TRANSPORT AND TELECOMMUNICATION INSTITUTE

Natalia Petukhova

TEMPORAL DATA MODELS IN INFORMATION SYSTEMS ON RAILWAY TRANSPORT

SUMMARY OF THE DISSERTATION

to obtain the scientific degree of Doctor of Science in Engineering

(Dr.sc.ing.)

Scientific area “Transport”

Scientific research direction “Transport and Logistics”

Scientific supervisor

Dr.habil.sc.ing., professor

Eugene Kopytov

RIGA – 2010

UDK 004.6:656.2

P 314

Transporta un sakaru institūts

Transport and Telecommunication Institute

Petukhova N.

P 314 Temporal Data Models in the Information Systems on the Railway Transport. Summary of the dissertation. Riga: Transport and Telecommunication Institute, 2010. 57 p.

ISBN

THESIS IS PRESENTED AT THE TRANSPORT AND TELECOMMUNICATION INSTITUTE TO OBTAIN THE SCIENTIFIC DEGREE OF DOCTOR OF SCIENCE IN ENGINEERING (Dr.sc.ing.)

OFFICIAL REVIEWER:

Dr.habil.sc.ing., professor Novitsky L.
Riga Technical University, Latvia

Dr.habil.sc.ing., professor Pranyavichus G.
Kaunas Technological University, Lithuania

Dr.sc.ing., professor Misnevs B.
Transport and Telecommunication Institute, Latvia

The defence of the thesis will be held at the special Promotion Council of Transport and Telecommunication Institute on the 30th of March, 2011 at 3:00 p.m.: 1, Lomonosova street, room 4-130, Riga, Latvia, phone (+371) 7100617, fax (+371) 7100535.

CONFIRMATION

I confirm that I have accomplished the dissertation presented to the Transport and Telecommunication Institute to obtain the scientific degree of Doctor of Science in Engineering. The thesis has not ever been presented to any other promotional council to obtain the scientific degree.

____ _ 2010. N. Petukhova

The dissertation is written in English, consists of 4 chapters and comprises 80 figures, 50 formulas, 8 tables and 4 appendices, 164 pages in total. The reference list comprises 134 sources.

ABSTRACT

Natalia Petukhova's doctoral thesis "Temporal Data Models in the Information Systems on the Railway Transport". The scientific supervisor Dr.habil.sc.ing., professor Eugene Kopytov.

The work presents the results of the research dedicated to the problems of using the temporal data in the information systems on the railway transport. The investigation has been pursued during the period from 2001 to 2010. As an object of the research results application the complex of the information systems used in the State Joint Stock Company "Latvian Railway" has been chosen. At the same time the results have a universal character and can be applied to other transport companies. The specificity of the temporal information systems on the railway transport has been investigated in the work under consideration; the main tasks of the information systems, connected with the temporal data accounting and processing on the Latvian Railway, have been found out; the problems, originated in the development and implementation of these systems, have been determined.

The dissertation takes into consideration the problems of the temporal objects presentation in the databases, as well as the problems of provision of their integrity and the questions of interaction with them. The peculiarities of the temporal database principles implementation within the framework of the relation data model have been investigated; the temporal data model with the object abstract identifier has been offered. The opportunity of its compatibility with the existing information systems has been examined. The new temporal form Historical-Overlapped is introduced; this form takes into account the special mode of the temporal object versions switching, which is intrinsic for some processes on the railway transport. The investigation of the task of the temporal object active version determination has been done. The work presents two alternative methods of this task solution: the logical rules method and the temporal elements method. The method of the access controlling the relation system data on the level of the relation tuples has been worked out, and it tackles the issue of the access control to the different time periods temporal data. The method of the interval data transformation is presented. This method is to solve the problem of the complicated computation in the process of the analytical queries executing. The work analyzes the issues of taking into account the temporal constituent in the procedures of the data warehouses management.

There are some examples of the obtained results implementation in the practical tasks solution. The research results are applied in the information systems, operated on the Latvian Railway.

CONTENTS

1. ACTUALITY OF THE PROBLEM

The majority of the processes on transport are connected with time. Timing is a fundamental for the tasks of planning and forecasting of the transport enterprise activity, as well as for practically all analytical and operative missions, demanding the usage of information systems data measured in different moments of time. Partly these data, called the *temporal data*, are active only on the definite time span; they are connected with the definite dates (time marks) or time slots. The accurate proper organization of temporal data in databases and the access facility to them to a large extent determine the functioning efficiency of the transport enterprise information systems.

Each transportation mode has some additional specific peculiarities, which should be taken into consideration in the process of generating of the information system using the temporal data. In particular, the railway transport is determined by inelastic schedule, which is attributable to many factors: transportation system physical limitation, the occurrence of many changes in schedule, the presence of numerous waypoints between the main points on the route, interconnection between the schedules of passenger trains and freight trains. Certainly, the design of the databases, capable of characterizing the railway transport system condition at any moment of history, is exclusively complicated and science-intensive task. As a result, despite the continuing development of the railway information systems complex, the systems users quite often undergo serious hardship in receiving the necessary information, characterizing the saved object condition in different moments of time. As a rule, the information systems databases hold only indicators' sampled values, and on frequent occasions only indicators' averaged on time basis values. In this respect, it is necessary to restore the data from different sources, to perform approximation and extrapolation; these activities bring much allowance, averages and discrepancy in the employed data set.

According to the author's view, the situation is conditioned by the fact, that the information systems continue using the technology of classical relation databases; and these databases are not directed on the storage of all changes in all databases objects. In other words, these technologies do not support the multi-version of these objects. The undertaken study of the operative and still developing information systems on the railway has shown that these systems use the temporal principles inefficiently. The following drawbacks of the information systems demonstrate this inefficiency:

- imperfection and low descriptive component of the temporal objects models;
- disk space wasteful usage;
- low efficiency of the interaction with the temporal objects;
- low databases expansibility;
- high complexity of the databases development;

- hardship with supporting the data integrity;
- difficulties with organizing the access to the information;
- problems with data providing security and privacy.

Under the condition of using the analytical information systems this list should be amplified by the following shortcomings:

- absence of the apparatus of high-performance temporal data processing, in consequence of which the enormous information potential, hidden in a bulk volume of the cumulative data, is used incompletely;
- imperfection of data preparing processes for analysis; the processes are not directed on the data temporal characteristics and there are hardships, connected with the data accessibility and integrity providing.

For many years the deterrent for the temporal databases technology development has been the immense requirements for the resources necessary for supporting multi-versions of the storage objects. But in view of the rampant development of the computing technologies this problem receded into the background. However, there are some other factors constraining the temporal databases usage in such big enterprises as the railway. Among them are the following: the absence of the unitary standard for the temporal data processing; the peculiarities of the temporal objects of the enterprise systems; the lack of necessary skills for the work with the temporal data, the majority of developers suffer of; step up requirements for the data integrity, accessibility and privacy.

The listed issues give evidence of the urgency of papers, directed on the efficiency enhancement of the temporal data usage in the railway transport information systems, which is a basis for the investigation under consideration.

2. DEGREE OF THE THEME STUDIES

Management of the information systems temporal data on the railway transport is a complex task, the investigation of which is carried out in the following directions:

1. Information systems development and functioning on the railway transport;
2. Temporal information systems and databases designing: standards, methods, models, approaches;
3. Temporal data operative and analytical processing;
4. Temporal data information security providing.

In the area of the *information systems designing and functioning* on the railway transport the works of Russian scientists E. Letsky, V. Pankratoff, V. Sharoff, Y. Mukhopad and some other researches can be mentioned. Their investigations cover the railway information systems issues in different countries, demonstrate the complexity and peculiarities of the railway functioning and its information systems development, and put forward some functioning models on the macro- and micro-levels. The authors show the importance of timing factor accounting, but at the same

time do not provide the peculiar solutions for the issues, having in view mostly the classical approaches.

Among the investigations implemented to the information systems on the Latvian Railway, the promotional paper by V. Demidov (year 2006) should be mentioned specially. The research is oriented on the increase of the efficiency of the decision making support system on the railways by the way of perfecting the structure and the functioning processes of the information providing system. The paper suggests the models describing the topology of the transport net of the railway and the passengers' transportation processes. It also presents the developed models and methods of data representation and transformation in the information analysis systems under consideration, including the data virtual stochastic models. The work presents the method of the hierarchically connected view, the conceptual models of the railway transportation prognoses, etc. If to speak about investigating the problem of the temporal data employment, it is important to note the solution of the whole range of the issues connected with accumulating and saving the history of changing the objects states, as well as the tasks of providing the access facilities to them at any moment of time. The development of the fundamentals of the temporal databases designing and implementation of the temporal logics is also worth mentioning (V. Demidov and the author of the thesis under consideration cooperated in these problems solving process [3, 5, 12]). At the same time a lot of temporal system problems occurring in the process of the information systems development and operating in transport system, V. Demidov did not solve or even did not consider. First of all, it is necessary to mention the events periodicity management and the special calendar employment, the issues of supporting the temporal data integrity and privacy.

The urgent problems and achievements in the temporal databases area are summarized in the cooperative work by S. Kuznetsoff and B. Kostenko. There are also remarkable fundamental works, devoted to the *temporal technologies*, developed by C. J. Date, H. Darwen, N. A. Lorentzos, C. S. Jensen. Their cooperative investigation "Temporal Data and the Relational Model" covers the problems of storage and interaction with the temporal data within the framework of the relation model. However, the authors mostly concentrate on the general problems of temporal databases generating and Tutorial D language usage, but do not approach the issues, connected with the periodicity consideration and special calendar consideration, which is exclusively important for the transportation information systems, primarily in the questions of traffic scheduling.

One of the most powerful investigations in the sphere of the temporal databases is the set of works by Christian S. Jensen written during the period from 1991 to 1999 in cooperation with the famous scientists from different countries. He worked together with J. Bair, R. Busatto, J. Clifford, C. E. Dyreson, F. Grandi. H. Gregersen, S. Hsu, T. Isakowitz, T. Y. C. Leung, L. Mark, J. Skyt, R. T. Snodgrass, M. D. Soo, A. Steiner, K. Torp. In 2000 this set of works was published in the form of thesis having the volume of more than 1300 pages. The majority of the above mentioned scientists took part in a committee which developed a specification of the temporal

data query language TSQL2 and SQL-3 standard. It is important to study TSQL2 and SQL-3 for developing the methods and models for temporal data operation.

The immense contribution in the temporal databases development of the international organization TimeCenter is worth mentioning. This center deals with the support of temporal applications on the basis of both traditional and up-starting technologies of the database management systems. The leading scientists in the above mentioned sphere work in this center. Among many outstanding papers Paolo Terenziani's researches attract attention, as well as his colleges' works. Their investigations give a special consideration to processing the temporal constraints for the events, reoccurring with the course of time (so called "periodicity"). They investigate the "symbolic" language of high level, used for presenting the periodicity, and user-oriented compared to the mathematical formulae language. The same work considers the temporal relation model, supporting the definite "symbolic" periodicity appointed by the user (for example, "the second Monday of every month"). Determining the temporal annex for the standard relation algebra operators, the additional new operators and functions are introduced. The temporal algebra, which is consistent supplement for classical (not temporal) one, is presented.

The implementation of *the operative and analytical temporal data processing* is found on the researches in the sphere of the business intelligence and decision-support systems, as well as the databases warehousing theory. Such scientists as B. Inmon, R. Kimball, N. Pendse, T. Pedersen made an immense contribution to the formation of the fundamentals of the multidimensional data modeling and the classical approaches to the tracing of any changes in dimensions.

The investigations belonging to the last-mentioned block are exactly connected with the *information security* problems. They cover the issues dedicated to the discretionary and mandatory security models. Several aspects of these models application in the databases systems are reflected in the works of the below mentioned authors: N. Jukic, S.V. Vrbsky, D.E. Denning, S. Jajodia, R. Sandhu, S. Osborn. The general concept of the information security is reflected in the standards "Trusted Computer System Evaluation Criteria", "Information Technology Security Evaluation Criteria", "Common Criteria for Information Technology Security Evaluation".

The level of the above mentioned works must be highly appreciated. However, it is necessary to remark, these papers do not provide the solution of many problems, emerging in the process of using the temporal data in the information systems on the railway. There are many researches devoted to the problems on the railway, and among them some can be mentioned specially. They are dedicated to the narrow issues of the optimization of the procedures connected with the temporal component, for example, the optimization task for the trains schedule algorithm development. But there are no complex works directed to the organization task and the task of using the temporal data, taking into consideration the specificity of the transport processes. According to all said above, it is possible to claim that the issue, covered in the work under consideration, is formulated for solution for the first time.

3. INVESTIGATION GOAL AND TASKS

The **goal** of the presented thesis is the enhancement of the development and functioning of the railway information systems by means of the temporal databases technologies application supported by the detailed elaboration of the problems of the temporal data models designing and employment, as well as the problems of the access to data methods.

In accordance with the set goal, the author of the thesis is to solve the following **tasks**:

1. Investigation of the railway temporal information systems design specificity and revealing the problems requiring the solution.
2. The critical analysis of the investigations, concerning the temporal databases and the temporal information systems design field, choice and substantiation of existing approaches for the practical application in the railway information systems, and the research under consideration trends statement.
3. Temporal data models development and the methods of interaction with the temporal objects, taking into account the specificity of data processing tasks in the information systems on the railway transport.
4. Temporal data security methods development, capable of supporting the exclusive requirements the railway enterprises put forward for the data integrity, privacy and accessibility.
5. Development of the relation facility extension for manipulating the data implementing the temporal data transformation in the analytical systems.
6. Application of the investigation results for solving the practical problems in the information systems of Latvian Railway.

4. METHODOLOGY AND THE METHODS OF INVESTIGATION

The dissertation research is based on the following concepts:

- results of the railway transport processes modeling;
- results of the scientific research projects, performed by the Transport and Telecommunication Institute and the Latvian Railway with the personal participation of the thesis author;
- author’s experience, acquired in the process of participation in different projects, devoted to the information technologies in the field of databases and Web systems, as an expert, architect, and databases and application developer;
- materials of the scientific conferences, with the author’s personal participation;
- technical documentation, scientific and technical literature and periodical publications on the topics covered by the concerned thesis.

In the process of thesis development the following scientific methods were used: the system analysis mathematical methods, Boolean algebra, relational calculus and sets theory.

5. SCIENTIFIC NOVELTY OF THE WORK

The presented work is a thorough integrated study of the railway transport temporal information systems composition and functioning.

The author performs the following scientific investigation results:

1. Principles of the temporal databases design in the relation environment, customized for the railway transport information systems, are offered;
2. Unique brand new models and methods for the work with the temporal objects have been worked out:
   - temporal data model with the object abstract identifier;
   - temporal relation model with the overlapping lifespans Historical-Overlapped;
   - model of the temporal object, involving periodicity and a special calendar;
   - user's temporal environment providing method;
   - methods of the temporal object active version determination: the logical rules method and the temporal elements method;
   - interval data transformation method;
   - method of the access control to the temporal data on the relation tuples level.

6. WORK PRACTICAL VALUE AND ITS IMPLEMENTATION

The work *practical value* lies in the designing and employing the temporal technologies on the railway transport.

The presented results allow:

- to equip the databases of the existing railway transport information systems with the temporal features and provide the temporal upward compatibility, excluding the necessity to upgrade the existing application-dependant software;
- to improve the temporal data representation in the databases and the methods of interaction with them.
- to increase the temporal data integrity, accessibility and privacy;
- to sophisticate the processes of the temporal data transformation and analysis in the warehouses of the railway data.

The following activities were done on the basis of the obtained results and with the author's personal participation:

- interactive information assistance system of the passenger trains schedule on the Latvian Railway has been designed;
- central system of the railway classifiers and codifiers has been improved; this system is employed in various information services on the Latvian Railway;
- variety of information sub-systems of the Latvian Railway has been upgraded; this variety includes the financial and statistical informational analytical system of the passenger transportation, the system of hardware peripherals registration, etc.

7. APROBATION OF THE RESEARCH

The scientific results, acquired in the process of investigation, were reported on the 16 scientific and research and practice conferences in Belgium, Israel, Latvia, Lithuania, and Russia. The list includes the following items:

1. The VI International Conference "**TransBaltica 2001**", Riga, Latvia, June 7-8, 2001;
2. The Research and Practice and Educational and Methodological Conference "**Science and Technology is the Step into the Future**", Transport and Telecommunication Institute, Riga, Latvia, May 2-3, 2002;
3. VII International Conference "**TransBaltica 2002**", Riga, Latvia, June 12-14, 2002;
4. 43rd International Scientific Conference of Riga Technical University. Riga, Latvia, October 10-14, 2002;
5. The International Conference "Reliability and Statistics in Transportation and Communication" **RelStat'02**, Riga, Latvia, October 17-18, 2002;
6. The International Scientific and Technical Conference "Civil Aviation at the Present Stage of Science, Facilities and Society Development", **MSTUCA**, Moscow, April 17-18, 2003;
7. The International Conference "Modelling and Simulation of Business Systems" **MOSIBUS 2003**, Vilnius, Lithuania, May 13-14, 2003;
8. The VI International Conference "Computing Anticipatory Systems" **CASYS '03**, Liege, Belgium, August 11-16, 2003;
9. The International Conference "Reliability and Statistics in Transportation and Communication" **RelStat'03**, Riga, Latvia, October 16-17, 2003;
10. The VI International Baltic Conference "Databases and Information Systems" **DB&IS 2004**, Riga, Latvia, June 6-9, 2004;
11. The International Conference "Reliability and Statistics in Transportation and Communication" **RealStat'05**, Riga, Latvia, October 13-14, 2005;
12. The Research and Practice and Educational and Methodological Conference "**Science and Technology is the Step into the Future**",

Transport and Telecommunication Institute, Riga, Latvia, December 15-16, 2006;

13. The International Conference “Reliability and Statistics in Transportation and Communication” **RealStat’07**, Riga, Latvia, October 24-27, 2007;
14. The International Conference “Modelling of Business, Industrial and Transport Systems” **MBITS’08**, Riga, Latvia, May7-10, 2008;
15. The International Symposium on Stochastic Models in Reliability Engineering, Life Science and Operations Management, **SMRLO’10**, February 8-11, 2010, Beer-Sheva, Israel.
16. The 9th Joint Conference on Knowledge-Based Software Engineering, **JCKBSE'10**, August 25-27, 2010, Kaunas, Lithuania.

The author’s works in the field of the transport information systems enhancement were commended by:

- the award of the "Social and Economic Systems, Model, Simulation, Agents, Decision Support" symposium for the best scientific article “*Principles of Creating Data Warehouses in Decision Support Systems of Railway Transport*”. The sixth international conference "Computing Anticipatory Systems" **CASYS '03**. August 11-16th, 2003, Liege, Belgium;
- the prize after Karlis Irbitis (doctoral scholarship) from Latvian Academy of Science, “Latvijas Gaisa Satiksme” corporation and Latvian educational fund „Education, science and culture” for the work “*Transport Information Systems Security Enhancement*”, 2005.

8. PUBLICATIONS

27 papers [1-27] have been published basing on the results of the researches. The sources include 17 scientific papers and 10 brief outlines of the scientific reports. They consider the problems of the information systems developing on the transport enterprise and the issues connected with the temporal databases design and implementation. The special attention is paid to the methods of huge data amount processing, such as the problems of the temporal data integrity, accessibility and confidentiality providing.

Moreover with the author’s direct participation there were developed two new concepts of the development of scheduling information system on the Latvian Railway [28, 32] and two technical design specifications for the systems: “Electronic Booking in the International Trains” [29] and the “Registration of the Hardware and Software System” [31], as well as the technology of the system “Electronic Booking in the International Trains” [30]

9. STRUCTURE OF THE THESIS

The paper consists of Introduction, 4 chapters, Conclusions and 4 appendices. It includes 80 figures and 8 tables, constituting 164 pages in total. The references list comprises 134 sources.

The introduction considers the topic urgency of the dissertation paper, formulates the goal and the tasks of the research. The novelty and the practical value of the obtained results are shown in combination with the brief executive summary of the work.

Chapter 1 describes the information systems peculiarities and specificity on the railway transport, demonstrates the time and the temporal data role in these systems. The temporal data evolution and their usage in the information systems are investigated. The temporal data employment problems are analyzed. The information systems tasks, connected with the temporal data registration and processing are revealed and classified. The investigation task specification is performed.

Chapter 2 considers the basic terms used in the temporal database theory. The critical review of the software solutions implementing the temporal technologies is carried out. The choice of the platform for the temporal information systems development is substantiated. The chapter reviews the temporal objects presentation in the databases, as well as the problems of providing their data integrity and interaction with them. It also determines the models and methods, intended for the employment in the transport temporal information systems; the classification of these methods and models is adduced. The problems of the selected models and methods implementation are formulated, and the detailed elaboration of the research tasks is introduced.

Chapter 3 considers the fundamentals of the temporal databases development in the relation environment using the proposed data model with the abstract identifier and the temporal logics. A special attention is dedicated to the introduction of the temporal objects, furnished by the periodicity and correlated with a special calendar. The new temporal form – Historical-Overlapped – is defined; this form considers the peculiar temporal object version change mode. The issues of interaction with the temporal data are solved. The method of providing the user's temporal environment is offered. The temporal object active version determination problem is solved. The method of the interval data transformation for analytical queries is suggested. The issues of the temporal data security providing are solved.

Chapter 4 gives the examples of the practical employment of the research results in the Latvian Railway information systems. The chapter presents the application scheme of the generated methods and models in the Interactive Local Train Traffic Timetable on the Latvian Railway. The processes of the data warehouses management are investigated. The special attention is paid to the temporal data transformation procedures. The practical employment of the generated methods in the Financial and Statistical Information Analytical System of the Passenger Transportation is considered.

The *Conclusion* contains the main research results.

10. DESCRIPTION OF THE PRINCIPAL RESULTS OF THE INVESTIGATION

10.1. Models and Methods of the Work with the Temporal Data in the Railway Transport Information Systems

The Latvian Railway employs various information systems developed in different times on the basis of different technologies. All these systems use the temporal data to this or to that extent. Only some temporal data are supported by the systems. The temporal component can either occur in the information systems processes, which interact with the data temporal dimension obviously or implicitly, or in processed data, which are temporal in their nature, even if they do not have a temporal dimension.

The interconnection scheme of the Latvian Railway enterprise information systems with the objects exhibiting temporal characteristics, as well as their affiliation to one of the temporal data types are shown on Fig. 1. The scheme contains 17 different systems: APIKS-2, APNIS, APFIS, APOVS, etc. These systems are functionally divided into 4 applied sectors areas: freight transportation, passenger transportation, Economics and Finance field and infrastructure management area. Furthermore, the reference information system represents the separate group. In the process of investigation within the frameworks of the given dissertation research three different types of the temporal data employed on the railway have been determined: multi-version data, accumulated historical data, and time sequences (see Fig.1). Every component is contemplated in details.

Multi-version data (version-based temporal data) reflect different object states as a set of versions. In the course of time many objects participating in the organization of the railway transport functioning are subjected to changes: varying train routes, non-permanent currency rates, fluctuating railway service rates, volatile energy supply prices, railway network division into lines and areas, etc. The previous states of these objects (versions) are vital for various analytical computations and prognosis tasks solution. To manipulate versions, which are valid in different periods of time, each object state is associated with a particular time period range – its active time. Such object states (versions) may refer not only to the present or past, but also to the future, for example: planned tariff change; schedule for the following season, etc.

The accumulated historical data (historical data) are various operations, papers and other homogeneous data immanently occurring in the system and accumulated within it (ticket receipts, invoices, freight procedures, etc). The data of these types characterize the functioning of the basic railway processes in time. Each issue of these data is inseparably associated with a particular date or time. Exactly this factor of time is the main dimension in the analytical calculations.

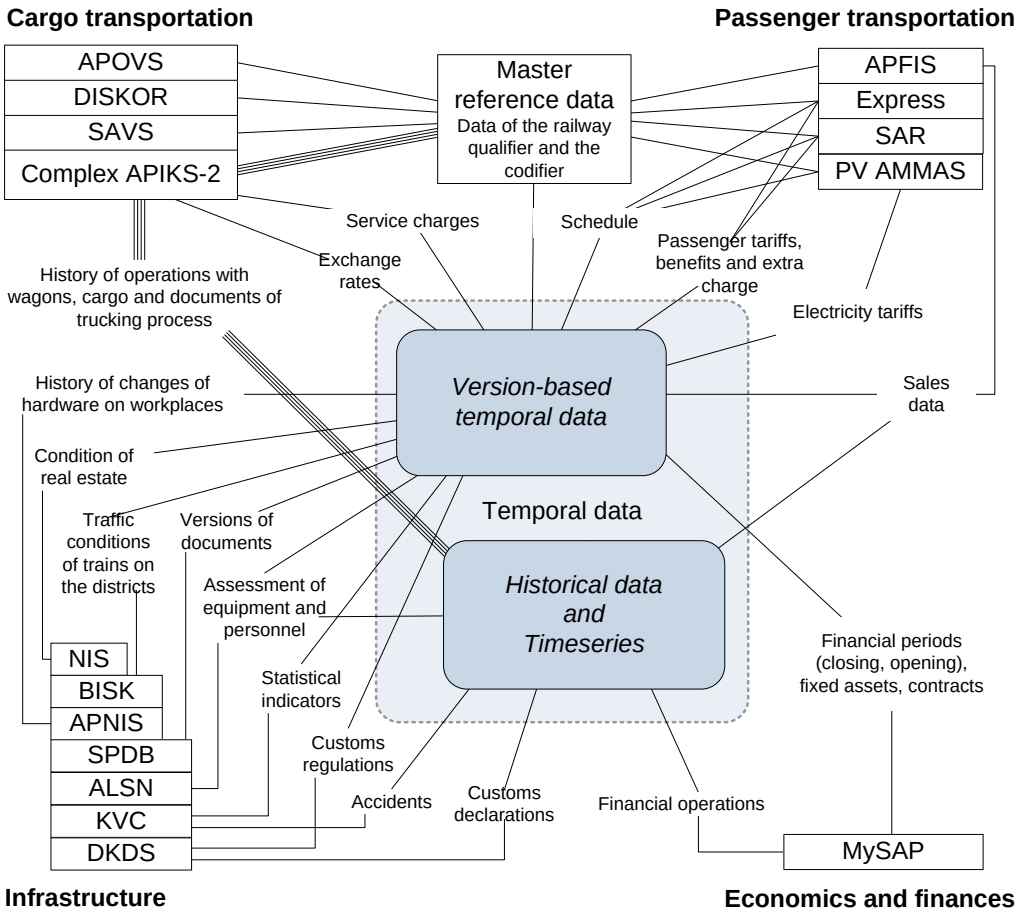


Fig. 1. Temporal data in the Latvian Railway information systems

Time sequences (time-series based temporal data, streaming data) are the data, collected in the process of the object observation. These observations are taken either in succession in time or with the uniform time intervals, or continuously, without intervals (for example, the carriage speed measurement results, their location coordinates). The observations of this type can be arranged chronologically.

There are also the *combined* types of the temporal data. For example, the accumulated data on the passenger tickets sales contain not only the information on the purchases themselves, but also the data on further cases of cancellation of the travels, in other words the cases of multi-version are presented.

The special models and methods are required for the temporal data support in the railway enterprises information systems. These methods and models partially existed formerly, some of them required some approbation and verification in the process of functioning, and several models and methods were developed in the course of the thesis writing. The aggregated body of the models and methods is presented in Fig. 2. The addition and structuring of the variety of the presented facilities have been performed. The analysis of the information systems employed on the railway has demonstrated that the components maintaining the temporal data, are located on different levels: the system level, the logical level and the application layer. Therefore the level is the first feature of structuring. The system level includes databases management system facilities; the logical one comprises the components which allow describing the temporal model; and the application layer incorporates the models and time-dependent/oriented (applied systems, the operation of which is

closely connected with the registration of the data temporal properties, oriented on the temporal data processing, as well as the time-dependent systems) application systems methods, taking into account the data temporal characteristics. The double line on a scheme shows the model groups and methods under investigation in this thesis. The scheme demonstrates the details of the groups requiring the development. A part of the methods (it is especially typical for the system level) are general methods, but some of them are the temporal data-oriented. The known models are indicated with the white colour, and the models generated by the author are shown in green. In its entirety the presented scheme reflects the plan of the research done within the frameworks of the thesis. At the same time the main attention is dedicated to the logical level and in particular to the representation of the information systems temporal objects in the database and the methods of the interaction with these temporal objects. It is an important point to be said that the presented scheme has a generality and it can be used for pursuing the similar researches in other application fields with some specifications and additions.

Every of the above-described and presented levels is observed in details.

System level. Almost none of the modern industrial systems are developed from the green field, although it was naturally an intrinsic practice for the 60-70s. Most of them use the existing data control platforms. These platforms take into account all the necessary methods for physical data arrangement. However, the information about the low-level methods of the data access of the chosen database management system is necessary for the effective temporal data control.

There are five groups of the models and methods suggested for employing on the system level. Such models and methods as the *data storing models*, *the methods of access to the physical storage objects*, and the *methods of increasing the data access speed* are connected with the table data arrangement and organization within the files on disk systems, application of various indexing methods (i.e. including data clustering), parallel data access organization. The similar tasks are solved by the *methods of storing and processing the bulk volume data*, but these methods are large-data-amount operation-oriented. The above mentioned circumstance is especially significant for the temporal systems; consequently this group of methods is shifted into the separate block. *The transactions* play an important part in the temporal model integrity implementation. The issues of the system level model and method employment are considered in the 2-nd chapter of this work. The practical application of them in the information system on transport is presented in the 4-th chapter (paragraph 4.2).

Logical level. The choice of the *temporal data model* (that is a temporal form) is a key point on the level of logical design of the database. The temporal model is considered similarly to the relational one, consisting of three parts: the structural part, the manipulation part and the integral part. The differences between the temporal models in the structural part are expressed in the temporal attribute set; in the integral one – in the temporal integrity limitations, and in the manipulation part – in the algebra of queries to the temporal data. Accordingly, the methods presented on the scheme have their own peculiar implementation for every data temporal model.

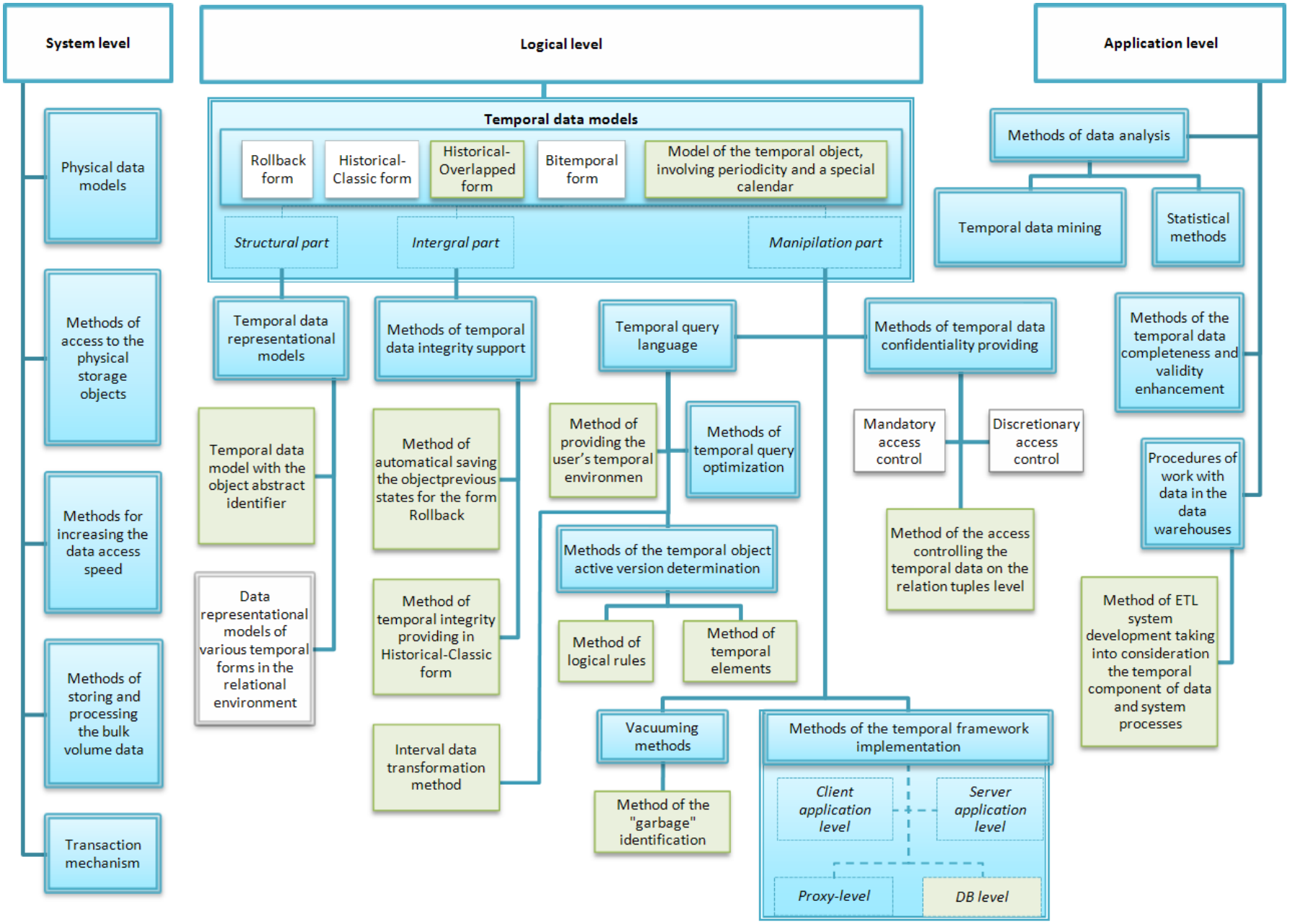


Fig. 2. Models and methods aggregate subjected to the usage in the temporal information systems on the railway transport

The structural part suggests the following pattern: the temporal data model with the object abstract identifier serves as a universal model of the temporal data representation in the relational environment. It allows presenting the temporal object in the form of relation set [3, 5]. Relations, describing the temporal object characteristics, may have various forms: Rollback, Historical-Classic, Bitemporal, Historical-Overlapped. The first-named three forms are widely known, but the fourth form, Historical-Overlapped, is developed by the thesis author and employed in the information systems on the railway transport [11]. The developed form takes into consideration the specific mode of the temporal object version change. This mode is inherent for some processes on the railways. Moreover, there is a variety of objects on the transport, possessing the property of periodicity; these objects are connected with the specific (alternated) calendar. *The temporal object model, possessing the periodicity and the calendar dependency* is suggested for the similar object description [11, 12, 13, 15].

The manipulation part comprises the methods, describing the correct interaction with the temporal data as well as a temporal framework, providing this interaction and taking into account the integral part.

Whereas the specialized temporal database management systems have not been employed in the industrial field for a number of reasons [3, 10], the developers have to generate the logic of the temporal objects manipulation by their own efforts and resources. The relational database management systems do not present at all or present partially the facilities for the temporal data control. In this respect the manipulation and the integral parts of the temporal model in the relational environment require the expansion of the corresponding relational apparatus by the temporal logic. This logic can be implemented on the different levels of the system architecture: on the *level of database*, on the independent transitional *proxy-level*, on the *level of server* or *client (user) application*. Therefore different *methods of the temporal framework implementation* can be used. The paper [10] considers the issues of the level of the temporal framework implementation. Further in view of the database level preferability, the author takes into consideration this level only and suggests the temporal framework methods, considering the peculiarities of the temporal data support in the relational environment in the structural, integral and manipulation aspects. The paper presents the method of the temporal framework implementation as the fundamentals of the temporal databases development in the relational environment. It constitutes the complex approach to the temporal databases in the environment of the relational databases management systems, which can be cooperated with not only from the position of the temporal applications. The temporal framework of the non-temporal database are organized by the aggregate employment shown on the scheme of the logical level methods.

The employment of the *temporal query language* (the special language constructions for the queries to the temporal data), as well as the *methods of the temporal queries optimization* and special techniques of referring to the temporal data allows avoiding the ponderous standard query language offers and increases the efficiency of the queries execution. The solution for many of the above mentioned problems is found in the *method of providing the user's temporal environment*,

developed in the thesis under consideration. This method allows setting the user at the definite point in time and executes the non-temporal queries to the temporal data relatively to this time. Besides this method is the facility of the temporal queries structural optimization.

Every temporal form requires the peculiar solution for the task of the determination of the temporal object version, active for this moment of time. The complexity of the efficient solution of this problem increases simultaneously with the increase of the number of the temporal peculiarities of the temporal form. The paper suggests two alternative methods of the temporal object active version determination for the problem solution: *the logical rules method* and *the temporal elements method* [12, 13, 15].

The majority of the multi-version data are stored in the information systems databases in the interval form. The interval form is improper for many computations from its inception, but their complexity multifold grows if the following aspects occur: the periodicity, the special calendar and the necessity of using the data from several interconnected tables for doing computation. The employment of method, suggested by the author, of the *interval data transformation* can solve the problem of "clumsy" data presentation for computation by turning the interval fact into the time sequence.

The methods of the *temporal data confidentiality* providing comprise the methods which protect the temporal data from unapproved access on reading and changing procedures. The problem of confidentiality providing is observed in chapter 3 within the framework of the solution of the problem of the *temporal data access control*. The known approaches to the access management – the *discretional access control* and the *mandatory access control* – do not cover all the requirements of the delimitative temporal data access strategy. In their traditional version they are not capable of considering the temporal measurement, containing the necessary delimitative information. One of the tasks, belonging to this problem cluster, is observed by the author in her work [9]. This work suggests the way of controlling the users' access to the different time periods data with the *method of the access control to the temporal data on the relation tuples level* [4, 7, 9].

The access to the temporal data is considerably more complicated and more demanding for the computational resources applied. The increase in the temporal data access efficiency is achieved by the way of employing the *method of databases cleaning up* of the out-of-dated unnecessary records (garbage collection, vacuuming). Every temporal form has the corresponding method of the "garbage" identification [10].

The integral part is represented by the *integrity providing methods*, comprising the methods of supporting the temporal normal form, the values integrity of the temporal attributes and other integrity delimitations. The integrity part also incorporates the support of the temporal relation functionality of different forms, exempli gratia, *the method of the automatic storage of the object previous version for the form Rollback* [5].

Despite the fact that the temporal relations are relational at the same time, the known facilities for the relational structures integrity providing are insufficient,

because the objects, incorporated in these systems, are the multi-version objects. In the classical approach, the relation model does not cover this case, but the temporally-oriented methods of the integrity provision are necessary. The issues of the integrity and accessibility providing are considered in the course of analysis and development of the temporal data models in the 3rd chapter of the dissertation.

The application layer. *The methods of the temporal data completeness and validity enhancement* are in demand in both the analytic and the transactional systems. This rank of methods does not need any additional adversaries; the only thing to be highlighted is the fact that the algorithms of the methods take into account the temporal data components. The following cases serve as an example of these methods practical employment: the omitted values regeneration, noise disposal from the data observed.

The temporal data analysis by virtue of any *analysis methods*, including the *statistical methods* and *data mining methods*, contrary to the trivial methods, is extended by the skill of "understanding" the data temporal structure.

The systems of the data bulk volume processing, such as the data warehouses, are inevitably connected with the temporal data. Any models and methods of developing the systems of the data bulk volume processing (for example, the virtual data model in the predicting systems [6, 8], the Passenger transportation reporting system [1, 2]) are to take it into account. The *processes of the data operating in the data warehouses* are connected with the temporal aggregates computation and with the time sequences activities. *The method of ETL (Extract, Transform, Load) system design* demonstrated in chapter 4 allow simplification of the data processing implementation and the temporal data security increase.

Further the methods and models, developed by the thesis author are reviewed in details.

10.2. Temporal Data Model with the Object Abstract Identifier

Taking into consideration both the huge investments in the relation databases, which are the basis for functioning of the majority information systems on the railway and the accumulated practical experience, the author suggests employing the relation environment for the temporal databases implementation with developing the corresponding temporal framework; the issue of transition to the specialized temporal databases management systems is beyond the tasks of the dissertation in review. Solving the problem of employing the relation environment proceeds in two directions: the design of the temporal data models in the relation environment; and development of the temporal logic with the relation facilities application.

Within the frames of working upon the first of the above mentioned directions the thesis author has developed *the temporal data model with the object abstract identifier* [3, 5] (model AOID), allowing viewing any temporal object in the relation database in terms of relation set. The object life section in this model is described with the lifespans of all its characteristics, determined in different relations and having the temporal attributes t_s and t_e. These attributes define respectively the time of lifespans beginning and the time of lifespans ending. The following part

reviews the structural part of the model – the data model of the temporal object in terms of relation.

It is assumed that the object O of the temporal system is characterized by the unique identifier *OID* and the set of properties *A* and can be represented in the terms of:

$$O = (OID, A).$$

The set of the object properties $A = \{a_1, a_2, \ldots, a_m\}$ is divided into two sub-aggregates: the aggregate of the static properties A^s, which are not subjected to the time changes, and the aggregate of the dynamic properties A^d, changing in time, moreover:

$$A = A^s \bigcup A^d, \quad A^s \bigcap A^d = \varnothing, \quad A^s = \{a_1^s, a_2^s, \ldots, a_k^s\}, \quad A^d = \{a_1^d, a_2^d, \ldots, a_n^d\},$$

where a_i^s and a_i^d are the static and dynamic properties respectively; *k* and *n* are the amount of the static and dynamic properties respectively; *m=n+k* are the general amount of the object *O* properties.

As the attribute values in the relation databases have to be atomic, it is not enough to introduce only one relation for presenting the numerous static and dynamic properties. The object *O* is presented as the interconnected relation scheme:

$$O = (R^{OID}, R_1^s, R_2^s, \ldots, R_l^s, R_1^d, R_2^d, \ldots, R_q^d),$$

where $R^{OID} = R^{OID}(OID, A_0^s)$ is the parent relation, describing the object abstract identifier *OID* and comprising the aggregate of the object static atomic attributes A_0^s, $A_0^s \subseteq A^s$; *OID* is the key of the relation R^{OID};

$R_1^s, R_2^s, \ldots, R_l^s$ are the child relations, describing the object A^s static properties ($l \le k$) and presented as the scheme:

$R_i^s = R_i^s(OID, A_i^s), i = 1,2,...,l$, where A_i^s is the subset of the object static properties, logically aggregated in one relation (on the principle of storage optimization, the necessity of the correspondence to the first normal form, information and functional unbundling and other criteria which are taken into consideration by the database developer), $A_i^s \subseteq A^s$. In case the relation R_i^s describes the object atomic property, the *OID* is the primary key, and if there are the multi-valued properties, the key comprises the additional attributes;

$R_1^d, R_2^d \ldots, R_q^d$ are the child relations, describing the discrete ranging in time dynamic attributes A^d ($q \le n$) and presented in terms:

$R_i^d = R_i^d(OID, A_i^d, T_i), i = 1,2,...,q$,

where A_i^d is the sub-aggregate of the discrete dynamic attributes, logically and on the basis of the simultaneous values changing aggregated in one relation, $A_i^d \subseteq A^d$. By employing such attributes unbundling for *q* relations the temporal normal form is

achieved; T_i is the time attributes vector, which can present valid time (VT) or transaction time (TT), or both of them simultaneously, depending on the relation temporal form: $T_i \subseteq T$, where $T = \{t_s^{VT}, t_e^{VT}, t_s^{TT}, t_e^{TT}\}$ is the set of time measurements attributes. In case the relation R_i^d describes the object atomic properties the relation key can be in the form of (OID, t_s^{TT}) or (OID, t_s^{VT}), depending on the temporal form. In the case of the multi-variant properties the key comprises the additional attributes.

The general view of the temporal data model with the abstract identifier, presented in the form of ER scheme, is shown on Fig. 3.

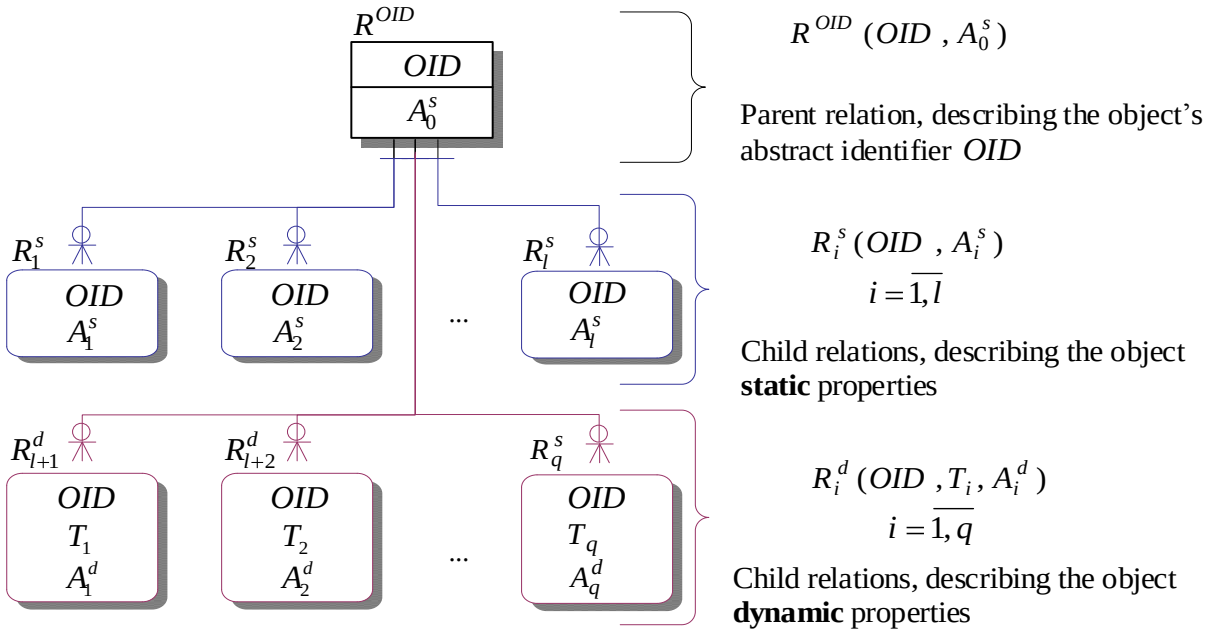


Fig. 3. Data model with the object abstract identifier in the form of ER scheme

10.3. Temporal Relation Model with the Overlapping Lifespans Historical-Overlapped

One of the vital problems occurring in the reviewed tasks of the information systems on the railway is connected with the overlapping periods of the object life cycles. As opposed to the classical notion of the life cycle and the principle of the temporal objects versions change, the object version life cycle might be disaggregated into several lifespans, meaning the possibility for the object version to enter the "shadow zone" and get out of it several times during one control session. The system of schedule accounting considered in the previous author's researches [5, 12, 13, 15] can serve as an example of the system with the listed peculiarities. The temporal object of this system is the train schedule. However, one and the same version might be the active one on different time segments. For example, the principal schedule of the railway traffic might be exchanged by another schedule, adjusted to the remedial operations, for several days. After the end of the repair works the original schedule comes into operation again.

The usage of such known temporal relations forms as Snapshot, Historical, Rollback and Bitemporal (Fig. 4) is not efficient for the description of the temporal objects with the mentioned peculiarities; it is seen not only through the wasteful disk space employment but also through the excessive risk of breaking integrity of the temporal data in case of their changing. Under the condition of the steady continuous increase of the data volumes and the advanced requirements for the data integrity and availability in the transport systems the listed shortcomings become crucial [11].

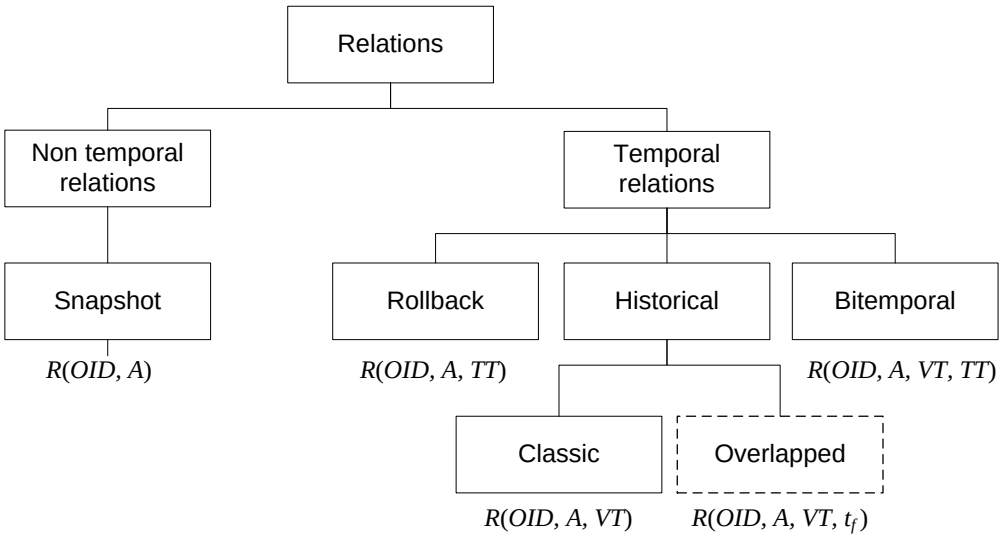


Fig. 4. Relation forms in the temporal databases

The author suggests the special temporal form without the above mentioned drawbacks. It is the relation form with the overlapped lifespans Historical-Overlapped (Fig. 4, the dotted line), considered in the author's works [11, 12, 13, 15]. It comprises the principal features of the classic temporal relations forms, exactly the possibility of existence of the several versions of the same object, becoming the active one consecutively. At the same time the investigated form permits the object versions to become active more than one time.

The temporal relation scheme with the overlapped lifespans Historical-Overlapped is performed in the terms:

$$R(OID, A, VT, t_f),$$

where t_f is the time of the transaction fixation (corresponds to the attribute t_s^{TT}). The transaction fixation time is generated in the moment of entering or changing the record in the database. The active version of the object *OID* is computed on the basis of this attribute values.

The existence of two and bigger number of versions with the overlapping time ranges does not contradict the fact of the relation Historical-Overlapped integrity. It means that the relation *R* can comprise two tuples *r* and *p*, with the following properties:

$$(r.OID = p.OID) \wedge (r.VT \cap p.VT \neq \varnothing).$$

The task of schedule storing of the trains departure from the station can serve as an example of the relation with the overlapped lifespans.

Example 1. The task of the schedule storing of the trains departure from the station.

The schedule of the traffic of the majority trains possesses the prolonged nature but as a result of the various situations there are possible pre-planned and unplanned changes of the departure time. The task is the following one: at any moment of time on any day in the past or in the future to determine the active for this day departure time of the specified train from the station under observation.

Fig. 5 presents the example of the scenario of altering the train schedule. This example is considered in the details. The core fundamental seasonal schedule *(Base)* is assigned for the train number 123P for the summer period from 01.05 to 30.08. Afterwards in the connection with the necessity of conducting the repair works for the period from 03.07 to 15.07 the train schedule was changed (*Repair works*). And on the City Anniversary 06.07 the special timetable was assigned (*City anniversary*).

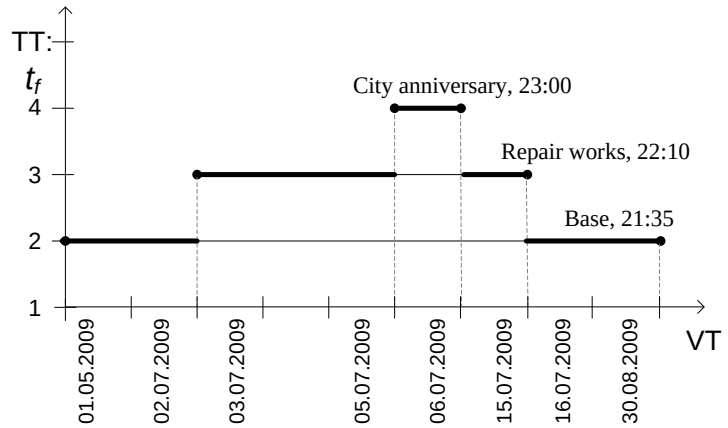


Fig. 5. Example of overlapping of object life-spans in the schedule of the train

Figure 6 demonstrates two ways of performing the information of the above example: in classical relation with non-crossing lifespans Historical-Classic and in the relation with the crossing lifespans Historical-Overlapped. The attributes of the valid time t_s^{VT} and t_e^{VT} define respectively the beginning and the end of the period of the schedule validity; the attribute t_f shows the transaction time.

Historical-Classic

OID Train	Departure time	t_s^{VT}	t_e^{VT}
123P	21:35	01.05.2009	02.07.2009
123P	22:10	03.07.2009	05.07.2009
123P	23:00	06.07.2009	06.07.2009
123P	22:10	07.07.2009	15.07.2009
123P	21:35	16.07.2009	30.08.2009

Historical-Overlapped

OID Train	Departure time	t_s^{VT}	t_e^{VT}	t_f
123P	21:35	01.05.2009	30.08.2009	1
123P	22:10	03.07.2009	15.07.2009	2
123P	23:00	06.07.2009	06.07.2009	3

Fig. 6. Relation "Schedule of the train departure from the station" in the forms Historical-Classic and Historical-Overlapped

This example demonstrates that the relation with the overlapped lifespans is more efficient from the point of view of the amount of tuples. At the same time it

expresses the task semantics more distinctively. The advantage in the semantics delivery is achieved due to the fact that in the relation Historical-Overlapped the tuple presents not only the order of the schedule changes, as the relation Historical-Classic does, but the fact of exchange occurrence as well.

The fundamental of the objects versions control in the relations Historical-Overlapped differs from the control in the relation Historical-Classic. The Historical-Overlapped distinctive peculiarity is the fact that occurring the new object version or changes in the existing one is not preceded by the processing of the temporal attributes values of the neighboring object versions (the previous and/or the following ones). On the contrary, the generation of the new object version or the modification of the existing one in relation Historical-Classic are connected with the proceeding of the boundaries of other versions life segments, in case their own life segments are partially or entirely overlaid as a result of the modification (in the majority of cases it refers to the previous and the following versions of the object).

The task of the identification of the object version which is active for the moment t in the form Historical-Overlapped is connected with the elaborated data access, because it is necessary to choose the most "young" object version using the attribute t_f besides the process of verification the fact that t is comprised in the object version activity range $[t_s^{VT}, t_e^{VT}]$.

The summary results of the comparison relation forms Historical-Classic and Historical-Overlapped are performed in table 1. Two possible variants of the real simulation were taken into account in the course of the comparison: the case when the same object version does not become topical more than one time and the case when the same object version becomes topical on several time segments.

Table 1. *The results of comparison of the relation forms Historical-Classic and Historical-Overlapped*

Situation / **Indicator**	The same object version does not become active on several lifespans: $g = h$		The same object version is active on several time lifespans: $g < h$	
	Historical-Classic	**Historical-Overlapped**	**Historical-Classic**	**Historical-Overlapped**
The size of the temporal relation (the number of tuples)	g	g	$[g + 1; 2g - 1]$	g
Integrity in case of the data alteration (the number of tuple operations when the new object version occurs)	0	0	$[0; u + d + i]$	0
Accessibility in the process of identification of the active object version at the moment t (the number of processed tuples)	{0, 1}	{0, 1}	{0, 1}	$[0; g(t)]$

The table 1 employs the following terms:

g is the number of the object versions;

h is the number the version active periods, $g \le h$;

$u \in \{0, 1, 2\}$ is the number of the tuples, corresponding to the partially overlaid life segments; these u tuples have to be changed;

$d \in \{0, 1, \ldots, h\}$ is the number of the tuples, corresponding to the entirely overlaid life segments; these d tuples have to be eliminated or marked as non-functioning;

$i \in \{0, 1\}$ is the number of the tuples, corresponding to the life sections which overlay the new version life segment; these i tuples have to be added, moreover

$(i = 1) \Rightarrow (d = 0)$; $(d > 0) \Rightarrow (i = 0)$;

$g(t)$ – number of the object versions, which are topical at the moment t.

10.4. Temporal Logic Implementation within the Framework of the Relation Model

The second direction of the task of design the temporal framework is the development of the temporal logic within the framework of the relation model. For its solution the author has suggested the enhancement of the manipulative and integral parts for the relation model. The principal tasks of this enhancement are the following:

- support of the time-based multi-version and temporal integrity providing;
- executing such commands as a removal operation DELETE, the renewal operation UPDATE and the operation of the expanded selection SELECT taking into account the temporal properties of the object;
- support of the temporal upward compatibility with the existing non-temporal applications;
- providing the access to all previous and all following object versions;
- assurance of the existence of the single active version of the object at any moment of time.

There are principal peculiarities of the temporal framework implementation.

For preservation the logic of the legacy applications it is necessary to allow the removal operation DELETE and the operation of the object properties renewal UPDATE. Nevertheless in the process of these operations execution no of the object states must disappear. In this case the following technique of executing the operations of the object modification is offered. At the time moment t_o of the i-th change in the object state the element of the temporal framework turns the non-active version of the object in the shadow zone with the automatic fixation of the time of the life cycle period ending $t_e^{(i)} = t_o$. Thereby the process is completed for the operation DELETE, but in case the operation UPDATE takes place, the temporal framework generates the new relation tuple (the new record) – the active $(i+1)$-th object version with the time of life cycle beginning $t_s^{(i+1)} = t_o$ (Fig. 7). This temporal logic corresponds to the Rollback temporal form.

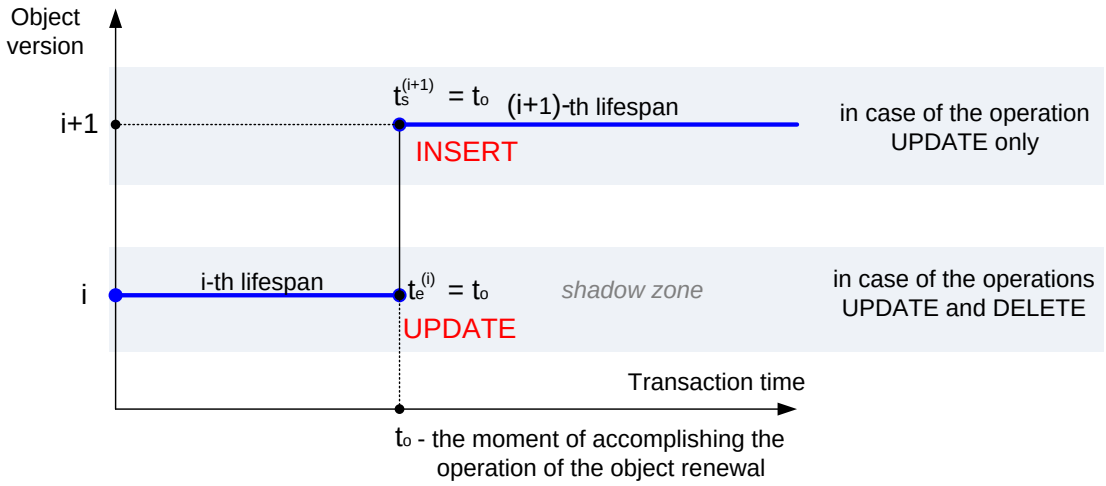


Fig. 7. Diagram of changing the object state in the temporal form Rollback

The scheme of work implementing the above described logic in the process of the operation UPDATE executing is presented in Fig.8. The standard facilities of the active databases are used for this implementation: SQL, views and triggers.

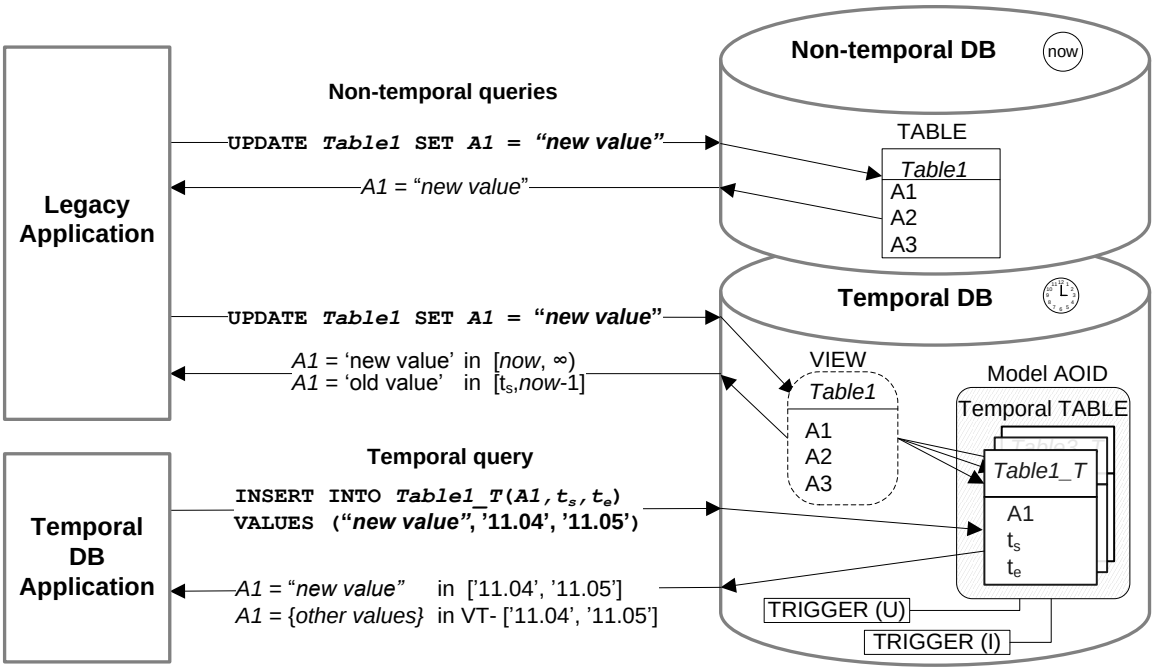


Fig. 8. Support of the queries of the temporal data update

In the process of adding the temporal properties to the information systems objects, the serious structural changes occur in the databases, starting with the primary key repertory and to the extent of the relation decomposition. These changes are bound to affect the logic of applications functioning. The support of the existing logic for the operational (legacy) applications is guaranteed by keeping the interface of their cooperation with the databases.

For solving the task of saving the "old" interface in the databases extended by the temporal properties, it is offered to organize the view of the data, possessing the structure and the components identifiers of the original database before the structural changes. In this case all the components of the database that were

subjected to the structural alterations have to be given the new names. Fig. 8 demonstrates the example, showing the process of transformation under the condition of the temporal extension of the table *Table1* into the set of tables projected in accordance with the model AOID: *Table1_T0, Table1_T1*… and so on depending on the number of the dynamic attributes. And with the usage of its name *Table1* the representation, reflecting the stored data in their original form, has been created. The representation *Table1* accomplishes the composition of the model AOID data and the time slice of all temporal tables, comprising all active values of the data at the moment of query. The *Legacy Application* interacts with the *Table1* by the means of queries, employed before the temporal extension of the table, and receives the results, corresponding to its original structure. The query for modification UPDATE, generated by the application, carries no data on the time range for the new value. The temporal framework intercepts this query automatically and by the means of the trigger *Trigger(U)* enlarges it with the set of activities oriented on keeping the previous version and setting the temporal attributes values for the new one.

Fig. 8 also presents the processing scheme for queries sent by the application, accounting the temporal data features (*Temporal DB Application*). The data modification in this case takes place by means of inserting the new version with all the time marks pointed. The temporal framework here plays the different part – employing the trigger *Trigger(I)* it actualizes the support of the data temporal integrity, performing the renewal of the time ranges of the neighboring versions and, if it is necessary, dividing the range of the previous version into two pieces.

Consequently the suggested method allows providing the database with two different interfaces simultaneously: the relation classical (non-temporal) interface and the temporal one. The first one provides the temporal upward compatibility with working non-temporal applications, giving the data actual for the moment of interaction, the second shows the temporal data in the original form. At the same time non-temporal interface supports all the operations of modification, and the temporal framework, invisible for the application, supports the temporal data integrity automatically. This method is considered and employed in practice in the 3rd chapter of the thesis and in the works [3, 5].

Nevertheless, for the whole range of the transport information systems it is not enough to interact with the values, active for the query moment only. First of all it refers to the systems, processing the accumulated historical data. Every entry of these data (payment, agreement, transport operation, etc.) is connected with the definite time moment, and consequently with the reference information active for this entry. However the queries sent to the reference data database by the working applications do not take into account the temporal data dimension, and therefore they are not able to get customized to the necessary time slice. *The method of providing user's temporal environment* is the solution for the problem of the non-temporal queries shift into the exact point in time. The fundamental of this method is the displacement of the current time point *now* and placement the user or application into the definite moment in time. The execution of all his/its queries

for the data occurs relatively the displaced point. This time point is specified in user's deliberately defined temporal environment and it is called environment variable. Hence the method allows performing the non-temporal queries to the multi-version data in various time slices. Below the principal concept of its functioning is shown in the performed example.

Assuming, there is a form of the non-temporal query *Q(O)* to the database, returning the object *O* data. After the process of allotting the database with the temporal dimension it comprises not one version of the object *O* state, but three of them: v_1, v_2 and v_3, (as it is presented in Fig. 9).There are two variants of the query functioning under the observation: under the condition of the non-defined variable of the user's environment and when this variable equals t_1 – a definite point of time in the past. In the first case the temporal framework executes the query relatively the current time *(now)* and the result of query *Q(O)* is the object version v_2, active for the moment of the query executing. In the second case the query is executed relatively to the determined time point in the time environment, equal to t_1 and in the result the version v_1, active for the moment t_1 will be selected.

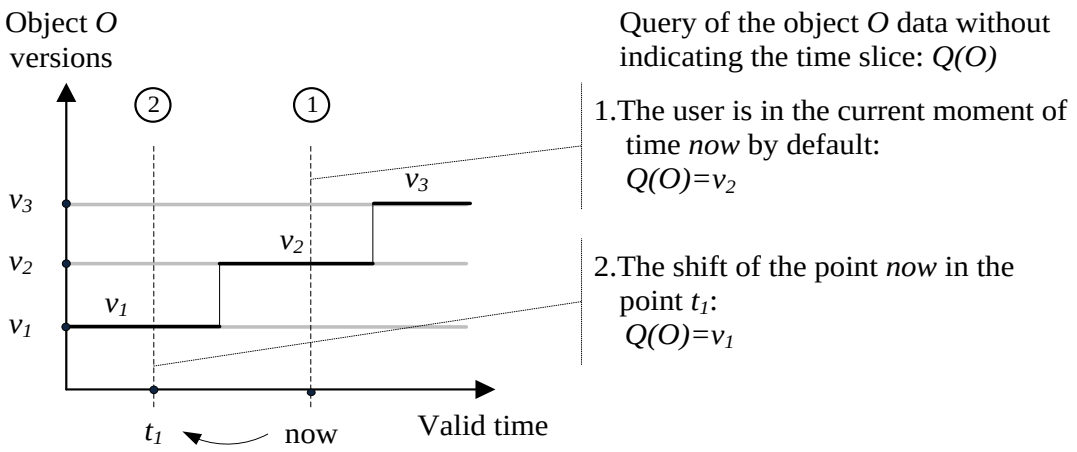


Fig. 9. Point "now" displacement for the user's queries

Fig. 10 demonstrates the transition from the non-temporal database to the temporal one for the existing application. It is shown in the scheme that the application generates the same non-temporal queries, however, besides the non-temporal interface holding, the new temporal database is capable of processing the queries in different time slices, which a necessary condition for more accurate application functioning. The example considers processing the queries of two independent users with different determined environment variables. It is obvious the users receive the different values as the responses to the same query, and these values are the active ones for the time moment, pointedly specified in the user's environment variable.

With the help of this method it is possible to manipulate efficiently the time slice, where the user has to work, without changing the application code and without employing the additional conditions in the queries. Under the condition of the environment variable alternation the queries start functioning in the different time layer automatically and in a flash. It is important, that using this method all users

have possibility to work independently as every user has his temporal environment determined.

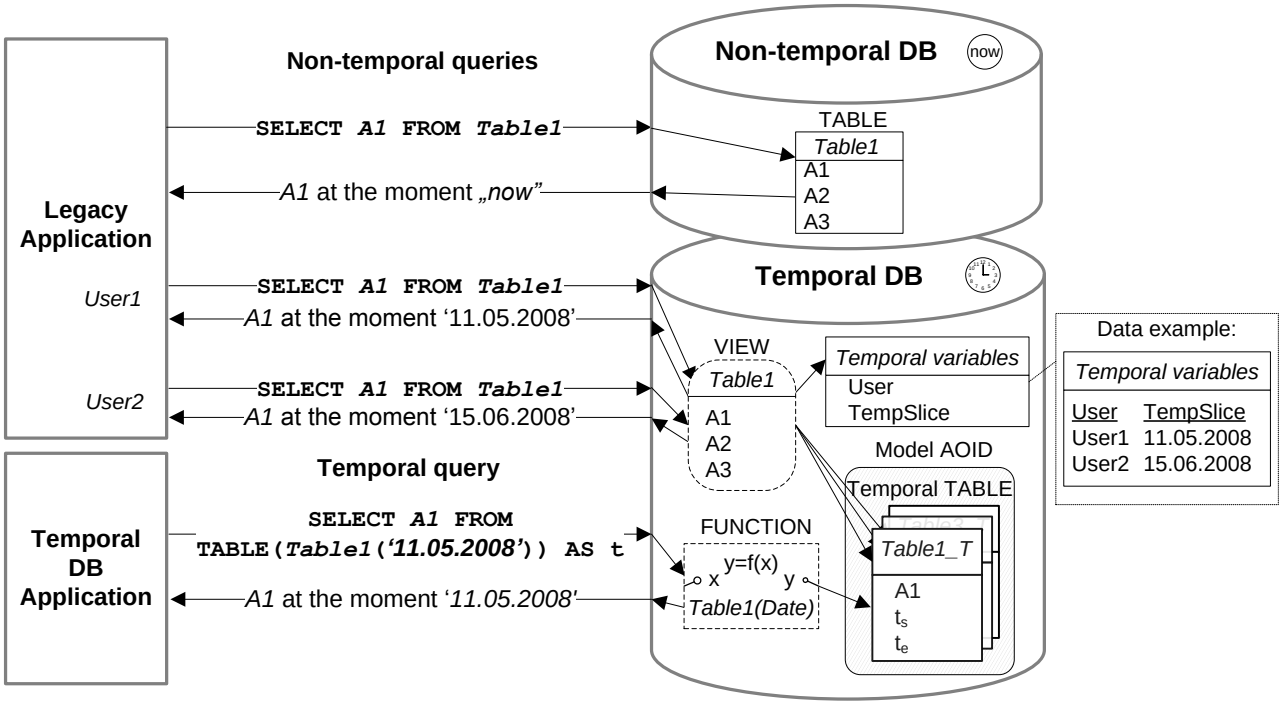


Fig. 10. Support of the interface of the data reading non-temporal operation with the user’s environment variable employment

The suggested method employment is reasonable in the case of new information systems development as well. Its application permits to simplify considerably the queries code for the multi-version data, as it delivers from the necessity to specify the additional conditions for getting into the determined time layer. This fact decreases the probability of the errors emergency in the queries code, and consequently it increases the reliability of the received results.

The user’s temporal environment is also suggested for using for the computation of the temporal aggregates, which are the necessary component of any analytical processing. Executing of the temporal aggregation procedures is connected with the determination of the time domain (the processed period with the determined dates of the beginning and the end), for which the aggregates are computed. The basis of this method is the time range determination within which the user operates, and its further forced employment in the computation. The complicated calculations, such as the preparation of the reports on the railway passenger transportation [1, 2, 6], use the series of the interconnected aggregates computations, and excluding the option of the time range specifying from the query will both simplify the query code and decrease significantly the probability of the errors emergency.

The user’s temporal environment is suggested for applying also as the facility of the temporal data security increasing. The special environment variable allows limiting the user’s acceptable field of the user’s operations by the data set, related to the determined time range. The application of this temporal environment variable is described in researches [9, 10]

10.5. Periodicity Model in the Railway Scheduling Tasks

The principal issues of the train traffic schedule storage in the databases are connected with the existence of the multitude of its versions conditioned by the seasonal cycles of schedule changes, the days of week, by the systematic and unplanned repair operations, the transfers of the working and festive days, and other factors. The basic problem is contained in the complexity of the data reliability providing and the immediacy of the schedule amending. Under the condition of the frequent alternations introducing these tasks become extremely urgent.

The *temporal object model with the periodicity and special calendar* and *the methods of the object active version determination: the logical rules method* and *the method of the temporal elements* (point 10.6 of this research and works [12, 13, 15]) were developed within the frameworks of this investigation for solving the above mentioned issues. The received results are applied in the process of creating the information system of the inland train traffic schedule on the Latvian Railway.

10.5.1. Periodicity Principle Employment in the Trains Scheduling

As a matter of practice the train can have several active schedules simultaneously during the long period of time (see chapter 1 of the dissertation under consideration). Furthermore every train schedule is organized inclusive of days of running with the different *periodicity properties*, for example: on working days, on weekends, on the definite days of week, the first working day of week, the last day off of the week (notably it can be even Monday), on even or odd days etc.

As an example the multi-variant train schedule is considered (Fig. 11). The figure performs that the train has two basic schedules – one for the working days *wd*, and one for the weekends *rd*. Both timetables are valid on the time interval [t_1, t_4], but on different week days. Then on Tuesdays and Thursdays (*Tue,Thu*) the amendment of this train schedule in the time period from [t_1, t_4] is introduced for the systematic repair operations providing. Therefore in the time interval [t_2, t_3] there are three timely timetable versions, activated on the days corresponding to the features of *wd*, *rd* and (*Tue,Thu*), moreover, only one schedule version *wd*, *rd* or (*Tue,Thu*), can be active one day.

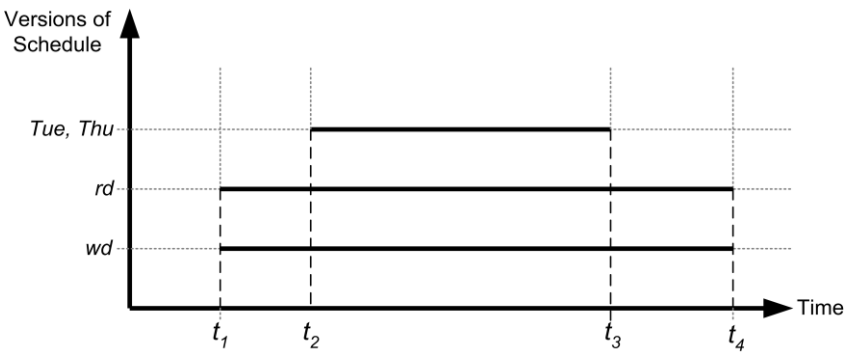


Fig. 11. Multitude of the active versions of the train schedule: *wd, rd* and (*Tue,Thu*)

This assertion is illustrated with help of the diagram in Fig. 12. The axis "Day of Week" shows the days of week, respectively 1 is Monday, 2 is Tuesday and so on. Time in this system is of the discrete nature, the level of the schedule granularity

equals one day period. As the mentioned figure presents, one of three versions is the active train timetable in different days of week, and the other two versions are in the "shadow" zone at this moment. Moreover, the train schedule on Tuesdays and Thursdays (characteristic *Tue,Thu*) overlaps the working day schedule with the feature *wd*.

However besides the recurring versions of the train schedule there are also one-time changes in the timetable, connected with the special (altered) calendar. He following changes can be listed in this category: the festive days, the additionally appointed days-off, or the canceled day-offs. There is also such special occasion schedule alteration when the calendar working day exchanges the places with the calendar day-off. For example, in Latvia in 2007 the 30th of April, Monday (a working day) was transferred on the 14th of April, Saturday (a day-off). As a result on the 30th of April the trains ran according to the schedule of the weekends, and on the 14th of April the trains ran according to the working days timetable – instead of the 30th of April.

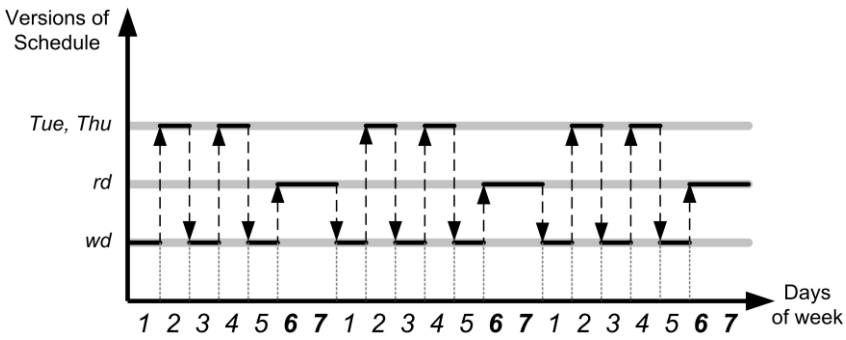


Fig. 12. Periodicity in the train schedule

Such types of changes in the timetable or assignations provoke the anomalies (paradoxes). For example, the 19th of November, Monday, in 2007 was announced as an additional day-off, and all the trains moved this day according to the schedule of the weekends and days-off (like on the 18th of November), and this event arouse the malfunction of the periodicity, as it is shown in Fig. 13. It is important to add that several trains with periodicity (Sundays) were canceled on the 18th of November, because the trivial Sunday is not only the day-off, but also the last day of weekend, and the additional Sundays trains are considered for the passengers transportation from the traditional locations of the resting facilities on the days-off.

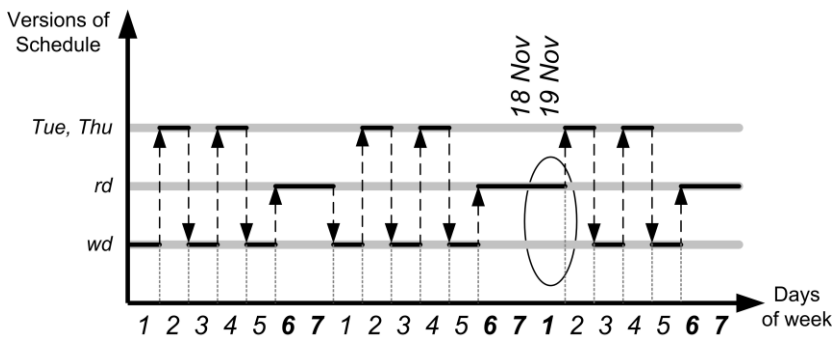


Fig. 13. Periodicity malfunction in the train schedule

For correct accurate functioning of the scheduling system, which is to take into account the above considered non-trivial rules, the well-defined formalization of the task assignment and the description of the scheduling temporal model are demanded. For this purpose the group of sets, characterizing the rules of arranging the trains running periodicity, the railway objects, and the entire system of scheduling itself are introduced.

10.5.2. Mathematical Description of the Railway Scheduling System

Two basic sets are introduced for the objects and rules description:

$S = \{s_1, s_2, \ ..., s_k\}$ is the set of all stations of the railways, where k is the capacity of the set S;

$N = \{n_1, n_2, \ ..., n_m\}$ is the set of the trains where m is the capacity of the set N.

The schedule for the train with the number $n \in N$ will be determined by the following vector type:

$$v = \{\langle s_1^{(n)}, T_1\rangle, \langle s_2^{(n)}, T_2\rangle, ..., \langle s_\alpha^{(n)}, T_\alpha\rangle\},$$

where the pair $\langle s_j^{(n)}, T_j\rangle$ determines the j-th train stop, the station $s_j^{(n)} \in S$ exactly, and the train departure time T_j; α is the number of the stops of the train running on the timetable v.

Several schedule versions, characterized by the operation period, the assignation time (fixation) and the periodicity, can be hold for the train number n. Consequently the following scheduling parameters are introduced:

t_s and t_e are the moments of time of the beginning and ending of the operational period of the schedule version in question respectively;

t_f is the time of the fixation of the schedule version in the databases;

C is the property of the periodicity employed for the determination of the days of the schedule in question functioning.

The property C presents the logical expression consisting of one or several elementary parameters of the periodicity connected by the logical operations signs $\vee, \wedge$ and $\neg$. Every elementary parameter of the periodicity p_i correlates with the predicate $p_i(x)$, that defines the day belonging to the exact group, for example, “the day is the even day of the week” and it can be *true* or *false*. The elementary parameters of the periodicity make up the set of properties which serves as the basis for the logical expressions generation, and these expressions designate any more complicated periodicity:

$$C(x) = f(p_i(x), \ i = \overline{1,\lambda}) \ , \tag{1}$$

where $p_i(x) \in \mathrm{B} \equiv \{true; false\}$, λ is an arity of the statement C.

The dissertation considers the different variants of the periodicity parameter, introduced by the set $P=\{p_1, p_2, ..., p_\pi\}$, where π is the set cardinality; the set comprises the following proposition as its elements: *ed* is "any day of the week" (it is used in case the train runs daily); *wd* is "a working day" (the train runs the working days); *rd* is "the day-off"; *Mon, Tue, Wed, Thu, Fri, Sat, Sun* "is the corresponding day of the week", *pd* is "the even days of the month"; *nd* is "the odd days of the month"; *lh* is "the last day-off of the week"; *fw* is "the first working day of the week, etc.

The predicate $p_i(x)$ determines the property of the day *x* to be the day with the property p_i. The set of the predicate validity $p_i(x)$ on the definition area $x \in [t_s, t_e]$ constitutes the aggregate of all days with p_i property in the range $[t_s, t_e]$. For example, the determination of the set of the predicate validity $Mon(x)$ for $x \in$ [01.05.2007, 31.05.2007] appears in the following form:

$\{x \mid x \in$ [01.05.2007, 31.05.2007], $Mon(x)\}=$
{07.05.2007, 14.05.2007, 21.05.2007, 28.05.2007}.

In the similar vein the truth set of the expression *C* for the specified version presents the aggregate of the days in the range $[t_s, t_e]$; this version is operative on these days.

For the specific schedule version identification for the train with number $n \in N$ the type of tuple $\langle n, C, t_f, t_s, t_e \rangle$ is employed.

Then *i*-th schedule version $v_i^{(n)}$ for the train with number $n \in N$ will be determined by the tuple type:

$$v_i^{(n)} = \{\langle n, C, t_f, t_s, t_e \rangle, \langle s_1^{(n)}, T_1 \rangle, \langle s_2^{(n)}, T_2 \rangle, ..., \langle s_\alpha^{(n)}, T_\alpha \rangle\}. \quad (2)$$

Assuming $V^{(n)} = \{v_1^{(n)}, v_2^{(n)}, ..., v_g^{(n)}\}$ as the set of versions of the schedule of the train with number $n \in N$, where *g* is the set capacity and it equals the number of the versions of train *n* schedule, then $V = \{V^{(n_1)}, V^{(n_2)}, ..., V^{(n_m)}\}$ is the set of versions of all trains timetables.

Further the sets describing the periodicity exclusions have been assigned prescriptively:

$EX(C) = \{ex_1^{(C)}, ex_2^{(C)}, \ldots, ex_\theta^{(C)}\}$ is the set of the days-exclusions $ex_j^{(C)}$, which have been assigned with the property *C*, where θ is the capacity of the set *EX*(*C*). Two sets can serve as the typical examples of the set *EX*(*C*): the set $EX(wd) = \{ex_1^{(wd)}, ex_2^{(wd)}, ..., ex_w^{(wd)}\}$, giving the periodicity attribute $C = wd$ to the additionally assigned working days, and the set $EX(rd) = \{ex_1^{(rd)}, ex_2^{(rd)}, ..., ex_q^{(rd)}\}$ the additionally assigned festive days (days-off) with the periodicity attribute $C = rd$;

$EX = \{\langle C_1, EX(C_1)\rangle, \langle C_2, EX(C_2)\rangle, \ldots, \langle C_l, EX(C_l)\rangle\}$ is the set of all exclusions for the different periodicity properties, where l is the capacity of the set EX; the set EX components re-determine the periodicity attributes for the dates, appointed in $EX(C)$;

$Z = \{z_1, z_2, \ldots, z_\mu\}$ the set of the transferred days of the trains schedule, where $z_i, i = \overline{1, \mu}$ is the pair of days $\langle d_1^{\{i\}}, d_2^{\{i\}}\rangle$, determining, that the timetable assigned on the date $d_1^{\{i\}}$, is assigned on the date $d_2^{\{i\}}$ as well; μ is the capacity of Z set.

Due to the complexity of managing the different periodicity schedule versions set, continuously appearing operative schedule changes, taking into consideration the non-trivial calendar, subjected to the changes, the development of the efficient methods of the active version determination is highly needed. In the investigation process two alternative methods have been suggested by the thesis author: *the logical rules method* and *the temporal elements method*, which are considered in this research and in the works [12, 13, 15].

10.6. Methods of Determining the Active Version of the Temporal Object

10.6.1. Temporal Elements Method

The peculiarity of the suggested method of determining the active version of the temporal object lies in the preliminary reckoning the dates of activity for every schedule version and saving it in the temporal elements for the posterior employment.

The temporal element (TE) represents the calculable set of dates and serves for the time determination when the exact version of the train schedule on the extent of its life cycle becomes active. The temporal element comprises all information on the periodicity of the timetable, taking into consideration all operative changes several versions overlapping and special calendar.

The timetable version $v_i^{(n)}$ of the train with the number $n \in N$, assigned with the expression (2), can be determined with the temporal element

$$v_i^{(n)} = (\langle n, TE_i^{(n)}\rangle, \langle s_1^{(n)}, T_1\rangle, \langle s_2^{(n)}, T_2\rangle, \ldots, \langle s_\alpha^{(n)}, T_\alpha\rangle), \quad (3)$$

where the temporal element $TE_i^{(n)}$ determines all the dates when the version $v_i^{(n)}$ is active.

The process of calculating the temporal element of the schedule version demonstrated by the tuple of relation in the form Historical-Overlapped, comprises two steps: 1) calculation of the basic value of the temporal element; 2) calculation of the actual value of the temporal element. There is the examination of these steps.

The first step produces the calculation of the basic value of the temporal element, marked TE^*. The calculations take into account only the period when the

version is active, and its periodicity. The times when the periodical events are active in the time specified range $[t_s, t_e]$ is performed as a pair $\langle [t_s, t_e], C \rangle$, and the set of the temporal element values $TE(C)$ in the time specified range $[t_s, t_e]$ is performed as the function of bounded extension $BExt([t_s, t_e], C)$. Applying the introduced designations and functions, the temporal element for the C periodicity in the time range $[t_s, t_e]$ is determined in the following way:

$$TE^* = BExt([t_s, t_e], C) \cup EX([t_s, t_e], C) - EX([t_s, t_e], \neg C), \quad (4)$$

where $EX([t_s, t_e], C) \subseteq EX(C)$ is the set of the days-exclusions with the obtained periodicity C, intervening in the period of schedule version functioning, in other word, $EX([t_s, t_e], C) = \{ex^{(C)} \mid ex^{(C)} \in [t_s, t_e]\}$;
$EX([t_s, t_e], \neg C)$ is the set of the days-exclusions without the property C.

The function $BExt([t_s, t_e], C)$ determines the points on the time axis (dates) within the frameworks of the time range $[t_s, t_e]$, responding the periodicity property C. The verification of every point $x \in [t_s, t_e]$ for correspondence to the logical expression C is the basis of this function. The result $C(x)=true$ means that the day x satisfies the schedule feature C, and $C(x)=false$ means that the version of the schedule is not active on x day. Then the set of points on the time axis within the time range $[t_s, t_e]$, responding to the periodicity property C, is generated by the following way:

$$(\forall x \in [t_s, t_e]) C(x) \Rightarrow x \in BExt([t_s, t_e], C), \quad (5)$$

As an example of calculation in accordance with the formula (5) the date definition in the time range $[t_s, t_e]$ is considered. These dates are the working days and the even days of the month. The task solution can be presented in the following way:

$$(\forall x \in [t_s, t_e])[pd(x) = true \wedge wd(x) = true] \Rightarrow x \in BExt([t_s, t_e], pd \wedge wd).$$

In general case in the process of computation of the predicate $C(x)$, it is necessary first to calculate all the predicates $p(x)$ comprised in it. Then the logical operations $\vee, \wedge$ and $\neg$, formulating the expression C, are applied to the results.

In the process of computation of the majority of $p \in P$ it is enough to know the checking point coordinates on the time axis. For example, under the condition of knowing the date it is possible to compute the day of week, and accordingly check it on the correspondence to the periodicity features *Mon, Tue, Wed, Thu, Fri, Sat* and *Sun*, and employing the rule of correlation of the days of week to the determination of the days-off, it is possible to check it on a correspondence to the properties *wd* and *rd*. However there are such periodicity properties, for checking which it is not enough to have the coordinates of the point under examination only, but it is necessary to know about the periodicity properties of other days. The very first and the very last days of the weekend can serve as examples of such "complicated" periodicity. It is necessary to know about the periodicity properties

of the neighboring days for their determination: the day before the date under examination for the first day of the weekend and the day after the date under examination for the last day of the weekend. The complexity is highlighted by the fact, that for these connected days some exchanges or transfers might be assigned, and it is necessary to take into consideration the periodicity properties with the account of all these changes.

The second step is implementation of the calculation of the actual value of the temporal elements, employing the fundamental value TE^*. The actual value of the temporal element TE_i of the schedule version, besides the properties of the periodicity and the facts of the dates transfer, takes into consideration the cases of the timetables overlapping. If such conflicts are detected, the identical dates are canceled in the temporal element of the schedule version with the earlier fixation time t_f corresponds to. Then the calculation of the element TE_i presents the recursion function. The recursiveness is shown in the fact that the calculation of the temporal element of one version comprises the temporal elements of other subsequent versions, which, in their turn, are determined in the same way:

$$TE_i^{(n)} = (TE_i^* - \bigcup_{v_j^{(n)} \in V(v_i^{(n)})} TE_j^{(n)}) \bigcup z_i^{(-)} - z_i^{(+)}, \tag{6}$$

where $V(v_i^{(n)})$ is the set of versions of the schedule of the train with number n, functioning of which occupies the part of the period of the version $v_i^{(n)}$ functioning and the time of their fixation is subsequent compared to the version $v_i^{(n)}$; $z_i^{(-)}, z_i^{(+)}$ are the sets Z elements, adjusting the temporal element of the version with the account of changes of type:

$z_i^{(-)} = \{d_1 \mid z = \langle d_1, d_2 \rangle, z \in Z, d_2 \in TE_i^{(n)}\}$ is the set of the dates, which are to be excluded from the temporal element of the version $v_i^{(n)}$, because there is exchange assigned for them;

$z_i^{(+)} = \{d_1 \mid z = \langle d_1, d_2 \rangle, z \in Z, d_1 \in TE_i^{(n)}\}$ is the set of the dates, which are to be comprised in the temporal element of the version $v_i^{(n)}$, because its temporal element $TE_i^{(n)}$ contains the date, the timetable of which is the standard sample for the date transfer.

It should be noted that after the adjustment temporal element in accordance with the formula (6) the temporal elements of different versions of train n schedule do not intersect, in other word,

$TE_i^{(n)} \bigcap TE_j^{(n)} = \varnothing$ for $i \neq j;\ i, j = 1,2,...,g.$

Fig. 14 illustrates the transfer of the schedule from the 13th of April, Friday to the 14th of April, Saturday, comprising the exclusion of the 14th of April from the temporal element of the schedule #2 for the days-off (*rd*) and inclusion it in the temporal element of the schedule #1 for the working days (*wd*).

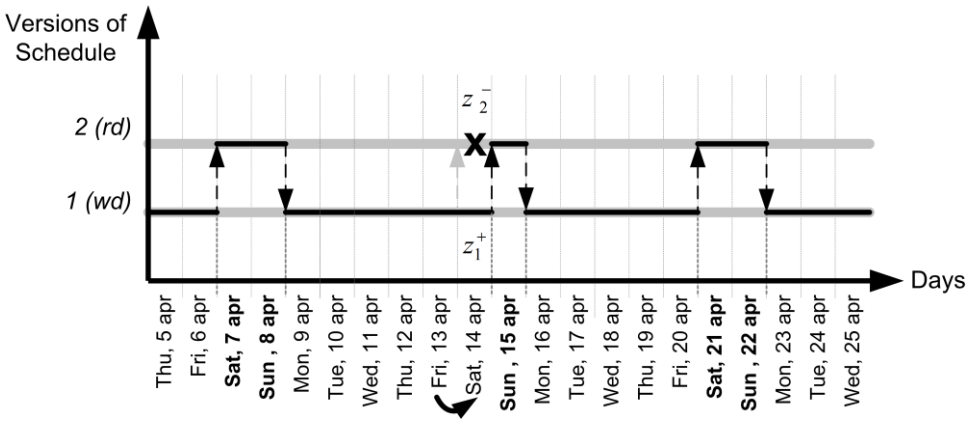


Fig. 14. Schedule transfer from one day to another

The implementation of the method of the temporal element generation with the employment of different approaches is reviewed in the 3rd chapter of this dissertation and in works [12, 13, 15].

The existence of the temporal elements allows considerably simplifying the numerous temporal functions, such as the determination of the active object version, the determination of the points on the axis of the effective time when the exact object version appears from the shadow zone. For *the active version of the schedule of the train with number n determination* it is suggested to employ the function of $ActVer()$ type:

$$ActVer(x, V^{(n)}) = \begin{cases} v_i^{(n)}, & \text{if } x \in TE_i^{(n)}; \\ null, & \text{otherwise}, \end{cases} \tag{7}$$

where x is the date for which it is necessary to know the schedule.

The result of the function $ActVer(x, V^{(n)})$ is the element of the set $v_i^{(n)} \in V^{(n)}$, presenting the required active version. If the train does not run on the pointed date x, the function returns the value of null[1]. The condition of non-crossing of the temporal elements of all train schedule versions guarantees the existence of only one active version.

One of the most serious problems of the temporal elements employment is the *maintenance of the temporal elements* in the consistent condition, because in the case of the schedule changes appearing the versions temporal elements might become out-of-dated and they are to be re-calculated. However some trains continue running for several years, and have the numerous versions of their timetable, the part of which reckon only towards the past, and the introduced changes, affecting the present and the future, in no way are reflected on the out-of-dated temporal elements. In other words, the range, affected by any changes, can be not overlapped by the life cycle of the version or be overlapped only partially. In this case it is unreasonable to upgrade the temporal elements of all train schedule versions, or to re-calculate them entirely, in general. The method of the partial re-calculation of the temporal elements is presented in the 3rd chapter of the dissertation.

[1] *Null* is a special marker used in Structured Query Language

10.6.2. Logical Rules Method

The essence of this method is in the computation of the active version of the timetable on the basis of the logical rules, taking into consideration all the peculiarities of the schedule: multi-version, provoked by the functioning changes, and polyinstantiation in case of the periodical schedule, exchanges and transfers. The method functions with the data of the schedule temporal database and begins taking into account all the changes in the timetable at the moment when they are introduced to the system (i.e. on-line).

The method calculations apply the set of the schedule versions $V^{(n)}$ for the train with the number *n*, the set of exclusions *EX* and the set of exchanges *Z*. The train schedule version is performed according to the formula (2), and only its identification part $\langle n, C, t_s, t_e, t_f \rangle$ is used in the calculations.

The method central parameter, relatively to which all the logical rules, built into the algorithm, are employed, is the day *x* – the date for which it is necessary to determine the active version of the train schedule. In the connection with the specific calendar existence it is essential to check the date *x* for the purpose of re-determination of the periodicity properties and exchanges. In case they have been discovered the date *x* is to be re-determined into the date $\hat{x}$ and the further computations are to be done relatively $\hat{x}$:

$$\hat{x} = \begin{cases} d_1^{\{i\}}, & \text{if} \quad \exists z_i \in Z \mid z_i = \langle d_1^{\{i\}}, d_2^{\{i\}} \rangle, d_2^{\{i\}} = x; \\ x, & \text{otherwise.} \end{cases} \tag{8}$$

Only the following schedule versions can be valid for the date $\hat{x}$: first, the versions, obtaining the range of functioning capable of covering the date under the question, and, second, the periodicity property complying with this date. Consequently, it is essential to single out of the set of the train *n* schedule versions only the versions corresponding to the functioning time and to the periodicity, in other word performing on the day $\hat{x}$ and conforming to the property *C*. There is the version complying the periodicity of the day $\hat{x}$; it is the version for which the set of its predicate $C(\hat{x})$ verity on the functioning field of this version $[t_s, t_e]$ comprises the date $\hat{x}$, in other word, $\hat{x} \in \{y \mid y \in [t_s, t_e], C(y)\}$. However it is not necessary to compute the verity set, it is quite enough to apply the proposition *C* itself to the day $\hat{x}$. In case the result is the veritable one, the version satisfies the condition of the day $\hat{x}$ periodicity. Thus, the set of all versions corresponding to the functioning time and the periodicity of the date *x* is sorted out of the set of the train $V^{(n)}$ schedule versions in accordance with the following formula:

$$V_{\hat{x}}^{(n)} = \{v \mid v \in V^{(n)}, \hat{x} \in [t_s, t_e], C(\hat{x})\}. \tag{9}$$

For the taking into account the periodicity exclusions for the statement *C* the predicates $p_i(\hat{x})$ the predicates are exchanged by the $\hat{p}_i(\hat{x})$. The predicate $\hat{p}_i(\hat{x})$ in its essence corresponds to the predicate $p_i(\hat{x})$ described in the point 10.6.1., but

before executing the computations, it performs the preceded additional inspection oriented on the detection of the exclusions existence. In case of discovering such date $\hat{x}$ the feature p_i is given to it. The below formula performs it.

$$\hat{p}_i(\hat{x}) = \begin{cases} True, & \text{if} \quad \hat{x} \in EX(C), EX(C) \in EX, p_i = C; \\ p_i(\hat{x}), & \text{otherwise.} \end{cases} \tag{10}$$

N.B. In case the scheduling system employs the periodicity properties, dependent on the periodicity properties of the neighboring dates, such as *lh* is "the last day-off of the week" and *fw* is "the first working day of the week", the predicate formulae $p_i(\hat{x})$ and $\hat{p}_i(\hat{x})$, comprised in *C*, have to be extended to $p_i(x, \hat{x})$ and $\hat{p}_i(x, \hat{x})$ respectively. It is impossible to detect accurately the periodicity properties of the neighboring dates, obtaining the information on $\hat{x}$ only.

In case of the schedule overlapping versions existence (the Historical-Overlapped) the set $V_{\hat{x}}^{(n)}$ might contain more than one element. Consequently it is essential to single out the sole element – the active version – from the $V_{\hat{x}}^{(n)}$. In the course of calculation the priority is given to the version which has been introduced to the system after all the others:

$$v_a^{(n)}(x) = (v \mid v \in V_{\hat{x}}^{(n)},\ t_f = \max_{V_{\hat{x}}^{(n)}} \{t_f\}). \tag{11}$$

The merit of this approach is the fact that all the computations take place on the basis of the operative data and all the changes are taken into account by the queries to the schedule at once after their introducing to the system. Concerning the shortcomings of the method, it should be noted that every time the above described searching algorithm have to be executed all over again, in case it is necessary to execute any query to the system, requiring the knowing of the active version of the train schedule.

The research performs the approbation of the suggested methods of the determination of the object active version in the information systems of the trains traffic schedule on the Latvian Railway and the work defines the efficiency of their application for the solution of different tasks, emerging in the practical operation. The procedure of the investigation in question comprises three steps:

1. Defining the information systems functional tasks, determining the object active versions, and choosing the tasks awaiting for solution;
2. Organizing the system of the method efficiency indicators for the object active version determination and the choice of their estimation method;
3. The methods approbation in the process of the information systems chosen tasks solution, their efficiency estimation and the elaboration of the guideline for their practical application.

The results of the comparative efficiency estimation and the possibilities of the employment of the methods of the object active version determination in the information systems of the trains traffic schedule is performed in table 2. Every of the

indicators possesses the value of “+” or “-“: it takes the value “+” if the method preponderates over its rival in this characteristics, and the value “-“ otherwise.

Table 2. *Comparison of the methods of the object active version identification*

	Method LR	**Method TE**
Efficiency Indicators	**Methods Comparative Efficiency Estimation**	
The necessity of the preliminary data preparation for the application to the queries after the changes introduction to the schedule	+	–
The complexity of the active version searching (the number of operation in the process of searching) *	+ (> 7)	+ (1)
The performance in the process of the active version searching **	–	+
The volume of the data processed in the process of the active version searching	–	+
The necessity to execute the restrictions on the time area for queries	+	–
The operational efficiency of the changes introduction	+	–
Providing the data integrity in the process of the schedule changing	+	–
The volume of the stored data***	+ (~27KB)	– (~2000KB)
Application Tasks of the Scheduling System	**Application Possibilities Estimation**	
The process of searching the train timetable active version for the specified day	+	+
Generating the sequence of the versions changing for the specified range	– (for the broad ranges) + (for the narrow ranges)	+
The process of searching the intervals of the determined version operation	–	+
The process of searching the schedule immediate changes	–	+
The revelation of the time points (the dates) of the schedule changes in the specified range	–	+

There are some explanatory notes on the indicators, presented in table 2.

* *The complexity of the process of searching of the active version* has been estimated by the number of executed operations. The logical rules method requires the employment of the suite of the logical rules requiring at least seven logical verifications, and their number depends on the versions number and their *C*. The average number of operations can be calculated according to the following formula: $7+|\overline{V^{(n)}}|\cdot\overline{C}$, where $|\overline{V^{(n)}}|$ is the average number of the single train schedule versions (the set capacity is $\overline{V^{(n)}}$), $\overline{C}$ is the expression central area *C* (the average number of the components in *C*). For the schedule of the Latvian Railway the average number of operations equals 15 (in 2004 the value of $|\overline{V^{(n)}}|$ was 6 versions, in 2005 they were 4, in 2006 – 3, and in 2007 – 4 versions, $\overline{C}$ = 1.8). The process of searching in the temporal elements method performs only one elementary verification operation for examining entry in the temporal elements.

** *The execution speed*. The time of searching depends on the peculiarities of the implementation, the databases management system, and so on. However, it inevitably correlates with the number of logical verifications. Therefore, the temporal elements method unquestionably gains the advantage in this position

*** *The volume of the data stored* has been compared in the identification part of the schedule, in other word, without considering the data on the stops at the stations, whereas the volume of these data is the same for both methods. The comparison has been done on the basis of actual passenger trains schedule on the Latvian Railway of year 2004: the total number of trains is 389, the number of the schedule versions is 2031.

10.7. Method of the Access Control to the Temporal Data on the Relation Tuples Level

The peculiarity of the temporal data protection lies in the fact that the data of the different time periods are logically disposed in the same place – in the integrated relation table. The delimitation of the users' access to the different time periods, in other word to the subset of the relation tuples, is not comprised in the suite of the standard facilities of the security management of the relation DBMS, which required the additional elaboration.

The method proposed by the author is established on the basis of the access restriction to the data of the exact table records for reading, modifying and cancelation procedures. This method application allows delimitation of the access in such a manner that the user can receive the access to one of the table records (one version) but cannot receive it to the others. The method implementation employs only the standard procedures SQL-2: the triggers and the conceptions, the system mathematical apparatus – the relational algebra. The method is surveyed and applied in chapter 3 of the dissertation and in the author's works [4, 7, 9].

For the task formal characterization the sets taking part in the access control procedures are determined:

$\Re = \{R_1, R_2, \ldots, R_n\}$ is the set of DB relation, $|\Re| = n$, where R_i, $i = \overline{1,n}$ gains the form of the scheme $R_i(A, L)$;

r is the tuple of the relation R_i, the protection object, $r \in R_i$;

$A = \{a_1, a_2, \ldots, a_k\}$ is the subset of the R_i attributes, having no impact on the access to its tuples, $|A| = k$;

$L = \{l_1, l_2, \ldots, l_m\}$ is the subset of the R_i attributes, determining the access to its tuples, $|L| = m$. There can be the temporal attributes, the specific security attributes or some other attributes or their scalar derivatives among them;

$U = \{u_1, u_2, \ldots, u_z\}$ is the set of the system registered users, $|U| = z$;

$\Theta = \{\theta_1, \theta_2, \theta_3, \theta_4\}$ is the set of the access operations to the tuple, $|\Theta| = 4$, Θ ={SELECT, INSERT, UPDATE, DELETE};

$\boldsymbol{S} = \{S_1, S_2, \ldots, S_h\}$ is the set of the security policy rules suit, regulating the users' access to the table data from $\Re$, $|\boldsymbol{S}| = h$. The rules suits $S_1, S_2, \ldots, S_h$ can have the different structure, depending on the specific exact rule of the safety policy. The suit $S_j \in \boldsymbol{S}$, $j = \overline{1,h}$, contains the samples of the rules $s_1^{(j)}, s_2^{(j)}, \ldots, s_f^{(j)}$ of the same structure, $|S_j| = f$. In the process of the method implementation this rules arrangement allows placing them in the unified relation as the set of tuples manipulation with which permit to control the users' access to the data. For example, to limit the access to the temporal data of the relation R_i depending on their location on the time axis, the rule $s_y^{(j)} \in S_j$ has the following form:

$$s_y^{(j)} = \langle R_i, u_c, t_s, t_e, \theta_q, g_d \rangle, \; j = \overline{1,h}. \tag{12}$$

The rule $s_y^{(j)}$ prohibits (g_d=0) or permits (g_d=1) the user $u_c \in U$ to execute the operation $\theta_q \in \Theta$ on the object version, represented by the tuple *r* of the relation $R_i \in \Re$, and actual in the time range [t_s, t_e].

The calculation of the fact of the tuple *r* accessibility is determined in the moment of the query to it with the help of the predicate $P_b \in \Pi$, where $\Pi = \{P_1, P_2, \ldots, P_h\}$ – is the set of the predicates of the information security system, employing the set $\boldsymbol{S}$ for computation; $|\Pi| = h$, $b = \overline{1,h}$. Every predicate has the following structure in the general case:

$$P_b(R_i, u_c, \theta_q, L, t), \tag{13}$$

where R_i is the relation, the tuple *r* attributed to; u_c is the user, executing the operation θ_q with the tuple *r*; *t* is the time of query (of the operation); L is the value of the tuple of relation R_i attributes, influencing the access permission. For the above mentioned example of the access delimitation according to the time range *L* consist of t_s and t_e, i.e. L=(t_s, t_e). In case if the P_b true the access to the tuple $r = (A, L)$ is allowed or, otherwise, is restricted.

The forced application of the predicates from Π is implemented in the data view, employed for the reading queries accomplishment and in the triggers, controlling the queries for the data modification.

Chapter 3 of the dissertation considers the method implementation example for providing the privacy of the railway rates temporal table data.

In the process of the method development it was taken into account that the data security system in respect of their temporal measurement is to be integrated with the operative databases security system, whereas the temporal measurement is not a sole parameter due to which the access to the transport information system data is restricted. That is why the data protecting method, proposed by the author, permits to differentiate the access not only regarding the temporal data measurement, but regarding any measurement or property appearing in the data: territorial,

geographical, administrative, etc. The author's research [7] is devoted to the development of the requirements to the complex security system of the relation data.

10.8. Interval Data Transformation Method

The models of the multi-version data representation, employed in the transport information systems, usually save data in the interval form which is inconvenient for the analytical calculations carrying out. The majority of the employed data analysis methods and correspondingly the majority of the analytical systems function with the point forms of the data representation (for example, with the time sequences), but not with the interval forms. In the point form the fact is associated with the set of the time moments of the previously chosen granularity (for example, with the days, months, or years). The procedures over the described sets are very convenient for the fundamental computations or for the formal characterization. However, the mathematical apparatus of the relation databases management systems does not provide the ready-to-use facilities for the transformation of the interval form into the point one, and this fact required the elaboration of the appropriate procedures to be performed by the investigation under consideration.

Within the framework of the set task, the aggregate scheme of the interval temporal data transformation was designed. This scheme is demonstrated in Fig. 15.

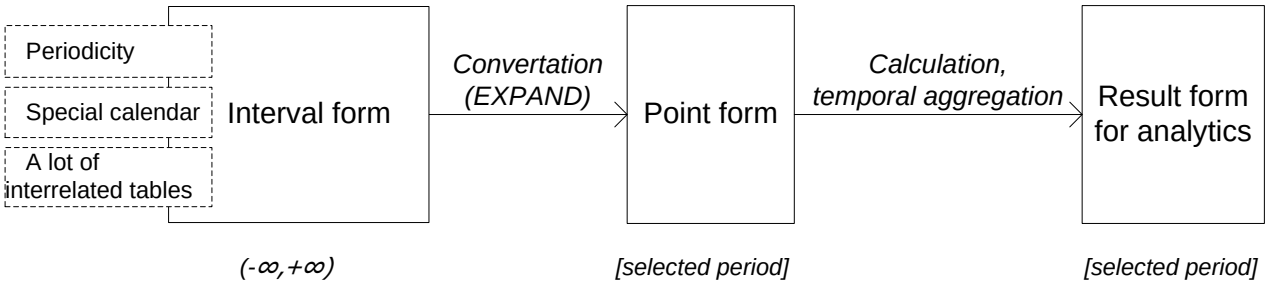


Fig. 15. Scheme of the temporal data transformation

There is the explanation of the presented scheme.

The classical interval form is initially inconvenient for some calculations, but their complexity is multifold increased under the following aspects presence: the periodicity, the specific calendar, the necessity to employ several interconnected tables data in the computation[2].

The application of the procedure EXPAND for the interval form temporal data, suggested by the author, solves the problem of the inconvenient data presentation for the computations, by the way of turning the interval facts into the time sequences, in which every value gives characteristics of this fact existence in the definite moment of time (*Point form*). Then the data undergo the further processing, often connected with the temporal aggregation procedures. This stage might require even more serious computations connected with the temporal concepts representation. The

[2] The information systems objects in the databases, according to the relation theory are represented in the form of the interconnected tables. That is why receiving the demanded result, as a rule, requires processing procedure for several tables.

required data form (*Result form for analytic*) is the final result of the processing procedure.

The initial sets for every stage embrace the different time periods. The set of the interval form temporal data (the first block in Fig. 15) is capable of covering the unlimited time range (-∞,+∞). With the transfer to the point form (the second and the third blocks in Fig. 15) the time area of the temporal data is narrowed to the field of analysis (selected period), and this fact considerably contracts the data bulk in the computations and substantially saves the computing resources, and solves the problem of versions existence with the indefinite values of the functioning periods beginning and ending.

There is the observation of the suggested method of the interval data transformation in the point form. There are two relations:

R1 is the temporal relation of the interval form, describing the set of the certain object versions and having the scheme $R1(A,t_s,t_e,C)$, where $A=\{a_1,a_2\ldots a_k\}$ is the set of the attributes, describing the object, *k* is the number of attributes. Every version has the functioning period from the moment t_s to the moment t_e and possesses the feature of periodicity *C*. Besides, the activity moments of this or that versions depend on the specific calendar *Z*, describing the one day properties transfer to the others (the additional days-off or additional working days, etc.);

R2 is the point form temporal relation with the scheme $R2(A,d)$, where d is the time moment when fact A is active.

It is necessary to reorganize $R1(A,t_s,t_e,C)$ into $R2(A,t_n)$, and the data set *R2* has to be restricted by the fixed time period (specified analysis period) $T^{(q)}=[t_s^{(q)},t_e^{(q)}]$, where $\forall d\in T^{(q)}$.

The transformation formula has the following view:

$$R2=\sigma_{d\in T^{(q)}}R1\times\tau(t_s,t_e,C), \tag{14}$$

where $\tau(t_s,t_e,C)$ is the function distributing the range $[t_s,t_e]$ on the dates *d* set, responding the requirement of the specified periodicity *C*:

$$\tau(t_s,t_e,C)=\{d\mid d\in[t_s,t_e],\, p(d,C)=true\}, \tag{15}$$

$p(d,C)$ is the periodicity predicate, which is true in the case if the date *d* has the property. The predicate takes into consideration the specific calendar *Z* (for example, the transfer of the weekends schedule on Friday and on Monday in the connection with the national holidays in Latvia on the 1st and on the 2nd of May in year 2009);

$R1\times\tau(t_s,t_e,C)$ is the construction, implementing the procedure of expansion EXPAND; this procedure is the central operation of the entire transformation and provides the transfer from the interval form to the point one. The selection $\sigma_{d\in T^{(q)}}(...)$ restricts the result by the analysis area $T^{(q)}$. The capacity of the resulting set can reach $|R1|\cdot|\tau(T^{(q)})|$.

After the executing the expansion procedure the following reorganizations (grouping, the natural combination, the projection) become trivial. The employment the proposed function $\tau(t_s, t_e, C)$ permits to simplify the computations ultimately. For its implementation the recursion procedure is used and with the help of the language SQL obtaining the dates set $D = \{d\}$ of the specified periodicity C between the time moments *dStart* and *dEnd* has the following view:

```
WITH D (d) AS
  (  VALUES dStart
     UNION
     SELECT d + 1 day as d    FROM D WHERE d < dEnd )
SELECT d FROM D WHERE p(d,C).
```

10.9. Implementation of the Research Results in the Data Warehouses Systems

In the process of investigation there has been developed the conceptual approach towards the corporate data storage and analyzing in the Latvian Railway information systems. This approach is based on the data warehouses technology, implemented in practice and in the author's works [1, 2, 8]. These works demonstrate the solutions for some important practical tasks, among them there are such issues as the bulk data processing system designing, the development of the designing fundamentals of the data warehouses in the systems of the decision making support on the railway, providing the stored data integrity and security, etc. Furthermore the special attention has been paid to the problem of the temporal data employment in the warehouses. This can be explained by the fact that the time is the principal measurement in the data warehouses. The essential processes of the data warehouses functioning is connected the time: unloading, transformation, loading (Extract, Transform, Load, or ETL), further data processing and the data marts production. The arrangement of the efficient interaction with the data time measuring in the warehouses determines the data integrity and security, the quality of their processing (the number of errors), the simplicity of the implementation of the warehouses functioning processes the immediacy of the changes introduction to them.

The basic problem of the interaction with the temporal data lies in the fact that either the operator or the programming facilities applied might interact with the data from another time zone, not relating to the time range under processing. It is explained by the imperfection of the data temporal measurement procedures, and sometimes by the complete absence of this measurement in the employed databases. This mostly refers to the reference data. This problem occurs on the different stages of the data processing in the warehouses. The technological process of the preparation the data for the analytical system of the passengers transportation on the Latvian Railway is performed by Fig. 16 in a schematic form. The data from the operative *data sources* undergo the processes of decoding and cleaning up, then they are placed in the *data warehouse* and distributed among the *data marts*. Distribution is the complex of transformation with the participation of the reference data database,

containing the data transformation rules dependant on time. After the data loading in the data marts the analyst estimates the results and makes the decision about the data readiness for the further analysis and the reports generation. In the process of estimation the analyst follows the aggregated data from the operational systems and other sources. If the result is unsatisfactory, the part of the technological process is repeated and completed with the data marts and data warehouses cleaning up procedures. Depending on the source of the error occurrence, the technological process is activated from the different points. The error source might be in the procedures of decoding and cleaning up, in the initial date, in the reference data database, in the processes of the data transformation.

The above mentioned procedures criticality increases under the condition of the malfunction of the specified order of the data loading in the analytical system, when the currently processed period is deeper in the past than the earlier processed period. In this situation the author has exposed the following problems:

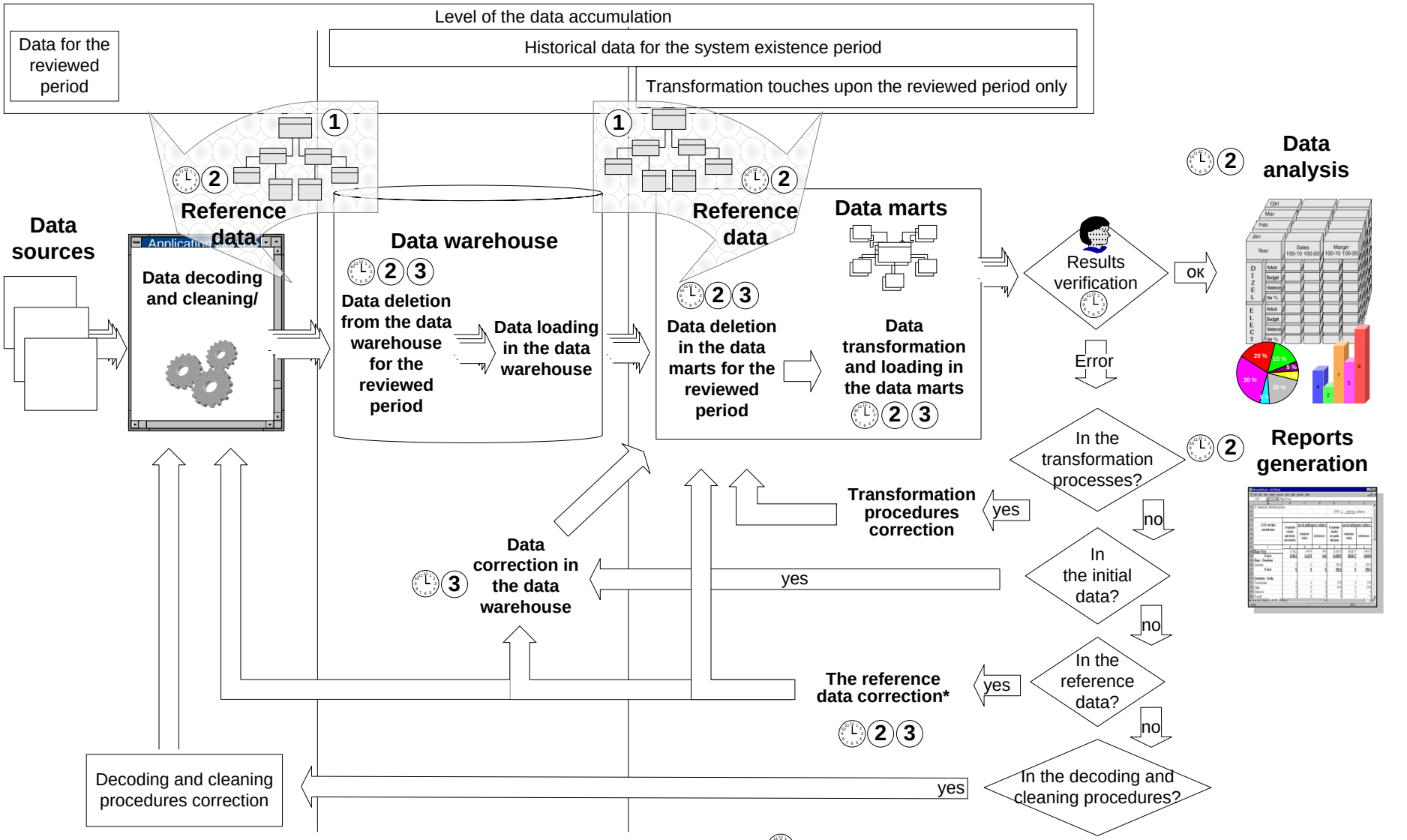
- data transformation for the data marts is based on the data of the previous processed period, and consequently the part of the technological process is to be executed cyclically, starting with the processed period and to the extent of the last one in the system;
- procedures of the marts cleaning up and the data transformation are not intended for the data processing for any time period and they operate only in the specified time range; this time range is determined by the sequence of their usage;
- reference data (in the non-temporal database or in case of the erroneously derived temporal slice) do not correspond to the processed period, therefore the data processing results might be false.

In the technological process of the information-analytical system functioning there have been exposed at least four critical procedures, in the process of executing which the probability of the malfunction of the data warehouse information resources integrity is rather high. They are the following:

- partial data deletion in the data warehouse;
- data correction in the data warehouse;
- reference data database correction;
- partial data deletion in the data marts.

Accordingly, the quality of this system functioning directly depends on the availability of the:

- temporal measurement in the reference data database;
- unsophisticated and convenient facilities for the temporal slices obtaining (the operation which will complicate the data processing code not in a great extent);
- adjustment facilities for the processing period for the data processing procedures simplification; as well as the prevention of the unpremeditated changes of other periods data.



The reference data correction* – depending on the fact what data have been corrected or replenished, the technological process recommences from the different stages. It is sometimes convenient to correct the data in the warehouse, but not to decode and to reset them again

- Time dimension support in the queries to the data
- (1) - Model AOID
- (2) - Method of providing the user's temporal environment
- (3) - Method of the access control to the temporal data on the relation tuples level

Fig. 16. Diagram of the data warehouse augmenting and the data preparation for the analysis

Implementation of the methods and models developed in the thesis permits to solve the mentioned issues: to decrease the risk of the data warehouse integrity breaking, to simplify the data transformation and preparation for the analysis for any time period and to increase the immediacy and safety of these operations:

1. For the solution of the task of entering the definite time slice, corresponding to the processed data, it is proposed to employ *the method of providing the user's temporal environment*. This task is in demand at many stages of the technological process, starting up with the decoding stage and ending with the stage of data transformation in the marts and the results verification.

2. For the access delimitation to the data warehouse, referring to the non-processed time ranges, it is offered to apply the *method of the access delimitation to the relation systems data on the relation tuples level* [4, 7, 9] on the purpose of the prevention of the data unpremeditated changes.

3. *The method of the access delimitation to the relation systems data on the relation tuples level* is suggested for employing for the reference data database changes as well.

4. Whereas the reference data database is a temporal one, the processes of the data changing by the user, occurring in the definite time slice, are not trivial. If the operator does not determine the values of the time attributes for the changed data, the new object version, appearing under the condition of the data changes, has the operation time equal to the user's time range, described in his *time environment*. Accordingly, the reference data changes do not touch upon the other periods.

The marks 🕒 and ①, ②, ③ in Fig. 16 perform the implementation areas of the developed models and methods in the process of the temporal data processing, and among them there are: the queries to the temporal reference data, the temporal sampling and aggregation, the data deletion of the specified time period, the access delimitation to the data referring to the unprocessed time range.

10.10. Investigation Results Implementation in the Interactive Data Referenced System of the Passenger Trains

The system of the passenger trains interactive scheduling SAR is one of the principle objects of the research under investigation. The object is interesting by the fact of operating with the temporal data obtaining the maximum amount of the temporal peculiarities in comparison with the other Latvian Railway information systems. The following peculiarities can be emphasized: the multi-version, the repeated cases when one and the same version becomes operative, the periodicity, and the peculiar calendar. The large data bulk and the requirements for the high speed of performance of the queries to these data increase the interest to the object.

The system of the passenger trains interactive scheduling has the WEB interface, applying which all the Internet users can request for some information on the passenger trains schedule for any day and in any direction in the interactive mode. The system permits to take into account any prompt changes in the existing schedule and to project the timetable for the long run.

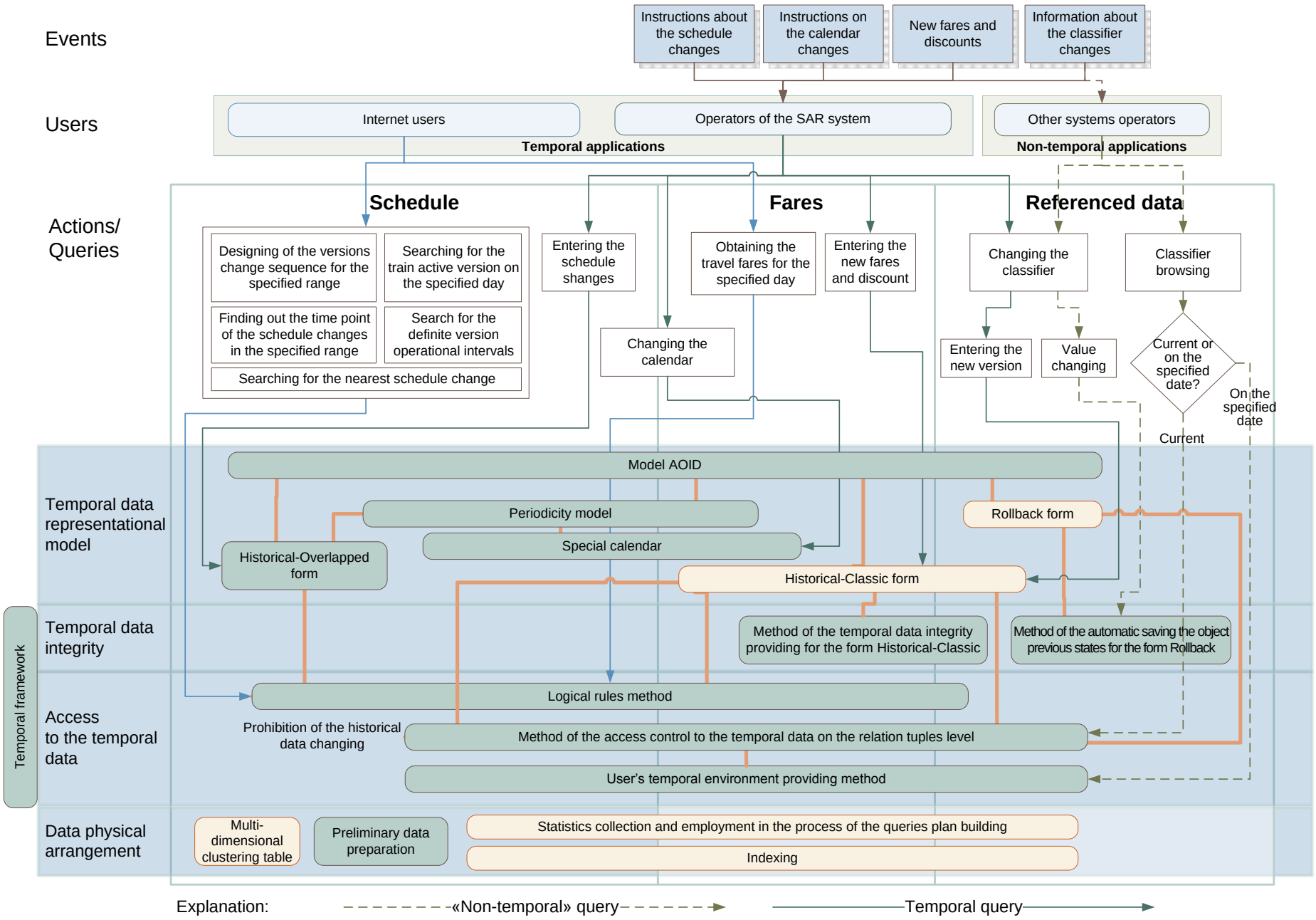


Fig. 17. The diagram of the developed methods and models employment in the system of the interactive passenger trains scheduling

The system manipulates with the data thematically, referential to three fields: the schedule, the fares, and the classifiers (reference data). The users, interacting with this system data, can be divided into three categories according to the nature of their queries: the Internet users, the operators of the scheduling and fares systems introduction, and the operators of some other systems, including the inherited systems, not capable of operating the temporal data.

The diagram of the developed methods and models employment in the system of the interactive passenger trains scheduling in the process of different users' categories interaction with the scheduling system data is performed in Fig. 17.

The methods and the models, in accordance with the presented functions, are divided into 4 groups: the temporal data representational model, the temporal data integrity, the access to the temporal data and the data physical arrangement.

The developed methods and models employment in the system of the passenger trains interactive scheduling allows:

1. Considerable simplifying of the system development;
2. Scheduling and fares data arrangement in the most space saving way, taking into consideration all the above mentioned data temporal peculiarities (the model AOID, the form Historical-Classic, the form Historical-Overlapped, the model of the periodicity and the special calendar);
3. Transformation of the segments of the referenced data central database into the temporal ones, and at the same time provision of the temporal upwarding compatibility with the other previously developed applications (the form Rollback, the form Historical-Classic, the method of automatic saving of the object previous states, the method of providing of the user's time environment);
4. Provision of the well-formed proper access to the data active versions (the logical rules method);
5. Provision of the data temporal integrity (the method of the temporal integrity provision for the Historical-Classic, the method of automatic saving of the object previous states);
6. Provision of the inviolability of the schedule historical data, the fares and the referenced data (the method of the access delimitation of the relation system on the relation tuples level);
7. Reaching the required speed performance with the employment of the different indexing methods, including clustering.

At the current moment the system of the passenger trains interactive scheduling, implemented with the usage of the developed methods and models, is successfully employed on the Latvian Railways.

CONCLUSIONS

The accomplished research principal results can be presented in the terms of the conclusion:

1. There have been determined the fundamental groups of the temporal data in the information systems of the Latvian Railway; the interconnection of these data and the issues under consideration have been examined. There have been defined the problems of employment of the temporal data in the operative information systems, including the following: the temporal data models imperfection; insufficient support of the data integrity; hardships of the data privacy provision; the absence of the efficient temporal data processing procedure; incompleteness of the processes of the data preparation to analysis.

2. There have been developed the series of the theoretical fundamentals in the field of the temporal databases, including: the new form of the temporal relation *Historical-Overlapped* introduction (relation with the crossed time segments); designing of the unique classification of the methods and models, subjected to the employment in the temporal systems under consideration; the formal characterisation of the temporal integrity restrictions for the relation temporal forms (the condition of the avoiding crossing of the time segments).

3. There have been suggested the fundamentals of the temporal databases designing in the relation environment, employing the temporal data model with the object abstract identifier and the temporal logic.

4. There have been designed the unique models and methods, oriented on the operation with the temporal objects with account for the railway transport tasks peculiarities:

- temporal relation model with the crossed lifespans;
- temporal object model with the periodicity and the specific calendar;
- method of providing the time environment for the user;
- methods of determining the temporal object active version, including: the logical rules method and the temporal elements method;
- method of the interval data transformation;
- method of the temporal data protection with consideration for the enhanced railway information systems requirements for the data integrity, privacy and accessibility.

5. The correctness of the proposed models and methods has been tested in various tasks of the information-analytical systems of the Latvian Railway, primarily in the train scheduling system. The results obtained confirm the validity of the models and the applicability of the methods.

6. The investigation results have allowed the development of the interactive information system for the passenger trains scheduling for the Latvian Railway, as well as achievement of the growing security of the temporal data of the passenger

statements processing system and upgrading the centralized database of the railway classifiers and codifiers. The proposed system of the scheduling support can serve as a basis of the railway transportation information system, but not only; it can be also used as the part of the management system, allowing the trains schedule simulation (by the way of automatic generation of recommendations for the analyst about the trains transfer on another day of the week after assigning the new day-off); it can be a part of the analytical system, supposed for the passenger traffic analysis and forecasting.

7. The research results have been approved in the Latvian Railway information systems. However, they can be implemented in the information systems of other railway companies. Moreover in many cases the obtained results have the universal nature and can be applied in temporal information systems regardless their destination.

8. The investigation results are employed in the process of lecturing in the courses “Databases and Databases Systems” (Study programme “Bachelor of Natural Sciences in Computer Science”) and “Databases Modern Technologies” (Study programme “Master of Natural Sciences in Computer Science”) in Transport and Telecommunication Institute.

LIST OF THE AUTHOR’S PUBLISHED WORKS

Papers

1. Kopytov E., Petoukhova N., Demidov V. Methodology of Huge Data Volume Processing System Development for Analysis of Latvian Railway Passengers Transportation. In: *Proceedings of VI International Conference “TransBaltica 2001”*, June 7-8, 2001, Riga, Latvia. pp. 201-208. (in Russian)
2. Petukhova N. Methodology of the System Development for Huge Data Volume Processing. In: *Informācijas tehnoloģija: zinības un prakse. VI konference. Konferences materiāli.* Riga: DATI, 2001, pp. 65-70. (in Russian)
3. Kopytov E., Demidovs V., Petukhova N. Use of Temporal Databases Principles in Information Systems of the Latvian Railway. In: Proceedings of *VII International Conference “TransBaltica 2002”*, June 12-14, 2002, Riga, Latvia, pp. 183-190. (in Russian)
4. Petukhova N. Method of Access Provision to Relational Systems Data on Relation Row Level, *Transport and Telecommunication*, Vol. 4, No 1, 2003, pp. 45-52. (in Russian)
5. Kopitovs J., Demidovs V., Petukhova N. Method of Temporal Databases design Using Relational Environment. In: *Scientific proceedings of Riga Technical University - Computer Science: Applied Computer Systems*, Series #5, Issue #13. Riga: RTU, 2002. pp. 236-246.

6. Kopitovs J., Demidovs V., Petukhova N. Virtual Models in Forecasting Systems of Railway Transportation. In: *The international conference: Modelling and Simulation of Business Systems (MOSIBUS 2003). May 13-14, 2003. Vilnius, Lithuania*. Kaunas: Kaunas University of Technology Press, 2003, pp. 265-268.

7. Petukhova N. Principles of Development of Complex Security Systems in Relational Databases on the Example of Latvian Railway. In: *Transport and Telecommunication*. Riga: Transport and Telecommunication Institute, 2004, Vol. 5, Nr. 2, pp. 36-44. (in Russian)

8. Kopytov E., Demidovs V., Petoukhova N. Principles of Creating Data Warehouses in Decision Support Systems of Railway Transport. In: *Computing Anticipatory Systems: CASYS 2003 - Sixth International Conference, Liège, Belgium, 11-16 August 2003. AIP Conference Proceedings*. Melville, New York: The American Institute of Physics, Vol.718, 2004, pp. 497-507.

9. Petukhova N. Development of a Complex Security System in Relational Databases for Railway Transport. In: *Scientific proceedings of Sixth International Baltic Conference on Computer Science and Information Technologies - Databases and Information Systems: DB&IS 2004. Scientific Papers University of Latvia. June 6-9, 2004*. Riga: LU, Vol.673, 2004, pp. 139-150.

10. Petukhova N. Problems of the Information Security Providing in the Temporal Databases. *Transport and Telecommunication*, Vol.7, No3, 2006, pp. 504-513. (in Russian)

11. Petukhova N. Investigation of Temporal Data Models in Information Systems on the Railway. In: *Proceedings of the 7th International Conference* "*Reliability and Statistics in Transportation and Communication" (RelStat'07). 24-27 October 2007. Rīga, Latvija*. Riga: TTI, 2007, pp.168-177.

12. Kopytov E., Demidovs V., Petukhova N. Modelling of railway schedule in Temporal Databases. In: *Proceedings of the International Conference "Modelling of Business, Industrial and Transport System" (MBITS'08). May 7-10, 2008, Riga, Latvia, Transport and Telecommunication Institute*. Riga: TTI, 2008, pp. 107-116.

13. Kopytov E., Demidovs V., Petukhova N. Method of Multiversion Railway Schedule Creation in Information Systems. In: *Proceedings of the 6-th International Scientific-Practical Conference" Research, Development and Application of High Technologies in Industry".* 16-18 October 2008. St. Petersburg, Russia, Polytechnic University, pp. 17-24. (in Russian)

14. Petukhova N. Temporal Models and Methods of Transport Information Systems. In: *Proceedings of the International Scientific Conference of Junior Researchers "Science – Lithuania's Future. TRANSPORT*". TECHNIKA, Vilnius Gediminas Technical University, 2008, pp. 319-325. (in Russian)

15. Kopytov E., Demidovs V., Petukhova N. Application of Temporal Elements in the Railway Schedule Systems. *Transport and Telecommunication*, 2008, Vol. 9, No 2, pp. 14-23.

16. Kopytov E., Petukhova N. Application of Temporal Databases in Modeling of Railway Processes. In: *Proceedings of the International Symposium on STOHASTIC MODELS in RELIABILITY ENGENEERING, LIFE SCIENCE and OPERATIONS MANAGEMENT (SMRLO'10). February 8-11, 2010, Beer Sheva, Israel*. Beer Sheva: SCE, 2010, pp. 586-594.

17. Kopytov E., Petukhova N., Demidovs V. Methods for Railway Schedule Periodicity Support in Temporal Databases. In: *Proceeding of 9th JOINT CONFERENCE ON KNOWLEDGE-BASED SOFTWARE ENGINEERING (JCKBSE'10). Kaunas, Lithuania, August 25-27, 2010.* Kaunas: Tehnologija, 2010, pp. 178-191.

Abstracts

18. Petoukhova N. Increase of Safety of Data in Relational Database for Records Level. In: *Science and Technology – the Step into the Future: Programme and Abstracts. Research-Practice and Educational-Methodological Conference, Riga, Latvia, May 2-3, 2002.* Riga: TTI, 2002, p. 26.

19. Petukhova N. Detailed Access to Data as a Tool of Increase of Information System Safety. *In: Programme and Abstracts of the International Conference "Reliability and Statistics in Transportation and Communication (RelStat'02)", October 17-18, 2002*. Riga, Latvia, 2002, pp. 73-74. (in Russian)

20. Kopitovs J., Demidovs V., Petukhova N. Method of Temporal Databases design Using Relational Environment. In: *Programme of the 43rd International Scientific Conference of Riga Technical University*. Riga: RTU. 2002. p. 19.

21. Kopitovs J., Demidovs V., Petukhova N. Principles of Creating Data Warehouses in Decision Support Systems of Railway Transport. In: *CASYS '03 – Sixth International Conference on Computing Anticipatory Systems*. *Liège, Belgium, 11-16 August 2003*. *Abstract Book.* Liège: CHAOS, 2003, Symposium 8, p.8.

22. Kopitovs J., Demidovs V., Petukhova N. Application of Virtual Data Models in Decision Support Systems. In: *Abstracts of the International Scientific and Technical Conference "Civil Aviation on the Modern Stage of Evolution of Science, Technics and Society"*. April 17-18, 2003, Moscow: MSTUCA, pp. 181-182. (in Russian)

23. Petuhova N. Principles of Development of Complex Security Systems in Relational Databases on the Example of Latvian Railway. In: *Proceedings of the International Conference "Reliability and Statistics in Transportation and Communication (RealStat'03*)", October 16-17, 2003. Riga: TTI, Latvia. 2003, pp.28-29. (in Russian)

24. Petukhova N. Relational Approach to Information Security in Temporal databases. In: *Programme and Abstracts of the 5th International Conference* "*Reliability and Statistics in Transportation and Communication (RelStat'05). 13-14 October 2005. Riga, Latvia*. Riga: TTI, 2005, pp. 62-63.

25. Petukhova N. The Problems of Information Security Management in Temporal Databases, *RESEARCH and TECHNOLOGY - STEP into the FUTURE*. Riga, Vol.1, No 4, 2006, pp. 115-116.

26. Kopytov E., Demidovs V., Petukhova N. Method of Railway Schedule Supporting in Temporal Databases. In: *Proceedings of the 5-th International Conference "Research, Development and Application of High Technologies in Industry" April 28-30, 2008, St. Petersburg, Russia*: St. Petersburg: Polytechnic University, 2008, pp. 78-79.

27. Kopytov E., Petukhova N. Application of Temporal Databases in Modeling of Railway Processes. In: *Book of Abstracts of the International Symposium on STOHASTIC MODELS in RELIABILITY ENGENEERING, LIFE SCIENCE and OPERATIONS MANAGEMENT (SMRLO'10). February 8-11, 2010, Beer Sheva, Israel, SCE – Shamoon College of Engeneering*, Beer Sheva: SCE, 2010, p. 135.

Regulatory Documents and Concepts

28. *Concept of Developing the Information System „Electronic Booking in the International Trains" (SAR-R)*, Riga: VAS LDz ISC, 2008, 18 p. (in Latvian)

29. *Requirements Specification of the Information System „Electronic Booking in the International Trains*" *(SAR-R)*, Riga: VAS LDz ISC, 2009, 43 p. (in Latvian)

30. *Technology of the Information System „Electronic Booking in the International Trains" (SAR-R)*, Riga: VAS LDz ISC, 2010, 50 p. (in Latvian)

31. *Requirements Specification of the Information System Modernisation „Registration of the Hardware and Software" (APNIS)*, Riga: VAS LDz ISC, 2008, 150 p. (in Latvian)

32. *Concept of Development of the Information System „Interactive Schedule of the Trains Traffic"*, Riga: VAS LDz ISC, 2007, 54 p. (in Latvian)