В. Л. Малых, А. Н. Калинин, Т. Ш. Юсуфов
**Объектно-реляционный подход к построению
хранилища данных**

Аннотация. В работе концептуально предлагается объектно-реляционный подход к построению хранилища данных. Подход альтернативен традиционным подходам к проектированию реляционных БД. Предлагается расширить реляционную БД за счет реализации в ней NoSQL хранилища типа ключ-значение. Предлагается методология проектирования объектно-реляционного хранилища. Предлагается другая точка зрения на проблему связности, целостности и нормализации данных в реляционной БД. Подход во многом основан на авторском опыте проектирования реляционных БД для медицинских информационных систем.

Ключевые слова и фразы: реляционная база данных, NOSQL хранилище данных, эволюционные базы данных, медицинские информационные системы.

Введение

Проблема адаптации структур хранения данных ко вновь возникающим требованиям к содержанию и структуре данных в базе данных (БД) известна давно. Особенно остро эта проблема стоит в предметной области медицинских информационных систем (МИС). В первую очередь это связано с широтой предметной области, с постоянным ростом медицинских знаний и отражением этих новых знаний в БД МИС. Столкнувшись с проблемой формализации многочисленных медицинских документов, в свое время один из авторов статьи сделал вывод о необходимости включения механизма концептуализации новых данных непосредственно в МИС [1]. Дальнейшее развитие и эксплуатация МИС полностью подтвердили этот вывод [2]. Решение было найдено с помощью включения в БД МИС моделей медицинских документов вместе со средствами создания и контролируемого исторического версионного изменения таких моделей, с возможностями разборки

моделей данных экземпляров документов и выгрузки разобранных данных в реляционные таблицы. Подход к адаптации, основанный на моделях (model driven approach) оказался весьма плодотворным, но обычно он применяется только лишь в части медицинских документов. В БД МИС проектируются и создаются в достаточно большом количестве реляционные таблицы (порядка 10^3 таблиц на полнофункциональную МИС), описывающие фактографические данные лечебно-диагностического процесса и смежных для медицинской организации (МО) деловых областей (управление кадрами, экономический и финансовый учет, материальное снабжение и учет, ресурсы МО и их использование, бизнес аналитика и т.д.). И проблема эволюции и адаптации реляционной модели данных МИС продолжала остро стоять. Предлагаемые подходы к решению этой проблемы можно в основном свести к следующим: а) выполнять постоянный рефакторинг БД в соответствии с изменением требований [3, 4]; б) пытаться строить универсальные структуры данных [5, 6]. Первый путь оказывается очень затратным, он приводит к большим затратам труда на ручное изменение модели данных в многочисленных экземплярах БД, и не решает проблему согласованности структур данных в различных экземплярах МИС. Второй путь представляется ограниченно применимым к МИС в силу гетерогенности, широты и постоянной эволюции предметной области.

В работе предлагается срединный путь решения проблемы, заключающийся в том, что эволюция и изменения структур данных сводятся к изменениям и вводу в БД новых моделей данных (метаинформации) на фоне достаточно зафиксированных (устойчивых) реляционных структур. Вновь концептуализируемые данные и изменения эволюционирующих данных предполагается описывать с помощью моделей, а сами данные должны будут продолжать храниться в устойчивых реляционных структурах. Срединный путь сочетает в себе как реляционный, так и объектный подходы. Такой дуалистический объектно-реляционный подход уже однажды очень успешно сработал [7]. Можно считать данную работу продолжением и расширением указанного подхода.

Работа также продолжает исследования [8, 9] в области поиска подходов к построению и реализации модели данных современных

медицинских информационных систем, проводимые в Исследовательском центре медицинской информатики ИПС им. А.К. Айламазяна РАН. Освещаются аспекты, не затронутые работой [9], в первую очередь вводятся в рассмотрение практические принципы проектирования объектно-реляционного хранилища данных, предлагается методология проектирования, приводятся примеры реального применения принципов и методологии.

1. Принципы проектирования

Традиционный широко применяемый подход к проектированию объектов БД (таблиц и представлений) опирается на ряд известных принципов:

- (1) абстрагирование (выделение сущностей и их атрибутов);
- (2) выделение отношений на сущностях;
- (3) поддержка связности и целостности данных в первую очередь за счет введения реляционных связей и прочих ограничений;
- (4) нормализация данных (исключение избыточного дублирования данных).

Все эти принципы выглядят достаточно бесспорными и им стараются следовать. Однако, при проектировании реляционного хранилища эволюционирующих данных для большой сложной развивающейся системы на передний план могут выходить совершенно другие принципы проектирования. Начнем их излагать по порядку.

1.1. Устойчивость реляционной модели хранилища данных

Все разработчики МИС знают, что конкретные экземпляры МИС, установленные у клиентов, неминуемо начинают эволюционировать. Появляется множество частных требований по развитию и доработке системы, которые затрагивают, в том числе, и реляционную БД — интеграционное ядро системы. Требования по модификации структуры БД не всегда настолько общие, что их следует учесть во всех экземплярах БД. В результате по известным рекомендуемым методикам [3, 4] производится частная модификация структуры БД в отдельных экземплярах системы. Вместо единой реляционной модели

Рис. 1. Эволюционное дерево экземпляров реляционной БД

данных получаем эволюционное дерево таких моделей, см. рис. 1. Работать не с одной моделью данных, но с целым деревом моделей становится весьма проблематичным. Часто возникают задачи подъема версии системы/подсистемы в каком-либо из узлов дерева до версии системы/подсистемы в другом узле дерева. Чтобы решать эти задачи либо приходится поддерживать достаточно сложную структуру контроля изменений экземпляров БД [4], либо приходится прибегать к попарному сравнению таблиц в соответствующих схемах БД, выделению различий и формированию скриптов на изменение структуры таблиц в узле. Оба этих решения достаточно трудоемки.

Устойчивость реляционной модели данных хранилища чрезвычайно важна также с точки зрения возможностей распределения хранилища по множеству узлов сети, решения проблемы многокомпонентности данных МИС [10]. Далее мы коснемся этого вопроса.

Поскольку проблема динамической концептуализации новых данных и их введения в различные экземпляры БД системы никуда не девается, то необходимо предложить ее решение, не нарушающее сформулированный принцип устойчивости реляционной модели.

1.2. Реализация в реляционной БД хранилища типа ключ-значение

Второй основной принцип проектирования — реализация в реляционной БД хранилища типа ключ-значение (key-value store). Информ-

Динамически концептуализируемые новые данные предлагается хранить в хранилище типа ключ-значение. Простейшей моделью такого хранилища может служить таблица с атрибутами имя и значение. Пример реализации такого типа хранилища представлен на рис. 2.

В таблицах U_DEPOSITS, U_DOC_ATTRS, U_DIC_ATTRS, U_NOMENCLATURES хранятся различные кортежи данных произвольной ограниченной длины (до 24 элементов в кортеже в данном примере). Имена элементов данных хранятся отдельно и описываются в таблице U_MODELS. Эта таблица играет роль метаданных по отношению к вышеперечисленным таблицам. В ней же описываются типы элементов данных кортежей, возможные ограничения на значения элементов, а также, возможно, определяются домены значений элементов кортежей в виде SQL выражений.

Собственно, указанные пять таблиц реализуют хранилище типа ключ-значение для подсистемы материального учета МИС. Это хранилище позволяет динамически добавлять новые атрибуты в депозиты, первичные и производные документы материального учета, справочник номенклатуры, прочие справочники подсистемы материального учета. При этом реляционная модель подсистемы сохраняет свою устойчивость. Очевидно, что на таких же принципах могут быть разработаны хранилища типа ключ-значение и для других подсистем МИС.

Использование принципов организации NoSQL БД в реляционной БД позволяет одновременно воспользоваться преимуществами обоих подходов, не увеличивая при этом число используемых БД.

Заметна тенденция к переходу от непосредственного написания SQL выражений в реляционных БД к промежуточному языку манипуляции данными, который затем транслируется в исполняемый SQL [12, 13]. В предложенном подходе хорошо просматривается подобная возможность, сводящаяся к написанию SQL команд над объектными моделями данных с использованием семантически понятных имен из метамodelей данных, с последующей трансляцией команд в исполняемые SQL команды над универсальными абстрактными моделями данных. Такой подход к работе с псевдо-SQL позволит упростить создание и понимание кода разработчиками.

Методически проектирование подобной «дуалистической» БД может производиться следующим образом.

2. Методология моделирования

Предлагается для моделирования использовать диаграммы классов языка UML, обосновывая это следующим:

- (1) диаграммы сущность-связь полностью поглощаются диаграммами классов;
- (2) диаграммы классов имплементируются в модели данных, т.е. полностью исполняют роль диаграмм сущность-связь;
- (3) диаграммы классов имеют существенные преимущества по сравнению с диаграммами сущность-связь (больше различных типов отношений, возможность использовать сложные типы данных (классы) для типизации атрибутов, возможность функционального описания концептов).

Предлагаемая ниже методология моделирования будет существенно опираться на возможности диаграмм классов, в частности на отношение генерализации.

Методология моделирования:

- (1) В первую очередь создаются базовые классы (концепты) и отношения (БКО). При создании базовых классов модели данных МИС следует опираться на уже разработанные биомедицинские онтологии [14] и использовать накопленный опыт:
 - Basic Formal Ontology (BFO),
 - Common Anatomy Reference Ontology (CARO),
 - Environment Ontology (EnvO),
 - Foundational Model of Anatomy (FMA),
 - Infectious Disease Ontology (IDO),
 - Ontology for Biomedical Investigations (OBI),
 - Ontology for Clinical Investigations (OCI),
 - Phenotypic Quality Ontology (PATO),
 - Relation Ontology (RO),
 - SNOMED,
 - Unified Medical Language System,

- National Cancer Institute Thesaurus,
 - HL7 Reference Information Model,
 - International Classification of Functioning Disability and Health,
 - и др.
- (2) Все моделируемые классы и отношения обязательно ссылаются через отношения генерализации на базовые классы или на их, более абстрактных по отношению к моделируемому классу, потомков. Это категорическое требование позволит получить связную иерархическую модель концептов предметной области. Сущности не должны рождаться без предварительного анализа, без осмысления их связей с другими уже существующими концептами, простым волевым путем. Недостатки хаотического проектирования сущностей всегда очевидны и легко заметны. Сформулированное категорическое требование ставит барьер на пути хаотического бессистемного проектирования.
- (3) Моделирование ведется безо всяких требований и всякого стеснения со стороны реализации. Важно получить ясную картину мира, и лишь потом перейти к проектированию реляционных таблиц, к диаграммам данных.
- (4) Разработка БКО и их ближайших классов наследников модели ведется ограниченными силами самых компетентных разработчиков. И только после этого можно попробовать двинуть разработку «в массы». Все разработанные классы и диаграммы имеют компетентных владельцев, несущих за них персональную ответственность. Изменения в классы и диаграммы вносятся только их владельцами.
- (5) Диаграммы строятся на основе уже определенных классов, и только при явном их дефиците предлагается ввести новый класс с обязательным происхождением от уже имеющихся классов. Рождение нового класса согласуется с владельцами классов, от которых новый класс производится. Может оказаться, что вместо создания нового класса разумно расширить уже имеющиеся, чтобы они удовлетворяли требованиям к новому классу. Таким образом, в методологию закладывается бритва Оккама, дабы без нужды не умножать сущности.

- (6) Далее базовые классы проектируются в «статические» реляционные структуры, соответствующие основному принципу проектирования.
- (7) Наследуемые от базовых расширенные специализированные классы реализуются в реляционной БД с помощью реляционных моделей базовых классов, которые дополняются частными хранилищами типа ключ-значение.

На рис. 3 приведен пример проектирования базовых классов.

3. Принципы проектирования (продолжение)

Заметим, что принцип устойчивости реляционной модели данных не запрещает абсолютно модификацию моделей базовых классов. Он предполагает ее достаточно редкой, но возможной. Причем все такие модификации должны рассматриваться как выпуск новой версии модели данных и должны применяться ко всем экземплярам системы.

Продолжим обсуждение принципов проектирования и обратимся к проблеме связности, целостности и нормализации реляционных данных.

3.1. Принцип ослабления реляционной связности данных

Главным механизмом поддержки целостности данных в реляционной СУБД является введение отношений на таблицах в виде ограничений (constraints). Например, рассмотрим реляционные ссылки по ключам справочника номенклатуры подсистемы «Аптека» МИС. Справочник номенклатуры ссылается по ключам на следующие частные справочники — домены значений: Справочник торговых наименований, Справочник латинских наименований лекарственных средств, Страны, Фирмы производители, Справочник фармгрупп, АТХ классификатор, Справочник МНН, Справочник единиц измерения, Справочник товарных групп, Справочник условий хранения, и т.д. и т.п. В результате для того, чтобы показать одну позицию номенклатуры со всеми характеристиками приходится связывать множество таблиц (некоторые из них многократно). Например, в известной нам реляционной реализации фармсправочника РЛС от ведущего производителя информационного обеспечения для врачей и фармацевтов [15] насчитывается порядка 50 таблиц. Очевидно, что отказ от связи с доменами значений по ключу и использование непосредственно значений из доменов существенно бы упростил реализацию справочника номенклатуры, все характеристики одной позиции номенклатуры могли бы извлекаться за один акт чтения одной строки или из «статической» реляционной таблицы, или из хранилища типа ключ-значение. Можно пытаться решать эту проблему с помощью снимков (snapshots), что фактически означает сборку по SQL запросу номенклатурного

справочника и долговременное хранение этой сборки с периодическим обновлением. Куда проще сразу же заложить в реализацию уже «собранный» справочник номенклатуры без множества реляционных связей с доменами.

Достоинством реляционных связей считается то, что можно обновить всего одну запись в таблице и все другие таблицы, ссылающиеся на эту запись «получат» обновление по ссылке (по ключу). Но эта возможность не всегда является благом. Например, первичные документы материального учета ссылаются на справочник номенклатуры. При этом справочник номенклатуры потенциально доступен для редактирования пусть и ограниченной группе пользователей. Пользователям также предоставляются возможности слияния дубликатов в справочнике, что сводится к перестановке ссылок в документах с одной позиции номенклатурного справочника на другую. Ограничить пользователей в этой функциональности не представляется возможным. В результате будут возникать инциденты, когда после внесения изменений в справочник номенклатуры электронные документы будут «менять» свое содержание и начнут расходиться с бумажными оригиналами. Приходится отбросить нормализацию данных и вводить дополнительные таблицы, которые хранят все атрибуты номенклатуры документов, и которые позволяют вносить изменения в номенклатуру документов только в рамках отдельного документа. После этого реляционные ссылки документов на справочник номенклатуры фактически полностью утрачивают свое значение, а сам справочник начинает играть роль домена значений, сохраняемых в контексте каждого отдельного документа.

Фактически предлагается принять в качестве дополнительного принципа проектирования принцип ослабления реляционных связей, сводящийся к отказу от реляционных ссылок по ключу в пользу использования значений из доменов (неявных ссылок по значению). В этом случае в реляционной БД станут появляться отдельные изолированные таблицы (домены), на которые не будет формальных реляционных ссылок, но значения из этих таблиц будут присутствовать в других семантически связанных с доменами таблицах.

3.2. Принцип фрагментирования реляционной модели данных

Понижение связности реляционной модели данных необходимо, чтобы реализовать принцип фрагментирования (распределенности, мозаичности) реляционной модели данных. Этот принцип требует, чтобы реляционная модель данных подобно картине, собранной из отдельных элементов (мозаики), легко могла быть разделена на отдельные целостные фрагменты, соответствующие определенным подсистемам и компонентам системы. Подобные задачи многократно возникают, когда требуется установить у Заказчика только отдельную подсистему МИС, например, подсистему «Диетпитание» или подсистему «Аптека». Локализация подсистемы требует локализации необходимой для ее работы реляционной подмодели данных. «Потянув» за множество реляционных таблиц, имеющих прямое отношение к подсистеме, а часто эти таблицы выделяются благодаря нахождению в отдельной схеме, в силу имеющихся множественных реляционных связей этих таблиц с другими с учетом транзитивности таких связей, как сеть «вытаскивают» достаточно большой «улов» различных реляционных таблиц. В итоге часто приходится отказываться от локализации реляционной подмодели подсистемы, и вместо нее устанавливают полную реляционную модель системы, даже если большую часть полной модели не предполагается использовать.

Для решения этой проблемы предлагается использовать реляционные связи внутри отдельных связных целостных фрагментов реляционной БД и минимизировать реляционные связи между такими фрагментами. Например, вместо широко распространенных ссылок по ключам (локальным идентификаторам) на подразделения и должностных лиц перейти на семантические ссылки по значению, используя названия подразделений, должность и ФИО сотрудника. Естественно, что при использовании ссылок по значению предполагается строгость в ведении соответствующих доменов — источников значений для ссылок. Да, при этом возникает избыточность и дублирование данных, но это снижает связность реляционной модели и позволяет локализовывать и распределять данные по отдельным узлам информационной сети, а меж-узловые связи данных строить

функционально через источники данных, обращающихся к требуемым узлам сети.

Наличие таких возможностей у реляционного хранилища весьма актуально для МИС в связи с необходимостью автоматизировать целые сети связанных МО [9, 10], в связи с наблюдающимся перераспределением данных между различными уровнями системы автоматизации здравоохранения, такими как федеральный, региональный и госпитальный уровни, а также в связи с использованием облачных технологий. Данные должны «приспосабливаться» и «уметь жить» в распределенной гетерогенной среде.

3.3. Нормализация данных

Нормализация реляционной модели данных считается хорошим принципом проектирования, но при этом часто забывается, что реляционная нормализация вовсе «автоматически» не приводит к реальной нормализации данных. Пример из практики. В одной МО, где ведется материальный учет, номенклатурный справочник вырос до 55 тысяч позиций. При этом в нем довольно много дубликатов, некоторые позиции имеют свыше десятка дубликатов. Это происходит потому, что пользователи имеют возможность свободно вводить новые позиции в номенклатурный справочник, а также новые позиции в номенклатурном справочнике могут создаваться при автоматической обработке электронных накладных от поставщиков. Обе эти возможности по понятным причинам закрыть нельзя. Поэтому реальная нормализация данных в МИС зачастую должна выполняться на функциональном уровне и не средствами поддержки целостности реляционной БД. Поэтому, отказ от реляционной нормализации данных, предполагаемый в излагаемых принципах проектирования, не ухудшит проблему нормализации данных МИС. Основным эффектом отказа от нормализации станет рост размера БД, но при этом будут предоставлены возможности масштабирования за счет распределения БД.

3.4. Принцип интеграции данных

Этот принцип предполагает возможность переноса любых связанных целостных данных из одного экземпляра БД в любой другой. Фактически такая возможность обеспечивается принципом устойчивости модели объектно-реляционного хранилища данных. Необходимо только оговорить еще некоторые технические детали. Необходимо будет отказаться от использования локальных идентификаторов (ключей) и перейти к использованию глобальных идентификаторов, позволяющих избежать конфликтов нарушения целостности первичных ключей при переносе данных. Следует учесть, что при переносе NoSQL данных в случае реализации хранилищ таких данных по типу, представленному на рис. 2, необходимо будет также переносить метainформацию, описывающую переносимые данные.

Заключение

В данной концептуальной работе нет возможности обсуждать детальные аспекты реализации предложенных объектно-реляционных хранилищ. Например, не обсуждаются вопросы историчности и эволюции метаданных, не обсуждаются вопросы хранения в БД больших объектов (large objects) и их функциональная роль. Авторы надеются, что упущение деталей не помешает восприятию концепции в целом.

В работе рассмотрена концепция объектно-реляционного хранилища данных. Изложение концепции ведется применительно к МИС, но область применения описанного типа хранилища несколько не ограничена предметной областью МИС. Выдвинутые в работе принципы проектирования хранилища обусловлены взятыми из практики требованиями предметной области МИС. Основное требование состоит в необходимости постоянной концептуализации и ввода новых данных в модель данных. Представляется чрезвычайно важным уметь удовлетворять это требование, сохраняя при этом неизменность (устойчивость) реляционной модели данных. Решением проблемы может встать реализация средствами реляционного хранилища объектного NoSQL хранилища данных. В работе приводится прототип разработки подобной модели данных для подсистемы материального учета

МИС. Принципы фрагментарности реляционной модели и принцип интеграции данных позволяют распределять данные по множеству узлов информационной сети и интегрировать данные в любых узлах. Эти возможности закладывают основу для решения проблемы многокомпонентности и распределения данных по различным уровням здравоохранения.

***Благодарности.** Авторы искренне благодарны Дмитрию Владимировичу Бельшеву и Евгению Владимировичу Кочурову за интерес к работе и конструктивную критику.*

Список литературы

- [1] Я. И. Гулиев, В. Л. Малых. «Архитектура HL-X», *Программные системы: Теория и приложения*. Т. 2, Физматлит, М., 2004, с. 147–168. [↑] [169](#)
- [2] Я. И. Гулиев, В. Л. Малых. «Архитектура HL-X поддержки документов в медицинских информационных системах», *Информационно-управляющие системы*, 2009, №2, с. 63–69, URL: http://www.interin.ru/datas/documents/article_arc-hl-x.pdf [↑] [169](#)
- [3] *Modernizing IBM i Applications from the Database up to the User Interface and Everything in Between*, IBM Redbooks publication, 2014, URL: <http://www.redbooks.ibm.com/abstracts/sg248185.html?Open> [↑] [170](#), [171](#)
- [4] *Эволюционный дизайн баз данных*, 2016, URL: <https://habrahabr.ru/post/312970/> [↑] [170](#), [171](#), [172](#)
- [5] И. А. Микляев. «Формализованное описание основной структуры представления данных матричной универсальной объектно-реляционной базы данных», *Вестник АГТУ. Сер.: Управление, вычислительная техника и информатика*, 2001, №1, с. 61–68. [↑] [170](#)
- [6] О. О. Варламов. «Эволюционные базы данных и знаний. Миварное информационное пространство», *Известия ЮФУ. Технические науки*, 2007, №2, с. 77–81. [↑] [170](#)
- [7] В. Л. Малых, С. П. Пименов, М. И. Хаткевич. «Объектно-реляционный подход к созданию больших информационных систем», *Программные системы: Теоретические основы и приложения*, Наука. Физматлит, М., 1999, с. 177–184. [↑] [170](#)
- [8] Д. В. Бельшев, Е. В. Кочуров. «Анализ методов хранения данных в современных медицинских информационных системах», *Программные системы: теория и приложения*, 7:2(29) (2016), с. 85–103, URL: http://psta.pstiras.ru/read/psta2016_2_85-103.pdf [↑] [170](#)

- [9] Д. В. Бельшев, Е. В. Кочуров. «Перспективные методы работы с данными в медицинских информационных системах», *Программные системы: теория и приложения*, 7:3(30) (2016), с. 79–97, URL: http://psta.psiras.ru/read/psta2016_3_79-97.pdf ↑ ^{170,171,181}
- [10] С. И. Комаров, Д. В. Алимов. «Применение механизма многокомпонентности при информатизации крупного ЛПУ с филиалами», *Врач и информационные технологии*, 2016, №6, с. 25–33. ↑ ^{172,181}
- [11] *Реляционные базы данных обречены?* 2011, URL: <https://habrahabr.ru/post/103021/> ↑ ¹⁷³
- [12] Л. А. Пономаренко, С. С. Танянский. «Обработка данных произвольной структуры языковыми средствами реляционной модели», *УСuМ*, 2010, №6, с. 47–53. ↑ ¹⁷⁴
- [13] Л. А. Пономаренко, В. А. Филатов, С. С. Танянский. «Минимизация вычислительных затрат при доступе к данным в распределенных экономических системах поддержки принятия решений», *УСuМ*, 2014, №3, с. 74–80. ↑ ¹⁷⁴
- [14] В. Smith. *Introduction to Biomedical Ontologies*, 2008, URL: http://ontology.buffalo.edu/smith/BioOntology_Course.html ↑ ¹⁷⁵
- [15] Справочник лекарств РЛС, URL: <https://www.rlsnet.ru> ↑ ¹⁷⁸

Рекомендовал к публикации

к.т.н. Я. И. Гулиев

Пример ссылки на эту публикацию:

В. Л. Малых, А. Н. Калинин, Т. Ш. Юсуфов. «Объектно-реляционный подход к построению хранилища данных», *Программные системы: теория и приложения*, 2017, 8:3(34), с. 169–187.

URL: http://psta.psiras.ru/read/psta2017_3_169-187.pdf

Об авторах:

Владимир Леонидович Малых

Зав. лабораторией Института программных систем им. А. К. Айламазяна. Научные интересы: архитектура медицинских информационных систем; моделирование и разработка больших ИС. Data Science в области медицинской информатики; поддержка принятия врачебных решений; обучаемые интеллектуальные системы

e-mail:

mvl@interin.ru

Алексей Николаевич Калинин

М.н.с. Института программных систем им. А. К. Айламазяна. Научные интересы: архитектура медицинских информационных систем; моделирование и разработка больших ИС; экономика деловых процессов медицинской организации в части учета прямых материальных затрат

e-mail:

ank@interin.ru

Теймур Шукюрович Юсуфов

М.н.с. Института программных систем им. А. К. Айламазяна. Научные интересы: архитектура медицинских информационных систем; моделирование и разработка больших ИС; финансово-экономический учет в медицинской организации

e-mail:

tsy@interin.ru

Vladimir Malykh, Aleksey Kalinin, Teymur Yusufov. *Object-relational approach to building a data storage.*

ABSTRACT. The work offers a conceptual approach to building object-relational data storage. The approach is alternative to traditional methods of relational databases design. It is proposed to extend the relational database with NoSQL type storage implement (key-value). Design methodology for object-relational storage is proposed. An alternative view of the issue of connectivity, integrity and data normalization in relational databases, is described. The approach is largely based on the author's experience of designing relational database models for medical information systems. (*In Russian*).

Key words and phrases: relational database, NOSQL data storage, evolutionary database, medical information system.

References

- [1] Ya. I. Guliyev, V. L. Malykh. "HL-X Architecture", *Programmnyye sistemy: Teoriya i prilozheniya*. V. 2, Fizmatlit, M., 2004, pp. 147–168 (in Russian).
- [2] Ya. I. Guliyev, V. L. Malykh. "Document Support Architecture in Medical Information Systems", *Informatsionno-upravlyayushchiye sistemy*, 2009, no. 2, pp. 63–69 (in Russian), URL: http://www.interin.ru/datas/documents/article_arc-hl-x.pdf
- [3] *Modernizing IBM i Applications from the Database up to the User Interface and Everything in Between*, IBM Redbooks publication, 2014, URL: <http://www.redbooks.ibm.com/abstracts/sg248185.html?open>
- [4] *Evolutionary Design of Databases*, 2016 (in Russian), URL: <https://habrahabr.ru/post/312970/>
- [5] I. A. Miklyayev. "Formalized description of basic structure of representation of matrix universal object-relational databases", *Vestnik AGTU. Ser.: Upravleniye, vychislitel'naya tekhnika i informatika*, 2001, no. 1, pp. 61–68 (in Russian).
- [6] O. O. Varlamov. "The Evolutionary Bases of Data and Knowledge. Mivar Information Space", *Izvestiya YuFU. Tekhnicheskiye nauki*, 2007, no. 2, pp. 77–81 (in Russian).
- [7] V. L. Malykh, S. P. Pimenov, M. I. Khatkevich. "Object-relational approach to the creation of large information systems", *Programmnyye sistemy: Teoreticheskiye osnovy i prilozheniya*, Nauka. Fizmatlit, M., 1999, pp. 177–184 (in Russian).
- [8] D. V. Belyshev, Ye. V. Kochurov. "Analysis of data storage methods for modern healthcare information systems", *Program Systems: Theory and Applications*, **7**:2(29) (2016), pp. 85–103 (in Russian), URL: http://psta.psiras.ru/read/psta2016_2_85-103.pdf

- [9] D.V. Belyshev, Ye.V. Kochurov. “Advanced methods of data management in healthcare information systems”, *Program Systems: Theory and Applications*, **7**:3(30) (2016), pp. 79–97 (in Russian), URL: http://psta.psisras.ru/read/psta2016_3_79-97.pdf
- [10] S.I. Komarov, D.V. Alimov. “Use of the multicomponent mechanism at informatization of the large hospital with branches”, *Vrach i informatsionnyye tekhnologii*, 2016, no.6, pp. 25–33 (in Russian).
- [11] *The relational database is doomed*, 2011 (in Russian), URL: <https://habrahabr.ru/post/103021/>
- [12] L. A. Ponomarenko, S. S. Tanyanskiy. “Processing data with a variable structure by means of use of the language specification of the relational model”, *USiM*, 2010, no.6, pp. 47–53 (in Russian).
- [13] L. A. Ponomarenko, V. A. Filatov, S. S. Tanyanskiy. “Minimization of computational costs of the data access in distributed economic decision support systems”, *USiM*, 2014, no.3, pp. 74–80 (in Russian).
- [14] B. Smith. *Introduction to Biomedical Ontologies*, 2008, URL: http://ontology.buffalo.edu/smith/BioOntology_Course.html
- [15] Register of medicines of Russia (in Russian), URL: <https://www.rlsnet.ru>

Sample citation of this publication:

Vladimir Malykh, Aleksey Kalinin, Teymur Yusufov. “Object-relational approach to building a data storage”, *Program systems: Theory and applications*, 2017, **8**:3(34), pp. 169–187. (In Russian).

URL: http://psta.psisras.ru/read/psta2017_3_169-187.pdf