

САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ  
УНИВЕРСИТЕТ

На правах рукописи

Симановский Андрей Александрович

СОГЛАСОВАННЫЕ ЭВОЛЮЦИОННЫЕ ТРАНСФОРМАЦИИ  
ВЗАИМОЗАВИСИМЫХ СЛАБОСТРУКТУРИРОВАННЫХ И  
РЕЛЯЦИОННЫХ СХЕМ

05.13.11 — Математическое и программное обеспечение  
вычислительных машин, комплексов и компьютерных сетей

АВТОРЕФЕРАТ

диссертации на соискание ученой степени  
кандидата физико-математических наук

Санкт-Петербург  
2007

Работа выполнена на кафедре системного программирования  
математико-механического факультета Санкт-Петербургского  
государственного университета.

Научный руководитель: доктор физико-математических наук,  
профессор Новиков Борис Асенович

Официальные оппоненты: доктор технических наук,  
Кузнецов Сергей Дмитриевич

кандидат физико-математических наук,  
Новиков Федор Александрович

Ведущая организация: ГОУ Южно-Уральский государственный  
университет

Защита диссертации состоится “\_\_\_” \_\_\_\_\_ 200\_ года в \_\_ часов на  
заседании диссертационного совета Д212.232.51 по защите диссертаций  
на соискание ученой степени доктора наук при Санкт-Петербургском  
государственном университете по адресу: 198504, Санкт-Петербург,  
Старый Петергоф, Университетский пр., д. 28, математико-механический  
факультет Санкт-Петербургского государственного университета.

С диссертацией можно ознакомиться в Научной библиотеке Санкт-  
Петербургского государственного университета по адресу: 199034, Санкт-  
Петербург, Университетская наб., д. 7/9.

Автореферат разослан “\_\_\_” \_\_\_\_\_ 200\_ года.

Ученый секретарь  
диссертационного совета  
доктор физико-математических наук,  
профессор

Б. К. Мартыненко

# Общая характеристика работы

## Актуальность темы

Ряд современных приложений, работающих с базами данных, таких, например, как электронные системы учета медицинской информации, требуют от уровня представления данных возможности модификации (изменения) со временем хранимых данных и их схем. Процесс изменения схемы данных в подобных системах является неотъемлемой частью жизненного цикла приложения. Таким образом, поддержка модификаций схемы хранимых данных со временем является актуальной задачей для производителей СУБД.

Существует несколько подходов к решению этой задачи в рамках СУБД. Они имеют различные области применимости, предоставляют неодинаковый набор возможностей. Эволюция схем ([12]) обеспечивает высокую производительность системы за счет меньшего объема предоставляемых возможностей. Системы поддержки эволюции схем упорядочивают действия при изменении схем, позволяют описывать семантику изменений в предметной области и отражать изменения предметной области, происходящие со временем, в базе данных, распространяя эти изменения на схему хранимых данных и сами данные. Они избавляют от необходимости создания и реализации *ad-hoc* алгоритмов по изменению схем и данных при каждой модификации. В настоящее время существуют системы поддержки эволюции схем данных для реляционной [6, 11], объектно-ориентированной [13, 3, 8, 7, 9, 5, 10], XML [14, 1, 4] моделей данных.

Использование XML данных требует от приложений организации эффективного хранения и доступа к XML данным. Одним из решений является сохранение данных в реляционных СУБД. Это позволяет не только организовывать хранение и доступ к данным, но и использовать такие отсутствующие в XML, но полезные для приложения механизмы, как индексы, транзакции, многопользовательский доступ и т.д. Также немаловажным фактором, влияющим на принятие такой технологии, является отсутствие на сегодняшний день реализации XML базы данных, удовлетворяющей требованиям широкого спектра приложений. Рассматриваемые приложения часто используют XML и такие языки запросов как XQuery в качестве интерфейса к уровню представления данных, который в свою очередь использует мощные промышленные реляционные СУБД в качестве хранилища данных, реализуя логику преобразования работы с моделью данных XML в работу с реляционной моделью [15]. Как следствие, возникает вопрос эволюции связанных

отображением пары XML и реляционной схем.

Ранее предлагаемые решения в рамках эволюции схем не рассматривали необходимость совместной эволюции схем данных, использующих различные модели. Недавно возник интерес к подобным совместным изменениям схем и отображений между ними, называемым также адаптацией схем, как к проблеме управления моделями данных [2], но вопрос исследуется только в рамках интеграции схем, подхода, хотя и наиболее общего для изменяющихся схем, но в то же время, и наиболее “тяжеловесного”, приводящего к значительным накладным расходам при работе приложения. Предлагаемое решение использует эволюцию схем для решения поставленной задачи изменения пары связанных отображением XML- и реляционной схем, решая задачу адаптации в частном случае, но более эффективным методом.

## Цели работы

Данная работа исследует применимость эволюции схем для описания эволюционных трансформаций взаимозависимых слабоструктурированных и реляционных схем. Цели, преследуемые работой, включают:

- Построение модели эволюции, описывающей изменение взаимозависимых слабоструктурированных и реляционных схем данных.
- Разработка множества элементарных операций, позволяющих описывать трансформации взаимозависимых схем и их отображений.
- Описание интеграционных трансформаций схем в рамках эволюционных преобразований взаимозависимых схем.

## Основные результаты

В работе получены следующие основные результаты:

- Разработана аксиоматическая модель эволюции XML- и реляционной схем, описывающих XML-документ и его представление, хранимое в реляционной базе. Предложенная модель позволяет выделить общую и независимые части обеих схем. Модель позволяет описывать семантические инварианты схем данных и сохранять их при эволюционных трансформациях схем.

- Разработана классификация элементарных преобразований, описывающих трансформации связанных схем. Набор преобразований включает как преобразования характерные исключительно для схем XML-документов или схем реляционных представлений документов, так и преобразования, совместно меняющие сразу обе схемы. Подобная классификация позволяет не только производить произвольные эволюционные трансформации связанных схем, но и вносить ограниченные изменения только в одну из схем, например, с целью оптимизации работы системы.
- Предложен метод декларативного описания трансформаций схем. Метод позволяет строить возможные цепочки элементарных преобразований и ранжировать различные пути трансформации по предложенным в работе критериям, позволяя, например, при внесении изменений в одну из схем, эффективно определить необходимость изменений в связанной схеме.
- Реализован прототип системы, позволяющий применять изложенные в работе методы построения эволюционных преобразований схем на практике.

## **Научная новизна**

В данной работе впервые предложена модель эволюции взаимозависимых слабоструктурированных и реляционных схем. Остальные полученные результаты также являются новыми и дополняют результаты предшествующих работ.

## **Практическая и теоретическая ценность**

С теоретической точки зрения, в работе предложена простая аксиоматическая модель, позволяющая описывать эволюционные трансформации взаимозависимых слабоструктурированных и реляционных схем и связывающих их отображений. Кроме того, в работе исследован вопрос реализации автоматизированного поиска интерграционных трансформаций схем.

Практическая ценность работы состоит в том, что предложенная модель может быть использована для автоматизации эволюционных преобразований существующих XML-реляционных систем, например, с помощью предлагаемого прототипа.

## Апробация работы

Результаты работы докладывались

- на Второй Всероссийской конференции “Методы и Средства Обработки информации” (Москва, октябрь 2005)
- на Девятой Восточно-Европейской конференции “Advances in Databases and Information Systems” (Таллин, Эстония, сентябрь 2005)
- на Шестой конференции “Baltic Conference on Databases and Information Systems” (Рига, Латвия, июнь 2004)
- на Первом коллоквиуме “Spring Colloquium for Young Researchers in Databases and Information Systems (SYRCoDIS)” (Санкт-Петербург, май 2004)
- на семинарах группы теории баз данных при лаборатории исследования операций НИИММ

## Публикации

Основные результаты представлены в шести работах автора, перечисленных в прилагающемся списке работ автора.

## Структура и объем диссертации

Диссертация состоит из пяти глав, заключения и списка литературы. Основной текст диссертации занимает 78 страниц машинописного текста. Библиография содержит 79 наименований. Общий объем диссертации 84 страницы. Рисунки и таблицы нумеруются по главам.

## Содержание работы

**Первая глава** содержит обзор исследуемого вопроса, в том числе обсуждает понятия связанные со схемами данных, темпоральными базами данных, эволюцией схем, XML-реляционными отображениями и адаптацией схем. В главе также дан обзор существующих систем поддержки эволюции схем для реляционных, объектных, XML моделей данных и отображений между XML и реляционными данными. В главе вводится понятие семантической интерпретации XML документа с точки зрения запросов.

Интерпретацией XML документа, соответствующего данному DTD, является отображение, сопоставляющее именам тэгов домены предметной области приложения. Всякая синтаксическая интерпретация  $I$  обладает свойством иерархичности:

$$I(tag_{name}) = f(tag_1, tag_2, \dots, tag_k, attr_1^1, attr_1^2, \dots, attr_k^{N_k}).$$

Где  $tag_i$  образуют последовательность вложенных (в соответствии с DTD) тэгов, а  $attr_i^j$  — соответствующие (в соответствии с DTD) тэгам атрибуты.

Во **второй главе** предложена аксиоматическая система поддержки эволюции пары взаимосвязанных XML- и реляционной схем. Основной идеей системы является выделение промежуточной схемы-медиатора реляционных и XML-данных, являющейся общей частью схем. Система описывает отображение исходных схем в схему-медиатор и распространение изменений схемы-медиатора в исходные схемы.

Первый раздел второй главы посвящен представлению данных в системе. Второй раздел второй главы содержит определение схемы медиатора.

**Определение 1.** *Граф XML схемы — это тройка  $(V, E, \sigma)$ , где  $V$  — алфавит имен тэгов,  $E \subseteq V \times V$  — отношение вложенности,  $\sigma : E \rightarrow \{*, null\}$  — отображение, определяющее кратность вложенности тэгов (звездочкой помечается множественная вложенность).*

**Определение 2.** *Факторизованный граф XML схемы — это пятерка  $(G, V', E', \sigma', A)$ , где  $G$  — граф XML схемы;  $V' \subseteq V^*$  — дизъюнктивный набор подмножеств алфавита тэгов, обладающий следующими свойствами:*

- $V_1 \neq V_2 \in V' \Rightarrow v_1 \cap v_2 = \emptyset$  (дизъюнктивные множества);
- $v_1, v_2 \in V_0 \in V' \Rightarrow (v_1, v_2) \in G.E \Rightarrow G.\sigma((v_1, v_2)) = null$  (“внутри” вершины факторизованного графа XML схемы нет дуг, помеченных звездочками);
- $\forall V \in V' \exists v_0 \in V \mid \forall v \in V v_0 >_* v$  (в каждой вершине факторизованного графа XML схемы есть “корень”, из которого достижимы все вершины исходного графа, лежащие “внутри” данной вершины факторизованного графа XML схемы).

Множество  $E' \subseteq V' \times V'$  определяется соотношением  $(v, u) \in E' \Leftrightarrow \exists v_0 \in v, u_0 \in u (v_0, u_0) \in G.E$  — множество дуг факторизованного графа XML схемы;  $\sigma' : E' \rightarrow \{*, null\}$  — маркировка дуг символами;  $A : U \subset \{v \in V \mid \nexists u \in V (v, u) \in G.E\} \rightarrow V'$  — отображение,

сопоставляющее подмножеству листовых вершин графа XML схемы вершины факторизованного графа XML схемы, определяет множества атрибутов вершин факторизованного графа XML схемы.

Четверка  $(V', E', \sigma', A)$  факторизованного графа XML схемы  $(G, V', E', \sigma', A)$  является схемой-медиатором для данной XML схемы. Отображение XML схемы в схему-медиатор является композицией отображения, получающего граф XML схемы по описанию XML схемы и факторизации графа XML схемы.

Третий раздел второй главы обсуждает отображение реляционной схемы в схему-медиатор. Четвертый раздел второй главы вводит аксиомы системы и обсуждает связь сущностей схемы-медиатора и объектов предметной области приложения.

**Определение 3.** Множество непосредственных предшественников. Непосредственными предшественниками  $P(v)$  вершины  $v$  графа схемы-медиатора  $(G, V', E', \sigma', A)$  являются вершины графа схемы-медиатора, соединенные исходящими дугами с данной вершиной, из которых нельзя достичь других вершин, соединенных исходящими дугами с данной:

$$P(v) = \{u \in V' \mid (u, v) \in E' \wedge \nexists (u_0, v) \in E' u >_* u_0\}.$$

В приведенной формуле  $>_*$  — отношение достижимости, т.е. транзитивное замыкание бинарного отношения  $>$ , заданного на  $V'$ , индуцируемого множеством  $E'$ , а именно  $v > u$  если и только если  $(v, u) \in E'$ .

**Определение 4.** Множество значимых (существенных) предшественников. Значимыми (существенными) предшественниками  $P_e(v)$  вершины  $v$  графа схемы-медиатора является некоторое выбранное подмножество вершин графа схемы-медиатора, находящихся в путях от корня до данной вершины, включающее непосредственных предшественников данной вершины. Множество определяется создателем схем.

**Определение 5.** Граф предшественников. Граф предшественников  $PL(v)$  вершины  $v$  графа схемы-медиатора включает в себя все вершины графа, из которых достижима данная:

$$PL(v) = \{u \in V' \mid u >_* v\}.$$

**Определение 6.** Множество непосредственных (родных) атрибутов. Непосредственными (родными) атрибутами  $N(v)$  вершины  $v$  является

подмножество листовых вершин графа XML схемы, лежащее “внутри” данной факторизованной вершины  $v$ :

$$N(v) = \{u \in v \mid \text{out}(u) = \emptyset\}.$$

В приведенной формуле  $\text{out}(u) = \{t \mid (u, t) \in E\}$ .

**Определение 7.** *Интерфейс.* Интерфейсом  $I(v)$  вершины  $v$  называется объединение множеств атрибутов вершин её графа предшественников:

$$I(v) = \bigcup_{x \in PL(v)} N(x).$$

**Определение 8.** *Множество унаследованных атрибутов.* Унаследованными атрибутами  $H(v)$  вершины  $v$  являются элементы разности как множеств интерфейса вершины  $v$  и её родных атрибутов:

$$H(v) = I(v) \setminus N(v).$$

**Определение 9.** *Множество значимых (существенных) атрибутов.* Множеством значимых (существенных) атрибутов  $N_e(v)$  вершины  $v$  является подмножество интерфейса вершины, включающее родные атрибуты вершины, определяемое создателем схемы.

**Определение 10.** *Множество непосредственных последователей.* Непосредственными последователями  $S(v)$  вершины  $v$  графа схемы-медиатора  $(G, V', E', \sigma', A)$  являются вершины графа схемы-медиатора, с которыми данная вершина соединена исходящими дугами, и которые недостижимы из других вершин, с которыми данная вершина соединена исходящими дугами:

$$S(v) = \{u \in V' \mid (v, u) \in E' \wedge \nexists (v, u_0) \in E' \ u_0 >_* u\}.$$

**Определение 11.** *Множество значимых (существенных) последователей.* Значимыми (существенными) последователями  $S_e(v)$  вершины  $v$  графа схемы-медиатора является некоторое выбранное подмножество вершин графа схемы-медиатора, находящихся в путях от данной вершины до конечной вершины графа, включающее непосредственных последователей данной вершины. Множество определяется создателем схем.

**Определение 12.** *Граф последователей.* Граф последователей  $SL(v)$  вершины  $v$  графа схемы-медиатора включает в себя все вершины графа, достижимые из данной:

$$SL(v) = \{u \in V' \mid v >_* u\}.$$

Аксиомы связывают определенные выше множества:

$$\begin{aligned}
& \forall v \in V' P_e(v) \subset V' \wedge S_e(v) \subset V'; \\
& \forall v \in V' v \notin \bigcup_{x \in P(v)} PL(x) \wedge v \notin \bigcup_{x \in S(v)} SL(x); \\
& \exists! v_R \in V' \forall v \in V' v_R \in PL(v) \wedge P_e(v_R) = \emptyset \wedge v \in SL(v_R); \\
& \exists! v_T \in V' \forall v \in V' v \in PL(v_T) \wedge v_T \in SL(v) \wedge S_e(v_T) = \emptyset; \\
& \forall v \in V' P(v) = P_e(v) \setminus \bigcup_{x \in P_e(v)} (PL(x) \setminus \{x\}); \\
& \forall v \in V' PL(v) = \{v\} \cup \bigcup_{x \in P(v)} PL(x); \\
& \forall v \in V' S(v) = S_e(v) \setminus \bigcup_{x \in S_e(v)} (SL(x) \setminus \{x\}); \\
& \forall v \in V' SL(v) = \{v\} \cup \bigcup_{x \in S(v)} SL(x); \\
& \forall v \in V' I(v) = N(v) \cup H(v); \\
& \forall v \in V' N(v) = N_e(v) \setminus H(v); \\
& \forall v \in V' H(v) = \bigcup_{x \in P(v)} I(x).
\end{aligned}$$

**Утверждение 1.** Пусть абстрактная интерпретация — это заданное на множестве интерпретаций XML схемы отображение во множество отображений  $\Theta : I^* \times DTD^* \rightarrow (V \rightarrow V^*)$ ,  $\Theta(DTD, I)(v) = \{u \in G_{DTD}.V \mid I(v) \equiv h(u) \wedge h \neq const\}$ . Тогда на множестве канонических XML схем система согласованных с синтаксической интерпретацией приложения множеств  $P_e$  и  $N_e$  полна относительно абстрактной интерпретации приложения.

В **третьей главе** приведена таксономия элементарных операций предлагаемой системы.

Преобразования схемы-медиатора представляют собой набор операций над графом с помеченными атрибутами вершинами, сохраняющих аксиомы. Операции над схемой-медиатором включают в себя добавление и удаление атрибута, добавление и удаление дуги, добавление и удаление вершины, а также слияние двух вершин и расщепление вершины. Операции над схемой-медиатором позволяют из некоторой исходной

схемы-медиатора получать любую наперед заданную схему-медиатор, имеющую данные множества значимых предшественников, последователей и атрибутов.

**Утверждение 2.** *Введенные операции позволяют из некоторой исходной схемы-медиатора получать любую наперед заданную схему-медиатор, имеющую данные множества значимых предшественников, последователей и атрибутов.*

Второй раздел третьей главы рассматривает преобразования XML схемы. Ряд операций, изменяющих граф XML схемы, удобнее ввести, оперируя графом XML схемы. Остальные операции не меняют граф XML схемы и выражаются исключительно в терминах DTD. Набор операций включает перемещение вершины вниз/вверх по иерархии, добавление и удаление “листовой” вершины, добавление и удаление верхней дуги, перемещение дуги вверх/вниз, маркирование/демаркирование дуги.

**Утверждение 3.** *Введенные операции, изменяющие граф XML схемы, позволяют получать произвольные графы XML схемы, соответствующие данной схеме-медиатору.*

**Утверждение 4.** *Совокупность введенных над XML схемой операций, затрагивающих только XML схему, позволяет описать, т.е. получить, имея некоторого представителя, класс всех XML схем, соответствующих данной схеме-медиатору.*

Четвертый раздел третьей главы посвящен операциям, преобразующим реляционную схему. Набор операций включает добавление и удаление дублирующего столбца из таблицы.

**Утверждение 5.** *Операции добавления и удаления дублирующего атрибута из реляционной переменной позволяют преобразовать любую исходную реляционную схему, таблицы которой имеют столбцы с именами, образующими фиксированное множество, и которой соответствует фиксированная схема-медиатор, в данную целевую реляционную схему, таблицы которой имеют столбцы с именами, образующими то же множество, и которой соответствует та же схема-медиатор, при условии, что в обоих преобразованиях реляционных схем в схему-медиатор каждый столбец использован в определении хотя бы одного представления.*

**Четвертая глава** исследует вопрос трансформации схем в рамках предложенной модели эволюции.

**Определение 13.** *Функцией эволюции называется функция  $SE : M^* \times S \times R^* \rightarrow Seq^*$ , где  $M$  — условия целостности схемы;  $S^*$  — множество схем;  $R^*$  — множество постусловий;  $Seq$  — множество последовательностей операций, преобразующих начальную схему  $S_0 \in S$  в схему, удовлетворяющую условиям  $R \in R^*$ , не нарушая условий целостности  $M \in M^*$ .*

В применении к предлагаемой системе  $M$  — набор множеств  $P_e$ ,  $S_e$  и  $N_e$ ;  $S$  — пара из реляционной и XML схемы.  $Seq$  — множество последовательностей операций, введенных в третьей главе.  $S$  — множество пар взаимозависимых реляционной и XML схем.

**Определение 14.** *Граф эволюции — это четверка  $(S^*, C, S_0, R)$ , где  $S^*$  — множество схем, удовлетворяющих условиям целостности,  $C : S^* \times S^* \rightarrow CH^*$  — множество дуг, соединяющих вершины-схемы, помеченных операциями;  $CH^*$  — множество операций;  $S_0 \in S^*$  — начальная схема,  $R$  — множество постусловий. Граф эволюции удовлетворяет условию применимости операций:*

$$C(S_1, S_2) = CH \Rightarrow CH.pre(S_1) \wedge CH.post(S_2).$$

Трансформация схемы может быть представлена как множество путей в графе эволюции, исходящих из данной начальной схемы  $S_0$  и заканчивающихся в схемах, удовлетворяющих условиям  $R$ .

Второй раздел четвертой главы посвящен эвристике, позволяющей сделать задачу поиска на графе эволюции конечной и ранжированию путей эволюции.

В **пятой главе** приводится описание прототипа, реализующего предложенную модель эволюции. Существующие программные средства, осуществляющие поддержку эволюции схем, не позволяют работать с парой взаимосвязанных схем. Прототип состоит из двух частей — SQL-скрипта, порождающего в реляционной базе данных, осуществляющей хранение данных, необходимые для системы поддержки эволюции таблицы и хранимые процедуры, и веб-приложения, реализующего интерфейс, обеспечивающий визуализацию и возможность изменять схемы.

**Заключение** содержит список основных результатов, полученных в работе.

## Список литературы

- [1] *Кукс С. В.* Аксиоматизация эволюции схемы xml-баз данных // *Программирование*. — 2003. — Т. 29, № 3. — С. 140–146.
- [2] *Bernstein P. A.* Applying model management to classical meta data problems. // *CIDR*. — 2003.
- [3] *Kim W., Chou H.-T.* Versions of schema for object-oriented databases. // *VLDB*. — 1988. — Pp. 148–159.
- [4] *Leonardi E., Bhowmick S. S.* Detecting changes on unordered xml documents using relational databases: a schema-conscious approach. // *CIKM*. — 2005. — Pp. 509–516.
- [5] *Lerner B. S.* A model for compound type changes encountered in schema evolution // *ACM Trans. Database Syst.* — 2000. — Vol. 25, no. 1. — Pp. 83–127.
- [6] *McKenzie L. E., Snodgrass R. T.* Evaluation of relational algebras incorporating the time dimension in databases // *ACM Comput. Surv.* — 1991. — Vol. 23, no. 4. — Pp. 501–543.
- [7] *Monk S., Sommerville I.* Schema evolution in oodbs using class versioning // *SIGMOD Rec.* — 1993. — Vol. 22, no. 3. — Pp. 16–22.
- [8] *Osborn S. L.* The role of polymorphism in schema evolution in an object-oriented database // *IEEE Transactions on Knowledge and Data Engineering*. — 1989. — Vol. 1, no. 3. — Pp. 310–317.
- [9] *Peters R. J., Özsu M. T.* An axiomatic model of dynamic schema evolution in objectbase systems // *ACM Trans. Database Syst.* — 1997. — Vol. 22, no. 1. — Pp. 75–114.
- [10] *Rashid A., Sawyer P.* Object database evolution using separation of concerns. // *SIGMOD Record*. — 2000. — Vol. 29, no. 4. — Pp. 26–33.
- [11] *Roddick J. F.* Sql/se: a query language extension for databases supporting schema evolution // *SIGMOD Rec.* — 1992. — Vol. 21, no. 3. — Pp. 10–16.
- [12] *Roddick J. F.* A survey of schema versioning issues for database systems // *Information and Software Technology*. — 1995. — Vol. 37, no. 7. — Pp. 383–393. [citeseer.ist.psu.edu/roddick95survey.html](http://citeseer.ist.psu.edu/roddick95survey.html).

- [13] Semantics and implementation of schema evolution in object-oriented databases. / J. Banerjee, W. Kim, H.-J. Kim, H. F. Korth // SIGMOD Conference. — 1987. — Pp. 311–322.
- [14] Xem: Managing the evolution of xml documents. / H. Su, D. Kramer, L. Chen et al. // RIDE-DM. — 2001. — Pp. 103–110.
- [15] Xquery implementation in a relational database system / S. Pal, I. Cseri, O. Seeliger et al. // VLDB '05: Proceedings of the 31st international conference on Very large data bases. — VLDB Endowment, 2005. — Pp. 1175–1186.

## Работы автора по теме диссертации

- S. V. Соох, A. A. Simanovsky. Regular Expressions in XML Schema Evolution. *Вісник Національного технічного університету “Харківський політехнічний інститут”*: Збірник наукових праць. Тематичний випуск “Системний аналіз, управління та інформаційні технології” 1(1). — Харьков, Украина, 2004 — с. 24–38.
- A. Simanovsky. Evolution of Schema of XML-documents Stored in a Relational Database. *Proceedings of 6th Baltic DBIS Conference*. — Riga, Latvia, 2004. — с. 192–204.
- A. Simanovsky. Applying the Reconfiguration-Design Formalism to XML Stored in a Relational Database. *Proceedings of the Spring Young Researcher’s Colloquium on Database and Information Systems*. — Санкт-Петербург, 2004. — с. 75–77.
- A. Simanovsky. Three Layer Evolution Model for XML Stored in Relational Databases. *Proceedings of the Ninth East-European Conference on Advances in Databases and Information Systems*. — Tallinn, Estonia, 2005. — с. 66–79.
- А. А. Симановский. Автоматизация разработки схем баз данных для XML-реляционных систем. *Методы и средства обработки информации. Труды Второй Всероссийской научно-технической конференции*. — Москва, 2005. — с. 113–118.
- А. А. Симановский. Совместные эволюционные трансформации взаимозависимых XML и реляционных схем. *Вестник Санкт-Петербургского университета. Серия 10*. 2007. Выпуск 1 — с. 102–111