Birutė PLIUSKUVIENĖ

ADAPTIVE DATA MODELS IN DESIGN

Summary of Doctoral Dissertation
Technological Sciences, Informatics Engineering (07T)

1498-M

Vilnius LEIDYKLA TECHNIKA 2008

VILNIUS GEDIMINAS TECHNICAL UNIVERSITY

Birutė PLIUSKUVIENĖ

ADAPTIVE DATA MODELS IN DESIGN

Summary of Doctoral Dissertation
Technological Sciences, Informatics Engineering (07T)

Vilnius LEIDYKLA TECHNIKA 2008

Doctoral dissertation was prepared at Vilnius Gediminas Technical University in 2003–2008.
The dissertation is defended as an external work.

Scientific Consultant

Prof Dr Habil Petras Gailutis ADOMĖNAS (Vilnius Gediminas Technical University, Technological Sciences, Informatics Engineering – 07T).

The dissertation is being defended at the Council of Scientific Field of Informatics Engineering at Vilnius Gediminas Technical University:

Chairman

Prof Dr Habil Romualdas BAUŠYS (Vilnius Gediminas Technical University, Technological Sciences, Informatics Engineering – 07T).

Members:

Prof Dr Albertas ČAPLINSKAS (Institute of Mathematics and Informatics, Technological Sciences, Informatics Engineering – 07T),

Prof Dr Habil Vytautas KAMINSKAS (Vytautas Magnus University, Technological Sciences, Informatics Engineering – 07T),

Prof Dr Habil Narimantas Kazimieras PALIULIS (Vilnius Gediminas Technical University, Social Sciences, Management and Administration - 03S),

Prof Dr Habil Rimvydas SIMUTIS (Kaunas University of Technology, Technological Sciences, Informatics Engineering – 07T).

Opponents:

Assoc Prof Dr Dalė DZEMYDIENĖ (Mykolas Romeris University, Physical Sciences, Informatics – 09P),

Prof Dr Habil Genadijus KULVIETIS (Vilnius Gediminas Technical University, Technological Sciences, Informatics Engineering – 07T).

The dissertation will be defended at the public meeting of the Council of Scientific Field of Informatics Engineering in the Senate Hall of Vilnius Gediminas Technical University at 2 p. m. on 18 June 2008.
Address: Saulėtekio al. 11, LT-10223 Vilnius, Lithuania.
Tel.: +370 5 274 4952, +370 5 274 4956; fax +370 5 270 01 12;
e-mail: doktor@adm.vgtu.lt
The summary of the doctoral dissertation was distributed on 16 May 2008.
A copy of the doctoral dissertation is available for review at the Library of Vilnius Gediminas Technical University (Saulėtekio al. 14, LT-10223 Vilnius, Lithuania) and at the Library of Institute of Mathematics and Informatics (Akademijos g. 4, LT-08663 Vilnius, Lithuania).

VILNIAUS GEDIMINO TECHNIKOS UNIVERSITETAS

Birutė PLIUSKUVIENĖ

ADAPTYVŪS DUOMENŲ MODELIAI PROJEKTAVIME

Daktaro disertacijos santrauka
Technologijos mokslai, informatikos inžinerija (07T)

Vilnius LEIDYKLA TECHNIKA 2008

Disertacija rengta 2003–2008 metais Vilniaus Gedimino technikos universitete.
Disertacija ginama eksternu.
Mokslinis konsultantas

prof. habil. dr. Petras Gailutis ADOMĖNAS (Vilniaus Gedimino technikos universitetas, technologijos mokslai, informatikos inžinerija – 07T).

Disertacija ginama Vilniaus Gedimino technikos universiteto Informatikos inžinerijos mokslo krypties taryboje:

Pirmininkas

prof. habil. dr. Romualdas BAUŠYS (Vilniaus Gedimino technikos universitetas, technologijos mokslai, informatikos inžinerija – 07T).

Nariai:

prof. dr. Albertas ČAPLINSKAS (Matematikos ir informatikos institutas, technologijos mokslai, informatikos inžinerija – 07T),

prof. habil. dr. Vytautas KAMINSKAS (Vytauto Didžiojo universitetas, technologijos mokslai, informatikos inžinerija – 07T),

prof. habil. dr. Narimantas Kazimieras PALIULIS (Vilniaus Gedimino technikos universitetas, socialiniai mokslai, vadyba ir administravimas – 03S),

prof. habil. dr. Rimvydas SIMUTIS (Kauno technologijos universitetas, technologijos mokslai, informatikos inžinerija – 07T).

Oponentai:

doc. dr. Dalė DZEMYDIENĖ (Mykolo Romerio universitetas, fiziniai mokslai, informatika – 09P),

prof. habil. dr. Genadijus KULVIETIS (Vilniaus Gedimino technikos universitetas, technologijos mokslai, informatikos inžinerija – 07T).

Disertacija bus ginama viešame Informatikos inžinerijos mokslo krypties tarybos posėdyje 2008 m. birželio 18 d. 14 val. Vilniaus Gedimino technikos universiteto senato posėdžių salėje.
Adresas: Saulėtekio al. 11, LT-10223 Vilnius, Lietuva.
Tel.: (8 5) 274 4952, (8 5) 274 4956; faksas (8 5) 270 0112;
el. paštas doktor@adm.vgtu.lt
Disertacijos santrauka išsiuntinėta 2008 m. gegužės 16 d.
Disertaciją galima peržiūrėti Vilniaus Gedimino technikos universiteto (Saulėtekio al. 14, LT-10223 Vilnius, Lietuva) ir Matematikos ir informatikos instituto (Akademijos g. 4, LT-08663 Vilnius, Lietuva) bibliotekose.
VGTU leidyklos „Technika“ 1498-M mokslo literatūros knyga.

General Characteristic of the Dissertation

Topicality of the problem. While solving problems of applied nature difficulties arise, since as an application domain changes so do the structures and contents of primary data and algorithms for solving problems. If future changes are not planned for, applied problems have to be designed, programmed and included into already functioning systems anew. In the latter case not only many problems arise in lessening the faults of problem solving, but the time and expenses for solving the problem increase as well. Because it is often very difficult or impossible to plan for the appearance of new problems or changes in the existing ones, the software dealing with problems of applied nature has to be adaptive. This can be achieved by forming a program from atomic program modules so that changes would have a minimal impact on program structure.

Therefore, in the dissertation the adaptation problem of the software whose instability is caused by the changes in primary data contents and structure as well as the algorithms for applied problems implementing solutions to problems of applied nature is examined. The solution to the problem is based on the methodology of adapting models for the data expressed as relational sets.

The term of adaptation is widely used, but various authors interpret it rather differently. Generally, adaptation is interpreted as software accommodation to the changing environment of problem domain.

Having analysed foreign scientific literature, it can be claimed that many authors relate the idea of adaptivity to extended object-oriented programs supplemented with adaptive programming. In this dissertation the idea of adaptivity is not directly related to object-oriented software or adaptive programming. Here adaptivity is interpreted as the ability of the technology for data processing design to adjust for solving different applied problems, and that is achieved with the help of adaptive data models. These models enable to use the same data processing principle for solving different problems. The limitation of this technology is the requirement that the data processed be expressed as relational sets (RS). The adaptive data models being created are designed for building structural-relational systems. Many Lithuanian and foreign companies and scientific institutions still use structural methods for building systems that focus on supplying relational data in spite of object-oriented, aspect and other technologies. This is due to the fact that the new technologies are not mature enough. Having this in mind, it is sensible to improve the structural methodology. That is the basis for the topicality of this work.

Aim and tasks of the work. The aim of this work is to create an adaptive technology for designing data processing that would enable to perform data selection, aggregation and processing by changing only the formal expression parameters of the adaptive data models forming this technology.

For achieving the aim of the work the methodology for software design is created that requires to solve these tasks:

1. To analyse methods that guarantee the adaptivity of the software for solving applied problems.
2. To form a methodology for adapting data models to solving applied problems when the data used are expressed as relational sets.
3. To define models for selecting and aggregating data expressed as relational sets or their identification and transformations.
4. To create a model for designing data processing or a set of algorithmic dependencies between data and a set of program modules adequate to the former in such a way that selecting subsets of the module set (or designing an applied problem) and choosing the sequence or order of their use, an applied problem with the following limitations could be solved: the structure of primary data and the structure of the data processing results is a relational set.

Scientific novelty. In writing this dissertation the following new results for the informatics engineering have been achieved:

1. The adaptation theory has been amplified applying it not only to the object-oriented model but also to relational sets.
2. A methodology for adapting data models has been formed for solving applied problems using transformations and component programming as a unified technology.
3. Transformations are classified as unconditional, conditional, conditional continuous, conditional cyclical and arithmetic. That is considered novel in standardising the transformations defined in structural design.

Research methodology. Methods of descriptive and comparative analysis have been used in the work to analyse adaptive methods described in the literature. The method of induction has been used to summarize the material analyzed. Elements of relational algebra, structural methods of system building and induction have been used in creating adaptive data models.

Practical value. When applying the adaptive technology for designing data processing the same principle of data processing is used for solving different problems. In other words, different applied problems, whose primary and result data are expressed as relational sets, can be solved using the same set of program modules implementing algorithmic dependencies. Therefore, if the contents of primary data, their structures or the algorithms for the applied problems being solved change, in a certain problem domain, programming can be avoided or minimised.

The technology created can be widely applied, since it is for the data supplied as relational sets, and the latter are adequate to two-dimensional data tables that are very widespread in practice.

Defended propositions

1. The adaptation methodology for data models expressed as relational sets.
2. The use of subsets of the same set of algorithmic data dependencies for solving different problems or the adaptation of the set for solving a particular applied problem.

The scope of the scientific work. The work consists of the general description, four chapters, general conclusions, the list of references (82 items) and the list of publications. The total volume of the dissertation – 128 pages, 25 figures and 20 tables.

In the **first** chapter the analysis of adaptive methods for extending object-oriented programs is provided. The adaptive methods that achieve adaptivity by creating programmes as loosely coupled collaborating components whose aggregation enables to solve problems of applied nature.

In the **second** chapter the most general features of the adaptive technology being created for designing data processing are presented. The stages for creating and use of the said technology are formulated and presented. The models for selecting and aggregating data or data identification and transformations in this technology are presented. Data identifiers and their possible combinations that guarantee the provision of primary data are described in detail. The layers guaranteeing the completeness of the data selected for solving problems of applied nature are indicated. Possible data transformation types are fully described. Formal implementation expressions for transformations and their examples are defined.

The **third** chapter is dedicated to the adaptive model for designing data processing or algorithmic data dependencies. Using the latter a program for solving an applied problem is formed as a subset of the set of all implementation modules for algorithmic dependencies. Thus, in this chapter the concept and the elements of algorithmic data dependencies are defined. The classification of algorithmic dependencies is presented. Each group of algorithmic dependencies is fully described. Algorithms for implementing algorithmic dependencies are defined and detailed.

In the **fourth** chapter examples of applying the adaptive technology for designing data processing in practice are presented. Three solutions to applied problems are described at the level of algorithmic dependencies.

In the general conclusions the main theoretical and practical results of the work and their significance are formulated.

1. The Analysis of Adaptive Methods

The pioneer of adaptive programming K. J. Lieberherr of Boston Northeastern University claims that object-oriented programs are easier to extend than programs written in non-object style, but they are still insufficiently flexible and difficult to tailor for solving different problems. The main feature of object-oriented programming methods is binding methods with classes or class groups. By explicitly binding each method to a particular class the details of class structure are unnecessarily coded into the program. Thereby programs become inert to expansion and reuse. In other words, contemporary object-oriented programs often have redundant information related to the application, and the potential for their reuse is diminished.

In the works of K. J. Lieberherr, J. Palsberg, P. Kroh and other scientists, adaptive programs are defined as extended object oriented programs for which adaptive programming shifts the responsibility for traversing the structure of many objects from different classes from the programmer to the compiler. The main goal is just to indicate traversal landmarks and operations performed there, and leave the generation of traversal code to the compiler so that "landmark" classes and operations performed could be determined. This abstraction facilitates reuse of programs, since the same unchanged adaptive program suits for solving many similar problems. For example, let's consider the adaptive program *Average* that traverses objects of class *Unit* and calculates the average of the fields *amount* inside them. This program can be compiled for the structure of business classes, converting *Unit* to *Employee* and *amount* to *salary*, for calculating the average of employee salaries. This program can also be compiled converting *Unit* to *Item* and *amount* to *price*. This case calculates the average price of all items in inventory

An adaptive program consists of two parts: traversal specification and wrapper specification. Traversal specification indicated classes whose objects must (or must not) be traversed in certain sequence and object variables that must (or must not) be traversed. Shell specification ties classes to operations that need to be performed when the class object is first reached during traversal or upon exiting them.

Although navigation specification only indicates the names of classes and object variables needed for solving programming problems, factual class structure for which an adaptive program is compiled can have intermediate classes and additional object variables. The compiler automatically generates all code used for traversal or ignores these objects. Likewise, wrapper specifications are needed only for the classes whose objects need unique processing. So the programmer writes important parts of the program, and the compiler fills in the rest.

Summarizing this part it can be claimed that the scientific research to make already quite flexible object-oriented programmes still more widely adaptable for solving different problems of applied nature proves that the problem of adaptability exits. Since the instability of software is caused by changes in primary data structures, their contents and the algorithms for solving applied problems, it is sensible to raise this main task: to implement solutions to applied problems in such a way that solutions could be achieved by transforming data into the primary data for a particular problem and forming the program for processing these data by selecting the modules required out of the already existing set of program modules.

2. The Models for Selecting and Aggregating Data

In this dissertation as opposed to the works of K. J. Lieberherr, J. Palsberg, P. Kroh and other scientists the idea of adaptivity is not directly related to object-oriented programs. The adaptive data models being created are for processing data structures expressed as RS‘s. Since relational sets of data are a widely known and used way of providing data, the adaptive technology for designing data processing composed of adaptive data models can be widely used in practice.

In this work the adaptive models of data expressed as RS‘s are grouped into:

- the data selection model whose basis is formed by RS identifiers;
- the data aggregation model based on transformations;
- the model for designing data processing made of algorithmic data dependencies and program modules implementing them.

The data models defined are adaptive since they can be automated or automatically adjusted for solving a particular applied task. The instability of adaptive data models is caused by changes in the contents of primary data, their structure and the algorithms for solving applied problems.

When using adaptive data models the structuring principle is applied to the programs being designed. In other words, the program being designed is broken into modules that are independent of each other and can be aggregated in various combinations.

The adaptive technology for designing data processing enables to realise solutions to problems of applied nature in three stages. The first stage is composed of models for selecting and aggregating data or identifying and transforming data into a relational set of primary data for an applied problem. The second stage is a model for designing data processing or selecting algorithmic dependencies and program modules implementing them so that a program for solving a particular applied problem be formed. In the third stage data are processed and solution results to a particular applied problem are achieved.

Applying the data selection model the data required for solving a particular problem can be selected out of the total data in the order and in quantity required. Data selection and their required quantity are guaranteed by identification methods. This is performed by selecting from the identifier (ID) RS only the identifiers assigned to the relational data sets needed for solving a particular problem (Fig 1).

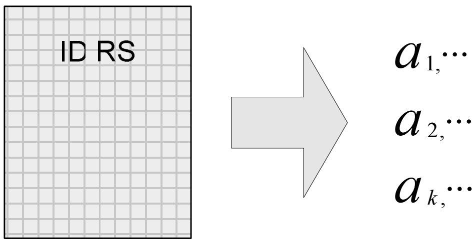

Fig 1. Picking of the required identifiers of relational sets

Using these identifiers the RS's wanted are selected out of the total RS's of primary data and supplied in the order wanted (Fig 2). The provision of RS's depends on the sequence of identifier selection, and that is controlled by the designer taking into account the algorithm for the problem being solved, which determines the order for providing data to be processed.

The data aggregation model with the help of transformations enables to transfer the data selected to a common RS. Transformations not only enable to transfer single attribute values needed for solving a chosen problem out of selected RS's (Fig 3), but also to perform arithmetic and logic operations with needed attribute values. Thus it could be said that with the help of transformations the structure and contents of primary data RS's can be changed.

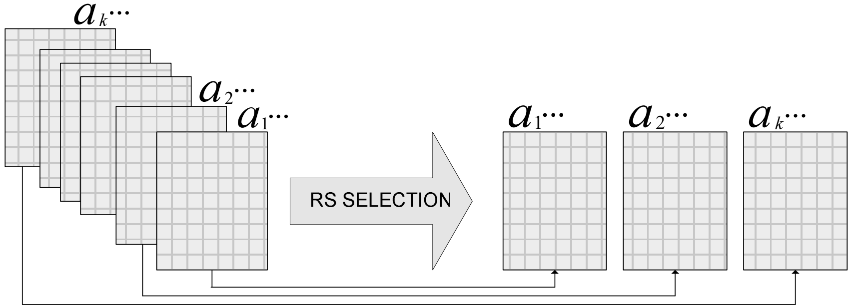

Fig 2. The selection of relational sets out of the totality of primary data

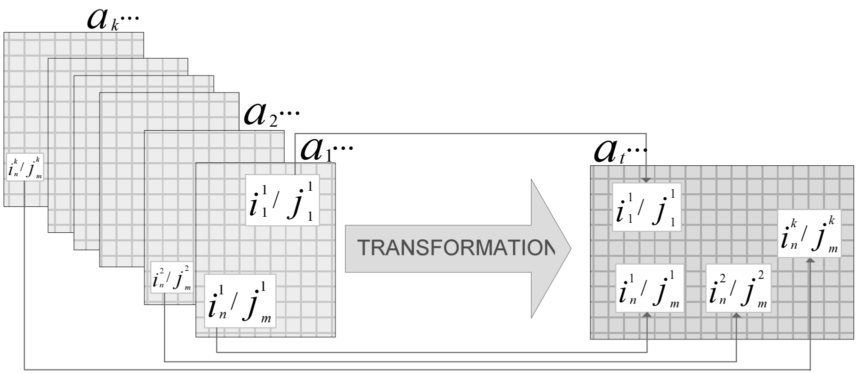

Fig 3. Data transformation

While solving another problem of applied nature, following the same principle, newly formed RS's out of the selected data needed only for solving the latter problem are provided for processing. In other words, adaptive data models enable to supply data by varying their formal expressions.

In this description, only a principle of selecting RS's by one factor, namely the RS schema code, is provided. A detailed description of a real model for selecting RS, in which other identifier features such as a subject code indicating the subject an RS belongs to and a time code indicating the time period or the moment of data validity take part in identifying an RS are provided in the dissertation.

3. The Model for Designing Data Processing

Having selected the relational data sets required for solving a certain problem by applying the data selection model, with the help of transformations we form a new relational set at the level of combinations of new attribute values. Then we solve an applied problem as a particular implementation of algorithmic dependencies between data, since algorithmic dependencies between attributes are defined. Therefore, while solving this problem the attributes are being associated, i. e., they depend on one another. As one of them changes other attributes can either change or not change. Also, the latter algorithmic dependencies may change. For one processing being designed they can be of some nature between the same attributes and for another processing they may need to be chosen in a similar manner. Thereby an accumulated system of program modules implementing algorithmic dependency algorithms is gradually expanded that in the long run can implement solutions to many or even all potential applied problems.

From a theoretical standpoint, the algorithmic data dependency (AD) is considered an algorithm that determines how data are processed while solving the problem required that is inside the same RS. The transformations described in the second chapter of the dissertation are attributed to algorithmic dependencies of exceptional nature, since transformation algorithms most often deal with processing the data from several RS's.

Algorithmic dependencies are organised into the structure adequate to structures of relational sets and can be expressed as a two-dimensional table or a string. Any expression of algorithmic dependencies $< p_{ij} >$ indicates that the value of an attribute has several AD's and have to be placed in a precisely defined order. Thus, $< p_{ij} >$ is an ordered RS of algorithmic dependency codes.

Every AD has a code, its implementation algorithm and a program module for implementing that algorithm. The data in this data processing system are really attribute values c_{ij} of RS. For every attribute value or for any subset of RS values an ordered set of algorithmic dependencies is chosen whose program modules for implementing algorithms of its elements perform data processing:

$$< p_{ij} > \rightarrow < c_{ij} > \rightarrow < c'_{ij} >, \quad (1)$$

where $< p_{ij} >$ is a set of algorithmic dependencies, $< c_{ij} >$ – a set of the primary data selected and $< c'_{ij} >$ – the result of the solution.

The system of data structures and their processing can be depicted with these formal constructions.

Assume a given open set of data algorithmic dependencies:

$$\{ p_1, p_2, \ldots, p_n, \ldots \}. \quad (2)$$

The algorithms of these algorithmic dependences are implemented by a likewise open set of program modules:

$$\{ m_1, m_2, \ldots, m_n, \ldots \}. \quad (3)$$

The openness of both sets means that a need for new *p* and *m* can arise while solving data processing tasks, since their current totality cannot perform certain actions required for solving the problem. Then a new algorithmic dependency p_{n+1} and its implementation module m_{n+1} are created that do not directly perform actions not possible previously, but after supplementing present sets with new algorithmic dependency and program module together with other p_j and m_j ($1 \leq j \leq n$) also encompass the implementation of the algorithm for solving a new problem. We can thus claim that the set of all algorithmic dependency classes and the number of algorithmic dependencies in the class is open to additions.

Identifying relational sets of algorithmic dependency codes a schema code *a* is sufficient, since the latter RS‘s do not depend on the factors of subject or time.

At this stage of the research algorithmic dependencies are divided into five groups:

1. The AD of RS transformations – p^t. All transformation methods described in the second chapter of the dissertation are attributed to this group.

2. The computational-infological AD – p^{ci}. This group is divided into two closely interconnected but separate subgroups. The computational AD subgroup is used to describe various arithmetic operations also getting the results of these operations. The infological AD subgroup is used to compare the results of various arithmetic operations while checking the compatibility and correctness of various processes.

3. The internal AD of attribute values – p^a. This is a group of very varied algorithmic dependencies, rather different from one another, that are relatively permanent. It means that as a particular attribute value is used in solving different problems, that attribute value can differ. Even such algorithmic dependency of an attribute value as an attribute value – number or text – cannot be considered permanent. If we define an attribute value as text, this value could not be used in an arithmetic operation. While solving one problem one may need to consider attribute values to be numbers, and solving another – to be text.

4. The AD of program steps – $p^{\rightarrow}$. An algorithmic RS (ARS) is a set composed of the identifiers for algorithmic dependencies of attribute values that unambiguously determines the use of attribute values in a certain operation of data processing. This is performed by calling the AD code at the ARS and thereby launching the program module implementing the latter. Depending on the course of solving a particular problem, AD codes can be called in a required sequence. This is enabled by the AD‘s of program steps since they indicate the start and direction of a program step. Program steps in an algorithmic relational set are grouped into serial and non-serial.

5. The external AD – p^{e}. The external dependencies of algorithmic RS regulate the interrelation of the ARS of the data being processed with the adjacent ARS. These links cannot be separated from the RS of the data being processed. The external algorithmic dependency is needed so that rather complex tasks could be designed by several designers in parallel speeding up the solution of a complex problem.

Algorithmic data dependencies and program modules implementing them presented in this chapter form the adaptive model for designing data processing. With the help of the latter model using the same program modules implementing algorithmic dependencies and only changing their sequence and quantities different applied problems whose data are expressed as RS‘s can be solved. That indicates the effectiveness of the methods used.

4. Application of Adaptive Technology. Tasks

In this chapter examples of practically applying the adaptive technology for designing data processing are presented. They can be considered one and only one task of data processing. In other words, applying a single principle we can effect solutions to different applied problems whose both primary data and results are expressed as relational sets. This is performed in three main stages. First, a model for data selection is realised that enables to select the RS‘s from a data source needed for solving a particular problem; the RS‘s contain the data (attribute values) for solving the problem. Since RS‘s in their structure and contents are created as data structures for some domain, for solving a particular problem they contain some unnecessary or excess data for a particular application. Thus later in the same stage a data aggregation model is used that with the help of transformations enables to form a single data RS that contains only the attribute values required for solving a particular problem. In the next or second stage a model for designing data processing is applied. Using this model only those AD‘s are selected out of the set of algorithmic dependencies that are required for the algorithm for solving a particular problem. At the same time the program modules implementing the latter AD‘s are also selected; they form the program for solving a problem. So in the last third stage, having implemented the program formed for solving a particular problem of applied nature, the processing of data from a single RS is performed. The design of the latter program is based on the structurisation principle, since it is formed from modules that are independent of each other and can be aggregated in different combinations.

So a solution to a particular problem is effected by the selection parameters for RS identification, RS transformation, algorithmic dependencies and program modules implementing them. Thus it is evident that it is more effective than separate programming for each problem solution.

Another method for applying the adaptive technology for designing data processing is also planned for that is based on essentially the same principle, but it generates the program for solving an applied problem automatically (it is described in more detail in the dissertation).

5. General Conclusions

1. An adaptation methodology has been created that, differing from the approach of many scientific works, is oriented not to the object model but to relational data sets. This narrows the application domain for adaptivity principles but enables to solve practically any applied problems whose data are expressed as RS‘s.
2. A methodology of data model adaptation for solving applied problems has been created that enables to design automatically applied problems and automatically solve them, when data structures and contents satisfy the initial conditions for solving the problem.
3. A concept of algorithmic data dependencies has been created and programmatic algorithmic dependencies have been realised that on the plane of RS‘s enable to supplement a set of algorithmic dependencies with new ones characteristic to the domain in question. New tasks can be solved ~ 2 times more effective, since it is sufficient to select already existing AD’s for data formation and processing and to set the sequence of their application.
4. The principles of adaptivity realised enable to solve different applied problems with the same algorithmic dependencies and program modules implementing them, as well as to implement the main principle of adaptivity – to use subsets of the same sets of AD‘s and program modules implementing them for solving different applied problems.

5. In the dissertation examples of different applied problem are realized at the level algorithmic dependencies that prove the capabilities of the adaptation presented.

6. The research results and capabilities presented in the work are not tied to any particular DBMS or programming language. That enables to expand the domain of applying the adaptation proposed and to exchange data with existing databases, since it is possible to form an algorithmic dependency capable of transforming a relational set into a DB structure.

List of Published Works on the Topic of the Dissertation
In the reviewed scientific periodical publications

1. PLIUSKUVIENĖ, B.; ADOMĖNAS, P. G. Duomenų algoritminių priklausomybių ir jų savybių kaita reliacinėse aibėse. *Lietuvos matematikos rinkinys*, t. 45. Vilnius: Matematikos ir informatikos institutas, 2005, p. 160–166 (in Lithuanian). ISSN 0132-2818 [*MatSciNet, VINITI, Zentralblatt MATH*].
2. PLIUSKUVIENĖ, B.; ADOMĖNAS, P. G. The aggregation of relational sets and the algorithmic dependencies of data. *WSEAS Transactions on Systems*, vol. 5, Iss. 4. Athens: WSEAS, 2006, p. 793–798. ISSN 1109-2777 [*INSPEC, Compendex*].
3. PLIUSKUVIENĖ, B.; ADOMĖNAS, P. G. An Adaptive Technology for Processing Data Relational Sets and its Application. *WSEAS Transactions on Information Science and Applications*, vol. 4, Iss. 4. Athens: WSEAS, 2007, p. 648–654. ISSN 1790-0832 [*INSPEC, Compendex*].

In the other editions

4. PLIUSKUVIENĖ, B.; ADOMĖNAS, P. G. The change of algorithmic data dependencies and their properties in relational sets. In *Proceedings of the 5th WSEAS International Conference on „Artificial Intelligence, Knowledge Engineering and Data Bases“ (AIKED 2006), held in Madrid on 15–17 February, 2006*. Madrid: WSEAS Press, 2006, p. 1–5. ISBN 960-8457-41-6.
5. PLIUSKUVIENĖ, B.; ADOMĖNAS, P. G. The main adaptability principles for the process of handling relational data sets. In *Proceedings of the 6th WSEAS International Conference on „Applied Computer Science“ (ACS ’06), held in Tenerife on 16–18 December, 2006*. Tenerife: WSEAS, 2006, p. 380–384. ISBN 960-8457-57-2.

6. PAKALNIŠKYTĖ, B.; ADOMĖNAS, P. G. Duomenų struktūrų identifikavimo modelis. *„Informacinės technologijos 2005: aktualijos ir perspektyvos“ IV mokslinės praktinės konferencijos pranešimų medžiaga, įvykusios Alytuje gegužės 25 d., 2005*. Alytus: Alytaus kolegija, 2005, p. 157–162 (in Lithuanian). ISBN 9955-9779-0-6.
7. PLIUSKUVIENĖ, B.; ADOMĖNAS, P. G.; SLIESORAITYTĖ, I. Reliacinių aibių agregavimas ir jų apdorojimo programų generavimas. *„Informacinės technologijos 2007“ konferencijos pranešimų medžiaga, įvykusios Kaune sausio 31 d., 2007.* Kaunas: Technologija, 2007, p. 351–355 (in Lithuanian). ISSN 1822-6337.

About the author

Birutė Pliuskuvienė was born in Kaunas on 29 of April 1977.

In 1999 earned a Bachelor‘s of Architecture at the Faculty of Architecture of Vilnius Gediminas Technical University (VGTU). In 2001 earned a Master‘s of Informatics at the Faculty of Fundamental Sciences (FFS) of VGTU. During 2001–2004 worked as an assistant at the Department of Information Systems in the FFS of VGTU. During 2003–2007 studied in a doctoral programme of Informatics Engineering at VGTU. Since 2004 Birutė Pliuskuvienė has been employed as a lecturer at the Department of Information Systems of Vilnius Gediminas Technical University.

ADAPTYVŪS DUOMENŲ MODELIAI PROJEKTAVIME

Mokslo problema. Sprendžiant taikomojo pobūdžio uždavinius, kyla problemų, susijusių su tuo, kad kintant probleminei sričiai keičiasi pirminių duomenų struktūra ir turinys bei sprendžiamų uždavinių algoritmai. Jei iš anksto nenumatomi būsimi pokyčiai, tenka taikomuosius uždavinius projektuoti, programuoti ir įtraukti į veikiančias sistemas iš naujo. Pastaruoju atveju ne tik kyla daug problemų, kai reikia mažinti uždavinio sprendimo sutrikimus, bet ir padidėja sąnaudos. Kadangi numatyti naujų uždavinių atsiradimą ar esamų uždavinių pokyčius dažnai labai sunku arba neįmanoma, todėl taikomojo pobūdžio uždavinius apdorojanti programinė įranga turi būti adaptyvi. Tai galima pasiekti sudarant programą iš atskirų programinių modulių, kad pokyčiai turėtų mažiausią poveikį programos struktūrai.

Taigi disertacijoje nagrinėjama taikomųjų uždavinių sprendimus realizuojančių programinių priemonių, kurių nepastovumą lemia pirminių duomenų turinio, jų struktūrų ir sprendžiamų taikomojo pobūdžio uždavinių algoritmų pokyčiai, adaptavimo problema.

Mokslo problemos aktualumas. *Adaptavimo terminas* vartojamas plačiai, nors įvairių autorių suvokiamas gana skirtingai. Bendrąja prasme adaptavimas suvokiamas kaip programinės įrangos prisitaikymas prie kintančios probleminės srities terpės.

Išanalizavus užsienio autorių mokslinę literatūrą, galima teigti, kad daugelis autorių savo darbuose adaptyvumo sampratą sieja su išplėstinėmis objektinėmis programomis, kurios yra papildytos adaptyviuoju programavimu. Šioje disertacijoje adaptyvumo samprata nėra tiesiogiai sietina su objektinėmis programomis ar adaptyviuoju programavimu. Šiame darbe adaptavimas traktuojamas kaip duomenų apdorojimo projektavimo technologijos gebėjimas prisitaikyti prie skirtingų taikomųjų uždavinių sprendimo ir tai užtikrina adaptyvieji duomenų modeliai. Šie modeliai sudaro sąlygas tą patį duomenų apdorojimo principą taikyti skirtingiems uždaviniams spręsti. Šios technologijos ribojimas yra toks, kad apdorojami duomenys turi būti išreikšti reliacinėmis aibėmis (toliau – RA). Kuriami adaptyvieji duomenų modeliai yra skirti struktūrinėms-reliacinėms sistemoms kurti. Daugelis Lietuvos ir užsienio firmų bei mokslo institucijų, nepaisant objektinių, aspektinių ir kitų technologijų, vis dar taiko struktūrinius sistemų kūrimo metodus, kurie orientuoti į reliacinį duomenų pateikimą. Taip yra todėl, kad naujosios technologijos dar nėra brandžios. Atsižvelgiant į tai, yra prasminga tobulinti struktūrinę metodiką. Tai ir yra šio darbo aktualumo pagrindas.

Disertacijoje apibrėžta adaptyvumo samprata yra analogiška sampratai, kuri pateikiama ir kai kuriuose moksliniuose darbuose, atliktuose Lietuvoje.

Darbo tikslas ir uždaviniai. Šio darbo tikslas – sukurti adaptyviąją duomenų apdorojimo projektavimo technologiją, kuri padėtų duomenis išrinkti, agreguoti ir apdoroti keičiant tik šią technologiją sudarančių adaptyviųjų duomenų modelių formalių išraiškų parametrus.

Darbo tikslui pasiekti kuriama programų projektavimo metodika, kuri reikalauja išspręsti šiuos uždavinius:

1. Išanalizuoti metodus, kurie leidžia užtikrinti taikomiesiems uždaviniams spręsti naudojamų programinių priemonių adaptyvumą.
2. Sudaryti duomenų modelių adaptavimo taikomųjų uždavinių sprendimams metodiką, kai naudojami duomenys yra išreikšti reliacinėmis aibėmis.
3. Apibrėžti duomenų, kurie yra išreikšti reliacinėmis aibėmis, išrinkimo ir agregavimo modelius, t. y. duomenų identifikavimą ir transformacijas.
4. Sukurti duomenų apdorojimo projektavimo modelį, t. y. algoritminių priklausomybių tarp duomenų aibę ir jai adekvačią programinių modulių aibę taip, kad atrenkant tų modulių aibės poaibius (t. y. projektuojant taikomąjį

uždavinį), parenkant jų naudojimo seką ar tvarką, būtų galima išspręsti taikomąjį uždavinį, turintį ribojimus: pirminė duomenų struktūra ir tų duomenų apdorojimo rezultatų struktūra yra reliacinė aibė.

Mokslinis naujumas. Rengiant disertaciją buvo gauti informatikos inžinerijos mokslui nauji rezultatai:

1. Išplėtota adaptavimo teorija, taikant ją ne objektiniam modeliui, o reliacinėms aibėms. Tai praplečia adaptyviojo projektavimo galimybes.
2. Sudaryta duomenų modelių adaptavimo metodika, skirta taikomiesiems uždaviniams spręsti, naudojant transformacijas ir surenkamąjį programavimą kaip bendrą technologiją.
3. Transformacijos suskirstytos į: besąlygines, sąlygines, sąlygines ištisines, sąlygines ciklines ir aritmetines. Tai laikytina naujumu, standartizuojant struktūrinio projektavimo transformacijas.

Praktinė vertė. Naudojant adaptyviąją duomenų apdorojimo projektavimo technologiją skirtingiems uždaviniams spręsti taikomas vienas ir tas pats duomenų apdorojimo principas. Kitaip tariant, naudojant tą pačią, algoritmines priklausomybes realizuojančių programinių modulių aibę, galime spręsti įvairius taikomuosius uždavinius, kurių pirminiai ir rezultatiniai duomenys yra išreikšti reliacinėmis aibėmis. Todėl pakitus pirminių duomenų turiniui, jų struktūroms ar sprendžiamų taikomųjų uždavinių algoritmams tam tikroje probleminėje srityje gali būti visiškai nenaudojami arba minizuojami programavimo darbai.

Sukurta technologija gali būti plačiai taikoma, kadangi skirta duomenims, kurie pateikiami reliacinėmis aibėmis, o jos yra adekvačios labai plačiai praktikoje paplitusioms dvimatėms duomenų lentelėms.

Ginamieji teiginiai

1. Duomenų modelių, išreikštų reliacinėmis aibėmis, adaptavimo metodika.
2. Tos pačios algoritminių duomenų priklausomybių aibės poaibių naudojimas skirtingiems uždaviniams spręsti, t. y. tos aibės adaptavimas konkrečiam taikomajam uždaviniui spręsti.

Darbo apimtis ir turinys. Darbą sudaro bendra darbo charakteristika, 4 skyriai, bendrosios išvados, literatūros sąrašas (82 pozicijos) ir publikacijų sąrašas. Bendra disertacijos apimtis – 128 puslapiai, 25 iliustracijos ir 20 lentelių.

Pirmame disertacijos skyriuje pateikta adaptyviųjų metodų, taikomų objektinėms programoms praplėsti, analizė. Išanalizuoti adaptyvieji metodai, kur adaptyvumas pasiekiamas kuriant programas kaip laisvai susietus besikooperuojančius fragmentus, kurių visuma leidžia išspręsti taikomojo pobūdžio uždavinius.

Antrame skyriuje pateikiami kuriamos adaptyviosios duomenų apdorojimo projektavimo technologijos bendriausieji bruožai. Suformuluoti ir pateikti minėtos technologijos kūrimo ir naudojimo etapai. Pateikti šioje technologijoje naudojami duomenų išrinkimo ir agregavimo modeliai, t. y. duomenų identifikavimas ir transformacijos. Detaliai aprašyti pirminių duomenų išrinkimą užtikrinantys duomenų identifikatoriai ir galimos jų kombinacijos. Išskirti taikomojo pobūdžio uždaviniui išspręsti atrinktų duomenų išsamumą užtikrinantys lygmenys. Išsamiai aprašyti galimi duomenų transformacijų tipai. Apibrėžtos transformacijų formalios realizavimo išraiškos ir jų pavyzdžiai.

Trečias skyrius skirtas adaptyviajam duomenų apdorojimo projektavimo modeliui, t. y. algoritminėms duomenų priklausomybėms. Ja naudojant taikomojo uždavinio sprendimo programa sudaroma kaip visų algoritminių priklausomybių realizacijos modulių aibės poaibis. Tad šiame skyriuje apibrėžta duomenų algoritminių priklausomybių koncepcija ir jų elementai. Pateiktas algoritminių priklausomybių grupavimas. Išsamiai aprašyta kiekviena algoritminių priklausomybių grupė. Apibrėžti ir detalizuoti algoritminių priklausomybių realizavimo algoritmai.

Ketvirtame skyriuje pateikti adaptyviosios duomenų apdorojimo projektavimo technologijos taikymo praktikoje pavyzdžiai, t. y. algoritminių priklausomybių lygmeniu aprašyti trys taikomojo pobūdžio uždavinių sprendimai.

Bendrosiose išvadose suformuluoti pagrindiniai teoriniai ir praktiniai darbo rezultatai bei jų reikšmė.

Darbo rezultatų aprobavimas

Pagrindiniai tyrimo rezultatai paskelbti (2003–2007):

- 3 – tarptautinėse duomenų bazėse referuojamuose periodiniuose mokslo žurnaluose;
- 2 – tarptautinėse duomenų bazėse referuojamoje konferencijų medžiagoje;
- 2 – recenzuojamoje Lietuvos konferencijų medžiagoje.

Bendrosios išvados

1. Sukurta adaptavimo metodika, skirtingai negu daugelyje mokslo darbų, orientuota ne į objektinį modelį, o į reliacines duomenų aibes. Tai

susiaurina adaptyvumo principų pritaikymo sferą, tačiau leidžia praktiškai išspręsti bet kuriuos taikomuosius uždavinius, kurių duomenys išreikšti reliacinėmis aibėmis.

2. Sukurta duomenų modelių adaptavimo taikomiesiems uždaviniams spręsti metodika, kuri leidžia automatizuotai projektuoti taikomuosius uždavinius ir automatiškai juos spręsti, kai duomenų struktūros ir turinys atitinka uždavinio pradines sprendimo sąlygas.

3. Sukurta algoritminių duomenų priklausomybių koncepcija ir realizuotos programinės algoritminės priklausomybės, kurios reliacinių aibių plotmėje leidžia papildyti algoritminių priklausomybių aibę naujomis, tai probleminei sričiai charakteringomis algoritminėmis priklausomybėmis. Nauji uždaviniai gali būti sprendžiami iki 2-jų kartų efektyviau, nes pakanka parinkti jau egzistuojančias duomenų sudarymo bei apdorojimo algoritmines priklausomybes ir nustatyti jų naudojimo eilę.

4. Realizuotieji adaptyvumo principai sudaro galimybes spręsti skirtingus taikomojo pobūdžio uždavinius tomis pačiomis algoritminėmis priklausomybėmis ir jas realizuojančiais programiniais moduliais, ir realizuoti pagrindinį adaptyvumo principą – skirtingiems taikomiesiems uždaviniams spręsti naudoti tų pačių algoritminių priklausomybių ir jas realizuojančių programinių modulių aibių skirtingus poaibius.

5. Disertacijoje realizuoti skirtingų taikomųjų uždavinių pavyzdžiai algoritminių priklausomybių lygmeniu, kurie patvirtina pateiktojo adaptavimo gebėjimus.

6. Darbe pateikto tyrimo rezultatai ir galimybės nesietinos su konkrečia duomenų bazių valdymo sistema ar programavimo kalba. Tai leidžia praplėsti siūlomo adaptyvaus taikymo sferą ir keitimąsi duomenimis su egzistuojančiomis duomenų bazėmis, nes galima sudaryti algoritminę priklausomybę, galinčią reliacinę aibę transformuoti į duomenų bazės struktūrą.

Trumpos žinios apie autorių

Birutė Pliuskuvienė gimė 1977 m. balandžio 29 d. Kaune.

1999 m. įgijo architektūros bakalauro laipsnį Vilniaus Gedimino technikos universiteto Architektūros fakultete. 2001 m. įgijo informatikos mokslo magistro laipsnį Vilniaus Gedimino technikos universiteto Fundamentinių mokslų fakultete. 2001–2004 m. dirbo VGTU FMF Informacinių sistemų katedroje asistente. 2003–2007 m. – Vilniaus Gedimino technikos universiteto informatikos inžinerijos mokslo krypties doktorantė. Nuo 2004 m. iki dabar Birutė Pliuskuvienė dirba lektore Vilniaus Gedimino technikos universiteto Informacinių sistemų katedroje.

Birutė PLIUSKUVIENĖ

ADAPTIVE DATA MODELS IN DESIGN

Summary of Doctoral Dissertation
Technological Sciences, Informatics Engineering (07T)

Birutė Pliuskuvienė

ADAPTYVŪS DUOMENŲ MODELIAI PROJEKTAVIME

Daktaro disertacijos santrauka
Technologijos mokslai, informatikos inžinerija (07T)

2008 05 10. 1,5 apsk. leid. l. Tiražas 100 egz.
Vilniaus Gedimino technikos universiteto
leidykla „Technika“, Saulėtekio al. 11,
LT-10223 Vilnius, *http://leidykla.vgtu.lt*
Spausdino UAB „Baltijos kopija“,
Kareivių g. 13B, LT-09109 Vilnius
www.kopija.lt