

Темпоральные модели базы данных и их свойства

© С.А. Тоноян, Д.В. Сараев

МГТУ им. Н.Э. Баумана, Москва 105005, Россия

Показано, что большинство современных баз данных (БД) хранит единственное — текущее — состояние объектов предметной области и относится к классу реляционных БД. Отмечено, что реляционные БД являются нетемпоральными и хранят последнее состояние объекта. Под влиянием внешних факторов эти объекты изменяются во времени, поэтому возникает необходимость регистрации различных их состояний. На основе анализа существующих моделей темпоральных БД и их основных свойств определены модели и подходы, позволяющие оптимально решать задачи обработки данных с изменяемой структурой в информационных системах. Выделен класс моделей, обеспечивающих преобразование нетемпоральных БД в темпоральные. Рассмотрены способы выполнения запроса к изменяющимся со времени данным и сохранения их целостности.

Ключевые слова: темпоральная база данных, реляционная база данных, SQL-запрос, многомерное пространство, темпоральные данные, нетемпоральные данные, отношение, оператор, множество, структура базы данных, структура данных.

Введение. Большинство современных баз данных (БД) является нетемпоральными и хранит единственное, как правило, самое последнее состояние объектов предметной области, что ограничивает область их применения. Существует множество предметных областей, в которых необходимо получать доступ не только к самому последнему состоянию БД, но и к прошлым и будущим ее состояниям [1–3].

Множество состояний объектов предметной области представляет собой совокупность снимков БД в определенный период времени. Такие БД и соответствующие им модели называются темпоральными и поддерживают операции изменения, которые переводят БД из одного состояния в другое, тем самым заменяя старые значения новыми.

В темпоральных БД присутствует атрибут времени, который позволяет перевести БД в состояние, соответствующее определенному периоду времени. В результате можно провести анализ БД за прошлые и текущий моменты времени с последующей их экстраполяцией в предстоящие периоды времени [3,4].

Основные понятия теории темпоральных БД. Поскольку в темпоральной БД хранится история изменений состояния объекта, то логично предположить, что объем информации в ней будет значи-

тельно больше, чем в нетемпоральной БД. Если же обратиться к количественной оценке информации, то однозначный ответ дать сложно, поскольку она будет сильно зависеть от способа организации темпоральности и частоты изменения информации [5–7].

Стремительный рост объема БД вследствие их темпоральности требует увеличения производительности системы и ставит перед исследователями задачу оптимизации способов хранения на уровне структур БД в целях уменьшения объема хранимой информации посредством минимизации ее избыточности. Избыточность состоит в том, что каждое изменение одного атрибута кортежа приводит к появлению нового кортежа, который описывает новое состояние моделируемого объекта. Универсальное решение этой проблемы в данный момент отсутствует. Один из способов — запись только изменившихся атрибутов кортежа, а для остальных — использование некоторого признака, показывающего, что атрибуты не изменились. Это приведет к сложности извлечения данных для получения информации о текущем состоянии объекта (придется проследить всю историю изменений).

Рассмотрим основные понятия теории темпоральных БД [4, 5, 7].

В темпоральных БД каждый кортеж содержит информацию о состоянии моделируемого объекта, а также о времени, когда эта информация была записана в БД. Такое размытие информации об одном логическом объекте по нескольким кортежам было названо *вертикальной* темпоральной аномалией [1,4]. Общую продолжительность периода времени, когда информация об объекте присутствует в БД независимо от всех происходящих с ним изменений, можно получить с помощью операции произвольного объединения. На первом этапе отбрасываются все атрибуты кортежей, содержащие различные значения, т. е. те атрибуты, которые вынуждают хранить информацию об объекте в нескольких кортежах. В результате чего получается несколько эквивалентных по значению кортежей, содержащих идентичные данные без учета временного параметра. На втором этапе вычисляется общая продолжительность для пересекающихся и последовательных периодов времени. Неопределенность данной операции состоит в том, что не всегда можно точно определить, какие атрибуты являются темпоральными, а какие нетемпоральными, т. е. непонятно, какие атрибуты необходимо исключить из запроса для получения требуемого результата.

В теории темпоральных БД существует также понятие сводимости к нетемпоральной форме. Пусть $M = (DS, QL, C)$ — нетемпоральная модель БД, а $M^T = (DS^T, QL^T, C^T)$ — темпоральная, где DS , QL , C — структуры данных, запросов и ограничений целостности.

Если db^T — экземпляр базы данных DST, то темпоральная операция Op сводится к нетемпоральной так:

$$\forall db^T \forall t: \phi^{M^T, M} t(op^T(db^T)) = Op(\phi^{M^T, M} t(db^T)),$$

а темпоральное ограничение — к нетемпоральному:

$$\forall db^T \forall t: \phi^{M^T, M} t(c^T(db^T)) = c(\phi^{M^T, M} t(db^T)),$$

где $c^T(db^T)$, $c(db)$ — операции вычисления ограничений при переходе между БД.

Наконец, темпоральная модель $M^V = (DS^V, QL^V)$ является полной по отношению к нетемпоральной модели $M = (DS, QL)$ тогда и только тогда, когда выполняются все условия темпоральности [4, 5, 7].

Способы реализации темпоральных БД. При описании темпоральной БД необходимо определиться с дискретизацией времени, которая показывает, насколько отличаются между собой соседние моменты времени. Речь идет о представлении времени, в пределах которого формировались запросы или отчеты, когда данные не меняются, а на определенном отрезке времени мы получаем некоторый их срез, что приводит к необходимости использовать точечное представление.

Существует несколько способов реализации темпоральных БД [3, 8, 9]:

1) создание «с нуля», т. е. самостоятельное создание модели БД. Способ не получил своего развития ввиду быстрого расширения функций БД и их свойств;

2) преобразование на уровне ядра реляционной БД, когда реализованы расширения синтаксиса языка, проверка ошибок и оптимизация. Способ удобен для пользовательских приложений, однако доступен только для разработчиков;

3) выделение модулей преобразования темпоральных запросов к БД. Здесь вместо реляционной надстройки используется темпоральная абстракция, которая интерпретирует результаты запросов, позволяет снизить число ошибок и отделить логику от технической реализации хранения данных;

4) создание между пользовательским приложением и БД некоторого промежуточного уровня в виде драйвера, сервиса, внешней обработки, библиотеки. Для пользовательского уровня промежуточный уровень является темпоральной БД, а для реляционных БД — приложением.

Общий вид многоуровневой архитектуры реализации системы управления базы данных (СУБД) приведен на рис. 1.

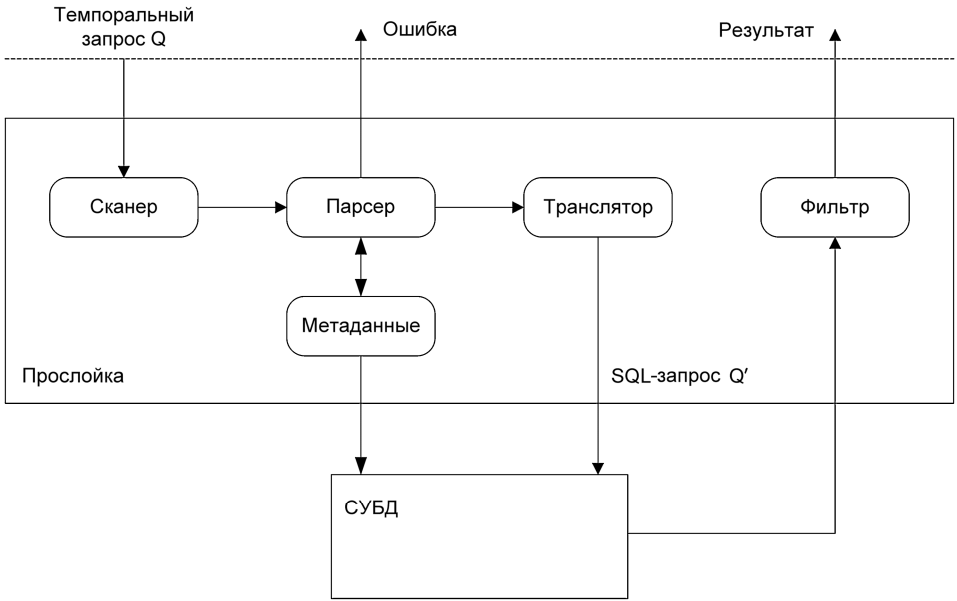


Рис. 1. Общий вид многоуровневой архитектуры темпоральной СУБД

Основные недостатки этих способов следующие [4, 10, 11]: изменение кода СУБД при применении второго и третьего способов, вследствие чего они доступны только разработчикам темпоральных БД;

невозможность доступа к предоставляемой и хранящейся информации и ее оптимизации при применении третьего и четвертого способов.

Основное их преимущество — во всех способах использованы уже существующие СУБД, которые только расширены благодаря темпоральной конструкции.

Следует подчеркнуть, что четвертый способ не зависит от конкретной СУБД и синтаксис SQL-запроса обычно остается прежним, поэтому обновления версии СУБД не влияют на работоспособность БД.

При реализации темпоральной СУБД на уровне приложения (рис. 2) единственным темпоральным параметром является атрибут «Дата». Темпоральная семантика языка должна быть вставлена в БД самим разработчиком кода. Вся дополнительная поддержка возлагается на разработчика, так как отсутствует поддержка со стороны СУБД. Для каждой такой БД часть кода темпоральных данных, которые появляются под влиянием разных факторов реального мира, разрабатывают заново. Вследствие этого отсутствует единая стандартизация обработки данных, что приводит к доработке и изменению кода программы, неэффективности и сложности поддержки информационной системы [5, 12].

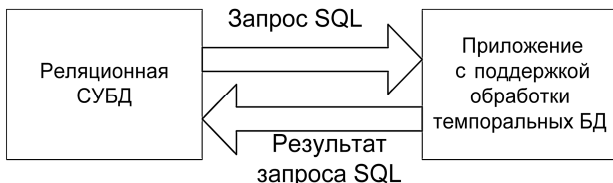


Рис. 2. Схема реализации темпоральной СУБД на уровне приложения

При расширении нетемпоральных данных до темпоральных последние можно описать, используя специальные временные атрибуты, языки запросов и языки алгебры для реляционных СУБД (рис. 3). В этом случае темпоральность достигается через надстройку над реляционной СУБД. Данный способ чаще всего используют при реализации темпоральной СУБД. Его преимущество — изменение нетемпоральных данных, недостаток — темпоральные данные наследуют нетемпоральную составляющую исходных данных (ее ограничения).

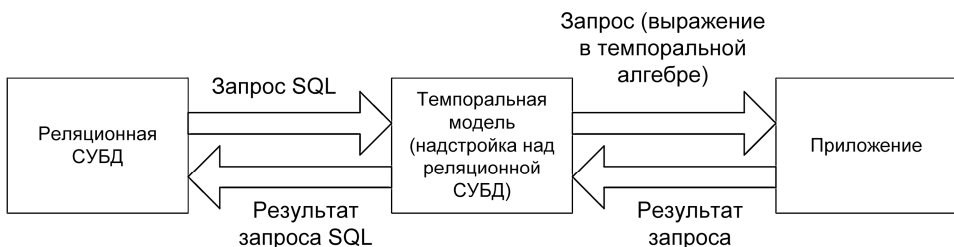


Рис. 3. Схема расширения реляционной СУБД до темпоральной

В настоящее время выделяют три основных типа темпоральных данных [3, 4, 11]:

- момент времени — событие, которое произошло или произойдет в определенный момент;
- интервал времени — длительность временного отрезка;
- период времени — конкретный отрезок времени.

Следует отметить, что запросы к темпоральным данным более сложные, чем к нетемпоральным. В то же время второй тип данных не поддерживает первый, что приводит к использованию двух дискретных моментов времени.

Модели представления темпоральных данных. Рассмотрим основные модели [1–3, 13–16].

Модель Снодграса. Имеется битемпоральное отношение $R(A_1, \dots, A_n, T)$, где A_1, \dots, A_n — набор атрибутов, T — битемпоральный атрибут. Тогда R можно представить в виде $R = (A_1, \dots, A_n, T_s, T_e, V_s, V_e)$, где T_s, T_e, V_s, V_e — атомарные темпоральные атрибуты, содержащие дату начала и окончания транзакционного и модельного времени.

Данное представление является естественным и часто используемым способом представления битемпоральных отношений.

Модель Джессена. Главной особенностью этой модели является то, что кортежи доступны только для чтения (т. е. никогда не обновляются). Такое представление данных хорошо подходит для основанного на архивах хранения битемпоральных отношений. Битемпоральное отношение R с набором атрибутов A_1, \dots, A_n может быть представлено в следующем виде: $R = (A_1, \dots, A_n, V_s, V_e, T, Op)$. Как и в предыдущей модели, атрибуты V_s и V_e хранят даты начала и окончания актуальности факта в моделируемой реальности соответственно, атрибут T — информацию о времени внесения кортежа в журнал изменений. Запросы на создание и удаление кортежей обозначаются в атрибуте Op соответствующими символами I (вставка) и D (удаление). Модификации данных представляют собой пару запросов (удаление и создание записи) с одинаковым атрибутом T .

Модель Гадия. Данный подход предполагает наличие битемпоральных меток у каждого из атрибутов кортежа, что обеспечивает возможность более гибкого моделирования реальности. Дано битемпоральное отношение $R(A_1, \dots, A_n, T)$, где T — атрибут, определенный на множестве битемпоральных элементов. Тогда битемпоральное отношение R может быть представлено в виде отношений, где каждый из атрибутов имеет свою темпоральную метку: $R = \{([T_s, T_e][V_s, V_e]A_1)\}, \dots, \{([T_s, T_e][V_s, V_e]A_n)\}$.

Кортеж состоит из n элементов. Каждый элемент представляет собой тройку значений: транзакционное время $[T_s, T_e]$, модельное время $[V_s, V_e]$ и значение атрибута A_i .

Модель МакКензи. В данной модели битемпоральное отношение — это последовательность состояний в модельном времени, проиндексированная транзакционным временем. В кортежах с модельным временем атрибуты имеют свои темпоральные метки. Битемпоральное отношение R с набором атрибутов A_1, \dots, A_n может быть представлено в виде отношения, в котором каждый атрибут помечается временной меткой: $R = (VR, T)$, где VR — отношение в модельном времени; T — транзакционное время. Схема состояний отношения модельного времени имеет вид $VR = (A_1V_1, \dots, A_nV_n)$, где A_1, \dots, A_n — набор атрибутов; V_1, V_n — атрибут модельного времени, каждый из которых соответствует атрибутам A_1, \dots, A_n и обозначает время актуальности его значения в моделируемой реальности.

Модель Бен-Зви. Пусть битемпоральное отношение R состоит из набора атрибутов A_1, \dots, A_n, T , где T — темпоральный атрибут, определенный на множестве битемпоральных элементов. Тогда R может быть представлено следующим образом: $R = (A_1, \dots, A_n, T_{es}, T_{rs}, T_{ee}, T_{re}, T_d)$, где T_{es} — атрибут времени, когда значение атрибута кортежа стано-

вится актуальным; T_{rs} — атрибут, хранящий информацию о том, когда атрибут T_{es} был сохранен в БД; T_{re} — атрибут, хранящий информацию о том, когда факт перестает быть актуальным в моделируемой реальности; T_{ee} — атрибут времени, когда T_{re} было зафиксировано в БД; T_d — атрибут, указывающий на время, когда запись была логически удалена из БД.

Отметим, что темпоральные модели БД могут иметь дополнительные критерии, например возможность работы с ошибочно введенными данными.

Проектирование темпоральных баз данных. В настоящее время существует множество средств для проектирования темпоральных БД, однако отсутствует единый механизм их реализации, что связано с разными типами темпоральных параметров. Рассматривая при проектировании темпоральной БД в качестве темпорального атрибута время, количество таблиц в БД будет зависеть от его типа. Если атрибутом является действительное время, то количество таблиц будет определяться приложением, а если транзакционное время — то объемом накопленной информации. Проектирование таблиц с поддержкой действительного времени достигается путем дополнения таблиц интервалами времени, что приводит к увеличению числа связей между таблицами. Проектирование темпоральных БД посредством расширения реляционных БД приводит к расширению реляционных ключей таблиц до верхней границы интервала транзакционного и (или) действительного времени. При этом необходимо обеспечить ограничения целостности БД [4, 11, 12].

Одним из основных показателей темпоральных БД в процессе эксплуатации является эффективность СУБД, заданная на этапе проектирования приложений для обеспечения запаса производительности, так как постоянный рост БД приводит к уменьшению быстродействия СУБД.

Одним из способов оптимизации темпоральных приложений является использование индексов. При добавлении дополнительных специальных столбцов для интервалов транзакционного времени необходимо задать верхний предел всем уникальным индексам. В этом случае возникает необходимость учета расположения основных и темпоральных столбцов в таблицах. При расположении темпоральных столбцов после основных запрос направлен к простой реляционной таблице, так как во всех операциях система сначала обращается к начальным столбцам, а затем уже к столбцам, отвечающим за время. В противном случае все операции будут зависеть от ограничений.

На практике чаще используются выборки из нескольких таблиц. Следует помнить, что при соединении нескольких таблиц алгоритм

запроса будет строиться на нестрогих неравенствах либо выполняться темпоральным ядром путем распаковывания и упаковывания темпоральных интервалов. Следовательно, оптимизация преобразований темпоральных запросов будет происходить автоматически. Запросы, создаваемые разработчиками, не дают значительного выигрыша во времени.

Выводы. Оперативная информация, поступающая из различных источников, фильтруется, интегрируется и складывается в реляционное хранилище БД. В силу внешних факторов накопленная там информация носит темпоральный характер. Это приводит к необходимости кардинально доработать, а иногда и разработать с «нуля» код поддержки БД и приложений. Разработка средств поддержки темпоральной БД позволяет в требуемый период времени извлекать данные из базы и обрабатывать, при этом не нужно кардинально дорабатывать существующую СУБД. Рассмотренные модели темпоральной БД и способы их реализации не являются универсальными, однако могут быть использованы в конкретных условиях в качестве основы для долгосрочной эксплуатации СУБД. На данный момент из нескольких десятков моделей темпоральных БД лишь небольшая часть используется на практике, что подчеркивает сложность их реализации.

ЛИТЕРАТУРА

- [1] Балдин А.В., Тоноян С.А., Елисеев Д.В. Язык запросов к миварному представлению реляционных баз данных, содержащих архив информации из предыдущих кадровых систем. *Инженерный журнал: наука и инновации*, 2013, вып. 11. URL: <http://engjournal.ru/catalog/it/hidden/1053.html>
- [2] Балдин А.В., Тоноян С.А., Елисеев Д.В. Обработка архива кадровых данных средствами 1С. *Инженерный журнал: наука и инновации*, 2013, вып. 11. URL: <http://engjournal.ru/catalog/it/hidden/1009.html>
- [3] Тоноян С.А., Балдин А.В., Елисеев Д.В. Методика модернизации стандартных модулей типовой конфигурации на базе технологической платформы «1С: Предприятие 8» с минимальными доработками. *Наука и образование. Электронное научно-техническое издание*, 2012, № 08. URL: <http://technomag.edu.ru/>. DOI: 10.7463/0812.0450231
- [4] Елисеев Д.В. *Методика обработки темпоральной реляционной базы данных в миварном пространстве*. Дис. ... канд. техн. наук. Москва, 2011, 149 с.
- [5] Федоров И.Б., Черненький В.М., ред. *Информационная управляющая система МГТУ им. Н.Э. Баумана «Электронный Университет»: концепция и реализация*. Москва, Изд-во МГТУ им. Н.Э. Баумана, 2009, с. 304–325.
- [6] Виноградова М.В., Игушев Э.Г. Конструктор баз данных на основе сущностей и их реквизитов с возможностью нормализации. *Наука и образование. Электронное научно-техническое издание*, 2012, № 01. URL: <http://technomag.edu.ru/>. DOI: 77-30569/242645
- [7] Елисеев Д.В., Балдин А.В., Тоноян С.А. Анализ использования типовой конфигурации «1С: Зарплата и кадры бюджетного учреждения 8» в вузах России. *Новые информационные технологии в образовании. Сб. науч. тр. 12-й Междунар. науч.-практ. конф. «Формирование новой информацион-*

- ной среды образовательного учреждения с использованием технологий ИС». Т. 2. Москва, ИС-Пабблишинг, 2012, с. 54–59.
- [8] Тоноян С.А., Тимофеев В. Б., Черненький С. В. Анализ и выбор конфигурации сети для финансово-экономической деятельности МГТУ им. Н.Э. Баумана на базе платформы «ИС:Предприятие 8». *Инженерный журнал: наука и инновации*, 2012, № 3. URL: <http://engjournal.ru/articles/110/110.pdf>
- [9] Балдин А.В., Елисеев Д.В., Агаян К.Г. Обзор способов построения темпоральных систем на базе реляционных баз данных. *Наука и образование. Электронное научно-техническое издание*, 2012, № 07. URL: <http://technomag.edu.ru/>. DOI: 10.7463/0812.0441884
- [10] Черненький В. М., Гапанюк Ю. Е., Мавзютов А. А. Разработка комплексных биомедицинских информационных систем на основе адаптивных объектов. *Вестник МГТУ им. Н.Э. Баумана. Сер. Приборостроение*, 2011, № 3, с. 105–112.
- [11] Варламов О.О. *Эволюционные базы данных и знаний для адаптивного синтеза интеллектуальных систем. Миварное информационное пространство*. Москва, Радио и связь, 2002, 286 с.
- [12] Григорьев Ю. А. Алгоритм синтеза частично оптимальной схемы реляционной базы данных. *Наука и образование. Электронное научно-техническое издание*, 2012, № 01. URL: <http://technomag.edu.ru/>. DOI: 77-30569/294486
- [13] <http://techno-new.developer.stack.net/doc/441884.html> (дата обращения 03.07.2014).
- [14] <http://www.swsys.ru/index.php?page=article&id=2196> (дата обращения 03.07.2014).
- [15] <http://network-journal.mpei.ac.ru> (дата обращения 03.07.2014).
- [16] <http://pdt.vscs.ac.ru/?module=Articles&action=view&aid=436> (дата обращения 03.07.2014).

Статья поступила в редакцию 17.07.2014

Ссылку на эту статью просим оформлять следующим образом:

Тоноян С.А., Сараев Д.В. Темпоральные модели базы данных и их свойства. *Инженерный журнал: наука и инновации*, 2014, вып. 12. URL: <http://engjournal.ru/catalog/it/hidden/1333.html>

Тоноян Славик Анушаванович родился в 1955 г., окончил МГТУ им. Н.Э. Баумана в 1972 г. Канд. техн. наук, доцент кафедры «Системы обработки информации и управления» МГТУ им. Н.Э. Баумана. Автор более 30 научных работ. Область научных интересов: информационные технологии. e-mail: tonoyansl@mail.ru

Сараев Дмитрий Владимирович родился в 1992 г. Студент кафедры «Системы обработки информации и управления» МГТУ им. Н.Э. Баумана. Автор 2 научных работ. Область научных интересов: информационные технологии. e-mail: saraev_dmitriy@list.ru

Temporal database models and their properties

© S.A. Tonoyan, D.V. Saraev

Bauman Moscow State Technical University, Moscow, 105005, Russia

A comparative analysis among modern database systems shows that most of them hold the only state — the current one of the domain objects and belongs to the class of relational databases. It is noted that relational databases are non-temporal and they store the only state — the last one of the object. Under the influence of external factors, these objects change over time and it is necessary to register the various states of the objects. Based on the analysis of the existing models of temporal databases and their basic properties are defined models and approaches to optimal solution of the data processing problems with variable data structure in information systems. A class of models, which translate non-temporal databases to the temporal ones, is single out. The methods of the query to the time-varying data, and maintain their integrity were considered.

Keywords: temporal database, relational database, SQL query, a multidimensional space, temporal data, the non-temporal data, relationships, operators, sets, structure of database, data structure.

REFERENCES

- [1] Baldin A.V., Tonoyan S.A., Eliseev D.V. *Inzhenernyi zhurnal: nauka i innovatsii — Engineering Journal: Science and Innovation*, 2013, iss. 11. Available at: <http://engjournal.ru/catalog/it/hidden/1053.html>
- [2] Baldin A.V., Tonoyan S.A., Eliseev D.V. *Inzhenernyi zhurnal: nauka i innovatsii — Engineering Journal: Science and Innovation*, 2013, iss. 11. Available at: <http://engjournal.ru/catalog/it/hidden/1009.html>
- [3] Tonoyan S.A., Baldin A.V., Eliseev D.V. *Nauka i obrazovanie. Elektronnoe nauchno-tehnicheskoe izdanie — Science and Education. Electronic Scientific and Technical Journal*, 2012, no. 8. Available at: <http://technomag.edu.ru/> DOI: 10.7463/0812.0450231
- [4] Eliseev D.V. *Metodika obrabotki temporalnoy relyatsionnoy bazy dannykh v mivarnom prostranstve* [Processing technique of temporal relational database in mivar space]. Ph.D. (Eng.) thesis. Moscow, 2011, 149 p.
- [5] Fedorov I.B., Chernenkiy V.M., eds. *Informatsionnaya upravlyayuschaya Sistema MGTU im. N.E. Baumana «Elektronnyi universitet»: kontseptsiya i realizatsiya* [BMSTU Information management system «Electronic University»: The conception and realization]. Moscow, BMSTU Publ., 2009, pp. 304–325.
- [6] Vinogradov M.V., Igshev E.G. *Nauka i obrazovanie. Elektronnoe nauchno-tehnicheskoe izdanie - Science and Education. Electronic Scientific and Technical Journal*, 2012, no. 1. Available at: <http://technomag.edu.ru/>. DOI: 77-30569/242645
- [7] Eliseev D.V., Baldin A.V., Tonoyan S.A. *Analiz ispolzovaniya tipovoy konfiguratsii «1C: Zarplata i kadry byudzhnogo uchrezhdeniya 8» v vuzakh Rossii.* [Analysis of the use of the typical configuration of «1C: Salary and Personnel of a budget entity 8» in Universities of Russia]. *Novye informatsionnye tekhnologii v obrazovanii. Sbornik nauchno-prakticheskoy konferentsii «Formirovanie novoy informatsionnoy sredy obrazovatel'nogo uchrezhdeniya s ispolzovaniem*

- tekhnologiy IC»* [New information technologies in education. Coll. scientific works of the 12th Int. scientific and practical Conf. «Formation of a new information environment of educational institutions using IC technologies»]. Vol. 2. Moscow, IC Publishing, 2012, pp. 54–59.
- [8] Tonoyan S.A., Timofeev V.B., Chernenkiy V.M. *Inzhenernyi zhurnal: nauka i innovatsii — Engineering Journal: Science and Innovation*, 2012, iss. 3. Available at: <http://engjournal.ru/articles/110/110.pdf>
- [9] Baldin A.V., Eliseev D.V., Agayan K.G. *Nauka i obrazovanie. Elektronnoe nauchno-tehnicheskoe izdanie - Science and Education. Electronic Scientific and Technical Journal*, 2012, no. 7. Available at: <http://technomag.edu.ru/>. DOI: 10.7463/0812.0441884
- [10] Chernenkiy V.M., Gapanyuk Yu.E., Mavzyutov A.A. *Vestnik MGTU im. N.E. Baumana. Priborostroenie — Herald of the Bauman Moscow State Technical University. Series: Instrument Engineering*, 2011, no. 3, pp. 105–112.
- [11] Varlamov O.O. *Evolutsionnye bazy dannykh i znaniy dlya adaptivnogo sinteza intellektualnykh system. Mivarnoe informatsionnoe prostranstvo* [Evolutionary data and knowledge bases for adaptive synthesis of intelligent systems. Mivar information space]. Moscow, Radio i svyaz Publ., 2002, 286 p.
- [12] Grigoryev Yu.A. *Nauka i obrazovanie. Elektronnoe nauchno-tehnicheskoe izdanie — Science and Education. Electronic Scientific and Technical Journal*, 2012, no. 1. Available at: <http://technomag.edu.ru/>. DOI: 77-30569/294486
- [13] <http://techno-new.developer.stack.net/doc/441884.html> (accessed on 03.07.214).
- [14] <http://www.swsys.ru/index.php?page=article&id=2196> (accessed on 03.07.214).
- [15] <http://network-journal.mpei.ac.ru> (accessed on 03.07.214).
- [16] <http://pdt.vscs.ac.ru/?module=Articles&action=view&aid=436> (accessed on 03.07.214).

Tonoyan S.A. (b. 1955) graduated from Bauman Moscow Higher Technical School in 1972. Ph.D., assoc. professor of the Information Processing and Control Systems Department at the Bauman Moscow State Technical University. Author of more than 30 publications in the field of computer and information technologies. e-mail: tonoyansl@mail.ru

Saraev D.V. (b. 1992) is a student of the Information Processing and Control Systems Department at the Bauman Moscow State Technical University. Author of two publications in the field of computer and information technologies. e-mail: saraev_dmitriy@list.ru