volume observed three days before</td>
</tr>
<tr>
<td>Lag_4d</td>
<td>Data volume observed four days before</td>
</tr>
<tr>
<td>Lag_5d</td>
<td>Data volume observed five days before</td>
</tr>
<tr>
<td>Lag_6d</td>
<td>Data volume observed six days before</td>
</tr>
<tr>
<td>Lag_7d</td>
<td>Data volume observed seven days before</td>
</tr>
<tr>
<td>Avg_3d</td>
<td>Average data volume within last 3 days</td>
</tr>
<tr>
<td>Avg_5d</td>
<td>Average data volume within last 5 days</td>
</tr>
<tr>
<td>Avg_7d</td>
<td>Average data volume within last 7 days</td>
</tr>
</tbody>
</table>

Source: Own preparation.
Cross analysis of data volume and incompleteness variables showed that data volume that is lower than 20 indicates incompleteness. On the other side, for data volume that is greater than 26 no incompleteness was observed. Fig. 2 presents data volume bar chart stacked with incomplete variable.

**Figure 1. Data volume observed during 500 days.**

**Figure 2. Data volume stacked with incomplete variable.**
4. APPLICATIONS OF STATISTICAL METHODS TO ESTIMATE DATA COMPLETENESS

For the purpose of the experiment the artificial neural networks were used. Extensive literature on neural networks can be found in [Duch et al. 2000, Tadeusiewicz 1993]. Generally, the neural network is a technique based on the neural structure of the brain and it process records one at a time, and learn itself by comparing classification of the record with the known actual classification of the record. The errors from the initial classification is fed back into the network, and used to modify the networks algorithm the second time around, and so on, for many iterations until it finds the best solution or until the algorithm is stopped.

The available data set was divided into training set and validation set, with 75% and 25% of observations, respectively.

The multilayer perceptron (MLP) models were build in the experiment, starting with the simple network with one hidden layer and two neurons in it, so the structure was MLP 21-2-1. The model obtained 90.4% correctly classified (PCC) cases observed on the validation set. PCC measure was calculated according following formula:

\[ PCC = \frac{n_{00} + n_{11}}{n} \times 100\% , \]

where

- \( n \) – number of observations,
- \( n_{00} \) – number of observations where \( \hat{y}_i = y_i = 0 \),
- \( n_{11} \) – number of observations where \( \hat{y}_i = y_i = 1 \),
- \( \hat{y}_i, y_i \) – stands for predicted and observed values, respectively, for \( i = (1,2) \).

The detailed classification results are presented in table 2.

<table>
<thead>
<tr>
<th>Predicted classification</th>
<th>Observed classification</th>
<th>PCC</th>
<th>PCC All</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>18</td>
<td>8</td>
<td>69.2%</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>95</td>
<td>96.0%</td>
</tr>
</tbody>
</table>

Source: Own calculation.

Next step was to add more neurons and to compare classification error with the error obtained with the primary model. It occurred that additional neuron improved overall classification, since PCC measure reached 93.6%. Table 3 presents classification results of MLP 21-3-1 model.
Table 3. Classification matrix for MLP 21-3-1 model.

<table>
<thead>
<tr>
<th>Observed classification</th>
<th>Predicted classification</th>
<th>PCC</th>
<th>PCC All</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>22</td>
<td>4</td>
<td>84.6%</td>
</tr>
<tr>
<td>0</td>
<td>4</td>
<td>95</td>
<td>96.0%</td>
</tr>
</tbody>
</table>

Source: Own calculation.

The procedure of constructing the neural networks with more neurons was stopped after MLP 21-6-1 model was built. Additional neurons in hidden layer did not improve classification and overtraining was observed. Therefore, the final model for data incompleteness classification was MLP 21-6-1. Table 4 presents its classification results.

Table 4. Classification matrix of MLP 21-6-1 model

<table>
<thead>
<tr>
<th>Observed classification</th>
<th>Predicted classification</th>
<th>PCC</th>
<th>PCC All</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>25</td>
<td>1</td>
<td>96.2%</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>98</td>
<td>99.0%</td>
</tr>
</tbody>
</table>

Source: Own calculation.

The final model gave PCC of 98.4% which meant that only two cases were not correctly classified. The results were found to be satisfactory and as a result the model was introduced to deliver information about the data completeness level.

5. CONCLUSION

The data completeness issue is very popular in commercial and technological environments. Application of neural networks presented in specific context of systems quality assurance confirmed the usefulness in the process of data completeness estimation. Such solution gives fair promise to deliver valuable information about the data status in management information systems. The cost of the application is not substantial, since it involves only the statistical software for the model preparation and the cost of expert (data steward) knowledge utilization during data collection step. Of course, the model and the system should be updated from time to time due to changes that occur on dynamic market. The future research in this area would evolve into linguistic summaries to support communication with information users.
REFERENCES


5. Lampa M. (2002) “What are we doing to help the Data Steward?”, The Data Warehousing Institute
