INTEGRATION OF CONCEPTUAL DATA MODELLING METHODS IN INFORMATION SYSTEM DEVELOPMENT

Ljubica Kazi, M.Sc*
University of Novi Sad, Technical faculty "Mihajlo Pupin" Zrenjanin, Serbia
leremic@tfzr.uns.ac.rs

Zoltan Kazi, Ph.D
University of Novi Sad, Technical faculty "Mihajlo Pupin" Zrenjanin, Serbia
zkazi@tfzr.uns.ac.rs

Biljana Radulovic, Ph.D
University of Novi Sad, Technical faculty "Mihajlo Pupin" Zrenjanin, Serbia
bradulov@tfzr.uns.ac.rs

Ofelia Stanciu
Tibiscus University, Faculty of Economics, Timisoara, Romania
ofelia.stanciu@gmail.com

ABSTRACT

One of the most important phases of information system development include data modelling processes, where conceptual data modelling presents an essential part. There are many approaches and methods formally based or those that give heuristics as guidelines. This paper aims to present integration of models for conceptual data modelling. Special emphasize is given to comparison of methods and integration of data modelling process with other processes in information system development.

Key words: data model, process, development, information system, integration

INTRODUCTION

One of the most important phases of information system development include data modelling processes, where conceptual data modelling presents an essential part. In fact, data modelling cant be separated from information system development, since it is based on results of previous phases and give results to following phases.

There are many approaches and methods formally based or those that give heuristics as guidelines. Some of these approaches are formally presented as algorithms that are basis for automated model generating. Automated systems can process their results only if they get data from projectants, so human role is crucial.

During data modelling process there are different activities and responsibilities of human roles. Most researchers and book authors in the field of database design agree that major responsibility to quality of data model is on database designer. His formal methodology and heuristics knowledge, intellectual skills and experience is crucial.

In this paper, we aim to analyze different methods and approaches for data modelling and to integrate them in order to have complete procedural framework for conceptual data modelling. This framework would improve educational and professional activities in conceptual data modelling.
INFORMATION SYSTEM DEVELOPMENT

Information system (IS) can be defined as a system which assembles, stores, processes and delivers information relevant to an organization or to society, in such a way that the information is accessible and useful to those who wish to use it, including managers, staff, clients and citizens. An information system is a human activity system which may or may not involve the use of computer systems (Avison et al., 2003). Information system is a backbone of every organization and its business.

Automated information system include components:
- software - including program applications, operative system, database management system, database, other applications (such as antivirus, office tools etc.)
- hardware - including personal computers, network equipment, servers, UPS, etc.
- lifeware - human personnel with their knowledge, skills, abilities, motivation etc.
- orgware - organizational forms, procedures, documents that describe business process procedures and constraints etc.

There are many methodology approaches to information system development, but most of them agree about major phases:
1. Business / Client requirements capturing and analysis - knowledge about business processes workflow and data needs ie captured by documentation analysis, questionaires, interviews, on-site monitoring etc. Client requirements are then more precisely captured by specific client who defines restrictions, specific conditions and functionality of solutions, expectations regarding functional and non-functional features.
2. Business process and data flow modelling - functional and information needs specification is formally presented by using business process and data flow diagrams, with data dictionary and semi-structured textual specifications where business processes, data flows and data stores structures are presented.
3. System design - include software specification and data modelling. Software specification include modelling of dynamic aspect of solution, i.e. software architecture and functionality, as well as static aspect - data models.
4. System construction and testing
5. System / module delivery and client feedback

These phases are implemented within the complexity scope of information system according to number of business processes in an organization, so more complex systems require division to subsystems and application of these phases separately for each subsystem. Recent approaches introduce agile application of activities of these phases, so smaller modules are to be implemented iterative and incrementally; client would get results frequently and gets more involved in the process (Agile, 2001).

DATA MODELLING PROCESS

Each data model has three components (Mogin et al, 1996):
- Structural component - entities (objects, events), relations
- Integrity component - business rules as constraints / restrictions - on values of attributes, relations etc.
- Operational component - operations on data (read, write, delete, edit)

The difference between data modelling and data projecting is in the scope of process. While data modelling focus is on database description, data projecting include whole process starting with requirements capturing, business process modelling, logical and physical database modelling as well as database deployment. (Naiburg et. al., 2001)

Multi-level architecture approach to database description gives three mayor categories (Mogin et.al, 2000):
- Conceptual shema - independent from DBMS, presents main concepts from business world
Nine major phases of the process of database projecting, i.e. database design and implementation (Mogin et al., 2000) are:
1. User requirements capturing, analysis and formal specification
2. Conceptual design (conceptual data modelling)
3. Database management system (DBMS) choosing,
4. Implementation design (logical design),
5. Design of internal database schema (physical design),
6. Description of schema and physical structure of database design by DBMS language,
7. Data entry to database,
8. Testing,

The role of conceptual design is presented in Figure 1.

CONCEPTUAL DATA MODELLING METHODS

In this section we will give list of different approaches with brief description of methods applied in conceptual data modelling. We present literature survey of authors from our country as well as foreign authors:
1. Grammar analysis of business domain textual description (Mogin et al., 1996) - basic approach to analysis of informal (not formal, not structured) text that describe knowledge about business
domain of an enterprise. Each noun is a candidate for entity, verb is candidate for relationship between entities. There are several heuristics regarding recognition and including specific forms of elements such as gerund, IS-A hierarchy etc.

2. Direct (complete) modelling vs. sequential (partial) modelling (Mogin et al, 2000) - Direct modelling is one approach where all requirements from all user profiles and all views to database applications needs are collected and gathered as a single requirements list that presents a basis for creation of unique, complex data model. Sequential modelling approach introduce partial modelling for each external schema and integration of external schemas to complete schema for whole database. Sequential modelling is proposed as better solution in case of large-scale systems.

3. Integration of submodels, direct modelling based on text analysis, using design patterns, normalization of relations, transformation of models by direct and reverse engineering (Lazarevic et al, 2003):
   - Integration of submodels - Complete system is divided to subsystems by using functional decomposition by applying SSA (structured system analysis) method for business process and data flow modelling. Each primitive process is assigned a data sub-model, that is designed upon appropriate data flows and data stores that are used for that particular primitive process. By integrating sub-models, we get complete data model, but we must take care of consistency while integrating possible similar entities that could have different names. Using CASE tools would enable data dictionary and consistency validation.
   - Integration of submodels based on XML schema - XML schema structure could be basis for entity and relationship matching and data sub-model construction. By integrating those sub-models, we can get complete data model.
   - Direct modelling based on text analysis - For each use case there is a textual specification on action steps and flow, with exceptions and extension points. This text presents a basis for grammar analysis, where nouns are candidates for entities and attributes, while verbs are candidates for relations between entities.
   - Using design patterns - Patterns are general solution for class of problems. Professional working environment enable using data modelling design patterns that present general meta-model for class of domains, with possible adjustment to particular business domain needs.
   - Normalization of relations - Process of creation of relational conceptual data model, where each relation of the relational model should has appropriate normal form, avoiding anomalies (add, delete, edit data) that could occur in using database implemented according to that model. The main principles that normal forms are based upon are: functional dependency of attributes of relation and decomposition of relations (without loss of information) to other relations. There are 1st, 2nd, 3rd, Boyce-Codd’s, 4th and 5th normal form. Data flows from SSA or XML messages data flows are starting material for analysis - they are not in any normal form and need to be processed for normalization. After analysis and decomposition to relations in normal forms, we proceed to synthesis, by consolidation and merging of relations that has equal identifiers.
   - Transformation of models by direct and reverse engineering - Direct engineering presents creating conceptual models according to needs and business specifications and implementing at particular DBMS. Reverse engineering implemented database bring back to a model representation.

4. Database modelling at each IS development phase (Naibur et al. 2001) - For each of information development phase there are database modelling team to be engaged. This approach include using UML models in each phase, starting from business process modelling, requirements definition, analysis and preliminary system design. Elements from all previously created models are mapped to data model elements, like classess are mapped to entities, relations among classess to relations among entities, etc.

5. Iterative refinement for conceptual modelling, relational model normalization and operative efficiency adjustment of physical model (Elmasri et. al 2007) - Iterative refinement is applied at ER modelling level. "Conceptual design is considered an iterative refinement process, where initial design is created and then iteratively refined until the most suitable design is reached". Refinement of ER conceptual design is based on heuristics guidelines for design choices. Normalization (previously described at 3.) is applied at relational model, while adjustments for
operative efficiency are made at physical model level with implemented databases in specific DBMS. These adjustments include physical database design decisions such as indexing, denormalization to speed up queries, horizontal and vertical partitioning etc.

NEW DATA MODELLING METHODS

In this section, we propose new methods of direct modelling (Table 1).

Table 1. Contribution - proposed set of new methods for conceptual data modelling

<table>
<thead>
<tr>
<th>IS development phase</th>
<th>Material</th>
<th>Method</th>
<th>Result</th>
<th>Activity order</th>
</tr>
</thead>
<tbody>
<tr>
<td>Requirements of Business domain and client</td>
<td>Textual description of main processing object lifecycle</td>
<td>Grammar analysis of nouns and verbs - candidates for entities and relations</td>
<td>1st draft model - only entities</td>
<td>1</td>
</tr>
<tr>
<td>Business process and data flow modelling</td>
<td>SSA model</td>
<td>Creating submodels for each primitive process</td>
<td>Final complete model verification by dividing to submodels and checking: completeness or entities and relations. Each submodel has entities for reading and writing data, since each primitive process has input and output data flows</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Data dictionary - attribute</td>
<td>Each attribute from data dictionary has entity mapping (&quot;belongs to some entity&quot;)</td>
<td>2nd draft model - adding attributes to existing entities (from 1st draft) or deriving more entities from attributes (attribute has to have an entity to belong to)</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Data dictionary - data store</td>
<td>Each data store from SSA model is to be normalized. Using substructure notations to analyze and derive sub-entities from a data store</td>
<td>3rd draft model - transforming each data store to set of entities and appropriate attributes, consolidating with entities from 1st and 2nd draft</td>
<td>3</td>
</tr>
<tr>
<td>Creating conceptual data model</td>
<td>1st, 2nd and 3rd draft data model</td>
<td>Creating a complete model by corrections and adjustments</td>
<td>Final model corrected by: Abstraction, redundancy validation, Adding missing attributes and identifiers, Extraction of general data from specific data</td>
<td>4</td>
</tr>
<tr>
<td>Creating physical model</td>
<td>Final conceptual model</td>
<td>Adding indexes for preserving semantic uniqueness</td>
<td>Physical (relational) model with added indexes</td>
<td>6</td>
</tr>
<tr>
<td>Implementing database</td>
<td>Automatically by using CASE tool</td>
<td>Database file, ready for data entry</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

These methods are used in teaching process of data modelling within subject: Information systems at fourth year of Bachelor study in IT engineering course at University of Novi Sad, Technical faculty "Mihajlo Pupin" Zrenjanin. During teaching and exam period, it has been shown that while students are facing real-world problems, in direct modelling of those large-scale systems they had problems of organizing complex set of draft results and finishing work on time. It was proved that sequential/partial modelling was a better choice.

INTEGRATION OF METHODS FOR CONCEPTUAL DATA MODELLING

In this section we present integration of all previously presented methods and strategies (Table 2).
### Table 2. Integration of methods for conceptual data modelling

<table>
<thead>
<tr>
<th>Strategy</th>
<th>Method</th>
<th>Material</th>
</tr>
</thead>
<tbody>
<tr>
<td>Iterative and increment refinement</td>
<td>Requirements collection</td>
<td>Text that describe business process and client requirements</td>
</tr>
<tr>
<td>At each IS development phase</td>
<td>Business process modelling</td>
<td>Business process models</td>
</tr>
<tr>
<td></td>
<td>System design</td>
<td>UML models of system design</td>
</tr>
<tr>
<td>Direct modelling</td>
<td>Complete model</td>
<td></td>
</tr>
<tr>
<td>Sequential / partial modelling</td>
<td>Integration of submodels for each:</td>
<td>- primitive process from SSA</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>taking material:</td>
<td>- attributes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- data flows (SSA)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- data stores (SSA)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- XML message</td>
</tr>
<tr>
<td></td>
<td>Grammar analysis of text regarding:</td>
<td>Lifecycle of business process</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Using design patterns</td>
<td>Use case specification - action steps</td>
</tr>
<tr>
<td></td>
<td>Normalization</td>
<td>Data stores</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Data flows</td>
</tr>
</tbody>
</table>

### CONCLUSION

In this paper we presented literature survey on conceptual data modelling methods. We proposed new methods that supplement existing methods with detailed approach at some aspects. We also presented integration of all methods in aim to enable continuity of models in information system development.

Integrated model will enable framework for professional and educational purposes in the field of data modelling. This framework consist of strategic decisions, methods and material that is used for each of proposed methods.

Future work would include analysis of decisions that are made at each of presented methods as well as considering automation of applying methods or evaluation of conceptual models regarding appropriate method application.

### REFERENCES

Agile Manifesto (2001), http://agilemanifesto.org/
Lazarevic B., Marjanovic Z., Anicic N. & Babarogic S (2003), Baze podataka, Univerzitet u Beogradu, Fakultet organizacionih nauka, Beograd
Mogin P. & Lukovic I. (1996), Principi baza podataka, Univerzitet u Novom Sadu, Fakultet tehnickih nauka, Novi Sad
Mogin P., Lukovic I. & Govedarica M. (2000), Principi projektovanja baza podataka, Univerzitet u Novom Sadu, Fakultet tehnickih nauka, Novi Sad