

УДК 004.65

### **МНОГОМЕРНОЕ ХРАНЕНИЕ ЖУРНАЛОВ СОБЫТИЙ ДЛЯ ИЗВЛЕЧЕНИЯ И АНАЛИЗА ПРОЦЕССОВ**

*Брейман Александр Давидович*, кандидат технических наук, доцент, Национальный исследовательский университет «Высшая школа экономики», 101000, Российская Федерация, г. Москва, ул. Мясницкая, 20, e-mail: abreyman@hse.ru

*Богословский Егор Максимович*, магистрант, Национальный исследовательский университет «Высшая школа экономики», 101000, Российская Федерация, г. Москва, ул. Мясницкая, 20, e-mail: 9163540822N@gmail.com

Журналы событий, сохраняемые современными информационными и техническими системами, как правило, содержат достаточно данных для автоматизированного восстановления моделей соответствующих процессов. Разработано множество алгоритмов для построения моделей процессов, проверки соответствия фактического поведения системы модельному, сравнения моделей процессов и т.д. Однако возможность быстрого анализа выбираемых пользователями частей журнала до сих пор не нашла полноценной реализации. В статье описан метод многомерного хранения журналов событий для извлечения и анализа процессов, основанный на подходе ROLAP. Результатом анализа журнала является направленный невзвешенный граф, представляющий собою сумму возможных последовательностей событий, упорядоченных по вероятности их возникновения с учетом заданных условий. Разработанный инструмент позволяет выполнять совместный анализ моделей подпроцессов, восстановленных из частей журнала путем задания критериев отбора событий и требуемого уровня детализации модели.

**Ключевые слова:** извлечение и анализ процессов, журналы событий, многомерные модели данных, многомерный анализ процессов, аналитические операции над многомерным кубом, OLAP, ROLAP, XES

### **MULTIDIMENSIONAL EVENT LOGS STORAGE FOR PROCESS MINING**

*Breyman Alexandr D.*, Ph.D. (Engineering), Associate Professor, National Research University “Higher School of Economics”, 20 Myasnitskaya St., Moscow, 101000, Russian Federation, e-mail: abreyman@hse.ru

*Bogoslovskiy Yegor M.*, undergraduate student, National Research University “Higher School of Economics”, 20 Myasnitskaya St., Moscow, 101000, Russian Federation, e-mail: 9163540822N@gmail.com

Event logs collected by modern information and technical systems usually contain enough data for automated process models discovery. A variety of algorithms was developed for process models discovery, conformance checking, log to model alignment, comparison of process models, etc., nevertheless a quick analysis of ad-hoc selected parts of a journal still have not get a full-fledged implementation. This paper describes an ROLAP-based method of multidimensional event logs storage for process mining. The result of the analysis of the journal is visualized as directed graph representing the union of all possible event sequences, ranked by their occurrence probability. Our implementation allows the analyst to discover process models for sublogs defined by ad-hoc selection of criteria and value of occurrence probability.

**Keywords:** process mining, event logs, multidimensional data models, multidimensional process mining, multidimensional analytical operations, OLAP, ROLAP, XES

**Введение.** Современные информационные, организационные и технические системы, как правило, регистрируют выполняемые операции в виде протоколов – журналов событий (ЖС, англ. event log). Примерами подобных журналов могут служить: перечень обращений к веб-серверу, журнал операций, выполненных станком с ЧПУ, перечень действий, выполненных сотрудниками службы поддержки при обработке обращения пользователя. Данные, накопленные в ЖС, могут использоваться для выбора решений в задачах планирования [1], организации диспетчеризации и оценке ее эффективности [3] и во многих других задачах. Хотя для разных систем и процессов содержание ЖС существенно различается, можно описать универсальную расширяемую структуру данных, применимую для любых систем. Такая структура позволяет унифицированным образом строить модели процессов, породивших данные ЖС.

Такие журналы, описывающие некоторый процесс, как правило, обладают «богатым» набором атрибутов, позволяющим рассматривать этот процесс и его составляющие с разных точек зрения. При анализе ЖС, описывающих процесс на разных уровнях абстракции и с разной степенью детализации, важно иметь возможность сравнения моделей, построенных на основе выборок из ЖС. Например, сгруппировав пользователей веб-сайта по сайту, с которого они перешли на данный сайт, а затем построив и сравнив модели процессов их поведения на сайте, аналитик может не только выявить менее эффективные каналы рекламы, но и определить качественные различия действий пользователей разных групп.

Однако большинство современных инструментов извлечения и анализа процессов позволяют исследователю за один шаг выявлять модель процесса только одного вида, только для журнала целиком и с единственной комбинацией настроек. Для того чтобы по данным того же ЖС построить частичную модель, необходимо прочитать файл журнала и выполнить анализ заново.

*Целями настоящей статьи* являются: систематизированное представление средств извлечения и анализа частичных моделей процессов из ЖС; анализ возможностей и вычислительной эффективности предлагаемых алгоритмов.

**Основные понятия извлечения и анализа процессов.** При описании основных понятий извлечения и анализа процессов целесообразно использовать терминологию ван дер Алста [9]. Будем рассматривать ЖС как зафиксированное в виде множества трасс (англ. trace) множество экземпляров процесса, т.е. отдельных выполнений системой одного и того же процесса. Каждая трасса обладает уникальным идентификатором экземпляра процесса (например, номер обращения пользователя) и содержит последовательность записей об отдельных событиях (англ. event). Запись о событии может содержать разнообразные его характеристики – атрибуты (англ. attributes). Например, момент времени, когда произошло событие; класс события; сотрудник, выполнивший операцию; ресурс; стоимость операции и т.д.

Трассы и журналы также могут содержать атрибуты, характеризующие эти объекты в целом, а не составляющие их элементы. Атрибуты могут быть составными, т.е. состоять из других атрибутов, причем уровень вложенности атрибутов не ограничен.

Выявление процессов (англ. process discovery) как одна из разновидностей извлечения и анализа процессов (англ. process mining) соответствует построению моделей процессов, породивших рассматриваемый ЖС. Модели могут быть представлены в виде систем переходов, сетей Петри, конечных автоматов, деревьев процессов и т.д.

Проверка соответствия процесса заданной модели (англ. process conformance checking) занимается сравнением пары моделей: заранее заданной (например, предписанной законодательством или внутренним нормативным документом организации) и построенной в результате выявления процесса.

Для хранения и анализа журналов событий IEEE Task Force on Process Mining предлагает использовать формат XES, разработанный в Технологическом университете Эйнховена и являющийся приложением XML [10]. Данный стандарт описывает 4 типа элементов:

- Журнал (Log) – элемент, содержащий описание группы трасс, отражающих разные экземпляры одного и того же процесса;
- Трасса (Trace) – элемент, содержащий описание одной трассы. Вложен в Log;
- Событие (Event) – элемент, содержащий описание отдельного события трассы. Вложен в Trace;
- Мета (Meta) – элемент, содержащий мета-информацию о журнале событий.

Формат XES позволяет задавать расширения, описывающие часто используемые наборы атрибутов элементов.

**Многомерный анализ процессов.** Во многих случаях ЖС можно исследовать в предположении о том, что процесс, его породивший, имеет различающиеся варианты (подпроцессы). Например, процесс обслуживания клиентов предприятием сферы услуг может различаться для обычных и приоритетных клиентов; для рабочих и выходных дней и пр.

Для построения модели каждого подпроцесса для части событий ЖС (например, оставив только трассы, относящиеся к выходным дням), необходимо прочитать файл ЖС; отфильтровать его по заданному условию; выполнить анализ заново. Для журналов небольшого размера такой подход вполне применим. Однако при обработке объемных журналов повторное их считывание и фильтрация занимает неприемлемо большой промежуток времени. Кроме того, интересно сравнение моделей, полученных для разных подпроцессов [5].

Представляется целесообразным обеспечить исследователей инструментом, позволяющим легко переключаться между моделями, выявленными с разных точек зрения и для разных подпроцессов, подобно тому, как системы OLAP позволяют легко получать результаты расчета различных агрегатов над данными, сгруппированными по-разному.

Один из ранних подходов к помещению ЖС в хранилище данных фиксированной структуры описан в [4]. Однако предложенная в [4] универсальная структура хранилища данных обладает невысокой «выразительной силой» и не позволяет выполнять глубокий анализ с использованием произвольных иерархий.

Для различения данных, относящихся к разным подпроцессам, при сохранении единой модели процесса, в [7] вводится понятие многомерной модели процесса, допускающей кратные узлы и дуги в графе модели. Для обеспечения возможности построения модели любого подпроцесса предлагается строить куб событий – EventCube – вариант многомерного куба, хранящего все события процесса во всех возможных группировках, т.е. все его кубоиды. Основное преимущество подхода состоит в высокой выразительности получаемых моделей, в которых наглядно видны различия подпроцессов. Основным ограничением данного подхода является необходимость разработки многомерного варианта для каждого алгоритма извлечения и анализа процессов.

Более универсальный подход к решению проблемы многомерности предложен в [8]. Предлагается рассматривать многомерный куб (называемый кубом процессов – Process Cube), хранящий все события и трассы ЖС, лишь как средство фильтрации событий и/или трасс. Иными словами, куб процессов позволяет для каждого подпроцесса получить журнал в том виде, в каком он был бы представлен, если бы содержал данные только этого подпроцесса. Преимуществом такого подхода является применимость к журналам подпроцессов всех имеющихся алгоритмов анализа и извлечения процессов.

Предложенный в [8] подход был развит в [6] и реализован в системе ProCube с использованием гибридного HОLAP хранилища, включающего реляционную СУБД MySQL и MOLAP-систему Palo. Результаты экспериментов с системой ProCube, приведенные в [6], продемонстрировали показательную зависимость времени получения ЖС из хранилища от количества событий в извлекаемом ЖС. Это ограничивает применимость данной системы – только для ЖС небольшого объема.

В настоящей работе исследуется возможность использования подхода ROLAP при реализации куба процессов [6], а также сравнительная эффективность реализаций ROLAP и HОLAP подходов.

Авторами настоящей статьи разработан метод многомерного хранения ЖС для извлечения и анализа процессов, основанный на подходе ROLAP, спроектирована база данных (БД) для хранения ЖС, а также разработаны система загрузки процессных данных из ЖС в формате XES в БД и система для пользовательского доступа к БД, дающая возможность пользователю-аналитику оперативно получать модели подпроцессов и визуально их сравнивать.

Результатом анализа данных является направленный невзвешенный граф, представляющий собой модель процесса. Множество узлов графа соответствует множеству событий ЖС, а множество ребер – отношению непосредственного следования событий.

В качестве примера рассмотрим ЖС, содержащий записи о процессе обслуживания страховой компанией заявлений клиентов о страховых случаях. Клиенты обращаются в компанию по телефону, звонки обрабатывают операторы двух колл-центров (англ. call-centers). Принятые заявления рассматривают сотрудники внутренних подразделений, например, отдела автострахования. Каждая трасса ЖС содержит последовательность событий следующих классов: входящий звонок; проверка достаточности данных; регистрация заявления; отказ в регистрации заявления; рассмотрение заявления; определение суммы выплаты; принятие решения о выплате; завершение обработки. Для каждого класса событий определены атрибуты: например, для входящего звонка – в какой момент он поступил; кто его совершил; какой сотрудник какого колл-центра его принял. На рис. 1 приведен пример начала ЖС с двумя трассами, а на рис. 2 – модель этого процесса.

**Проектирование базы данных для хранения журналов событий.** В соответствии с [8], представим ЖС в виде гиперкуба особого вида – куба процессов. Измерениями такого куба являются атрибуты трасс и событий ЖС. Каждой комбинации значений измерений соответствует ячейка куба процессов, содержащая подмножество событий ЖС, обладающих данными значениями атрибутов. Если значения измерения образуют иерархию, находясь в отношении «родитель – потомок» или «часть – целое», то отдельные события ЖС могут принадлежать нескольким ячейкам куба процессов.

Куб процессов, как и OLAP-куб в методе многомерного моделирования [2], представляет собой гиперкуб. Однако применить метод многомерного моделирования к процессным данным не представляется возможным, поскольку содержание ячейки куба процессов (множество событий) не сводимо к множеству числовых характеристик фактов (метрик). Соответственно, все операции, применимые к OLAP-кубам, опирающиеся на возможность вычисления значений агрегатов как функций от метрик, оказываются в исходной форме неприменимыми к кубам процессов. Тем не менее, эти операции можно переопределить с учетом характера хранимых в кубе процессов данных. Так, например, результатом операции свертки (roll-up) может быть объединение событий из сворачиваемых ячеек [8].

ЖУРНАЛ СОБЫТИЙ				
<b>ТРАССА</b>		<b>ТРАССА</b>		
ID	1049	ID	1050	
АТТРИБУТЫ	<b>СОБЫТИЕ</b>		<b>СОБЫТИЕ</b>	
	<b>КЛАСС</b>	ВХОДЯЩИЙ ЗВОНОК	<b>КЛАСС</b>	ВХОДЯЩИЙ ЗВОНОК
	<b>МОМЕНТ</b>	03.01.2014 08:15:02 MSK	<b>МОМЕНТ</b>	03.01.2014 10:50:00 MSK
	<b>КТО</b>	КЛИЕНТ	<b>КТО</b>	КЛИЕНТ
	<b>ГДЕ</b>	КОЛЛЦЕНТР1	<b>ГДЕ</b>	КОЛЛЦЕНТР2
	<b>КАК</b>	ВЫПОЛНЕНО	<b>КАК</b>	ВЫПОЛНЕНО
	<b>СОБЫТИЕ</b>		<b>СОБЫТИЕ</b>	
	<b>КЛАСС</b>	ПРОВЕРКА ДОСТАТОЧНОСТИ ДАННЫХ	<b>КЛАСС</b>	ПРОВЕРКА ДОСТАТОЧНОСТИ ДАННЫХ
	<b>МОМЕНТ</b>	03.01.2014 08:15:25 MSK	<b>МОМЕНТ</b>	03.01.2014 10:51:11 MSK
	<b>КТО</b>	ОПЕРАТОР	<b>КТО</b>	ОПЕРАТОР
<b>ГДЕ</b>	КОЛЛЦЕНТР1	<b>ГДЕ</b>	КОЛЛЦЕНТР2	
<b>КАК</b>	НАЧАТО	<b>КАК</b>	НАЧАТО	
<b>СОБЫТИЕ</b>		<b>СОБЫТИЕ</b>		
<b>КЛАСС</b>	ПРОВЕРКА ДОСТАТОЧНОСТИ ДАННЫХ	<b>КЛАСС</b>	ПРОВЕРКА ДОСТАТОЧНОСТИ ДАННЫХ	
<b>МОМЕНТ</b>	03.01.2014 08:17:12 MSK	<b>МОМЕНТ</b>	03.01.2014 10:52:42 MSK	
<b>КТО</b>	ОПЕРАТОР	<b>КТО</b>	ОПЕРАТОР	
<b>ГДЕ</b>	КОЛЛЦЕНТР1	<b>ГДЕ</b>	КОЛЛЦЕНТР2	
<b>КАК</b>	ВЫПОЛНЕНО	<b>КАК</b>	ВЫПОЛНЕНО	
<b>СОБЫТИЕ</b>		<b>СОБЫТИЕ</b>		
<b>КЛАСС</b>	РЕГИСТРАЦИЯ ЗАЯВЛЕНИЯ	<b>КЛАСС</b>	ОТКАЗ В РЕГИСТРАЦИИ ЗАЯВЛЕНИЯ	
<b>МОМЕНТ</b>	03.01.2014 08:17:12 MSK	<b>МОМЕНТ</b>	03.01.2014 10:52:43 MSK	
<b>КТО</b>	ОПЕРАТОР	<b>КТО</b>	ОПЕРАТОР	
<b>ГДЕ</b>	КОЛЛЦЕНТР1	<b>ГДЕ</b>	КОЛЛЦЕНТР2	
<b>КАК</b>	НАЧАТО	<b>КАК</b>	ВЫПОЛНЕНО	
<b>СОБЫТИЕ</b>		<b>СОБЫТИЕ</b>		
<b>КЛАСС</b>	РЕГИСТРАЦИЯ ЗАЯВЛЕНИЯ			
<b>МОМЕНТ</b>	03.01.2014 08:21:01 MSK			
<b>КТО</b>	ОПЕРАТОР			
<b>ГДЕ</b>	КОЛЛЦЕНТР1			
<b>КАК</b>	ВЫПОЛНЕНО			
<b>СОБЫТИЕ</b>				
<b>КЛАСС</b>	РАССМОТРЕНИЕ ЗАЯВЛЕНИЯ			
<b>МОМЕНТ</b>	03.01.2014 08:20:20 MSK			
<b>КТО</b>	МЕНЕДЖЕР			
<b>ГДЕ</b>	ОТДЕЛ АВТОСТРАХОВАНИЯ			
<b>КАК</b>	НАЧАТО			
...				

Рис. 1. Пример журнала событий страховой компании

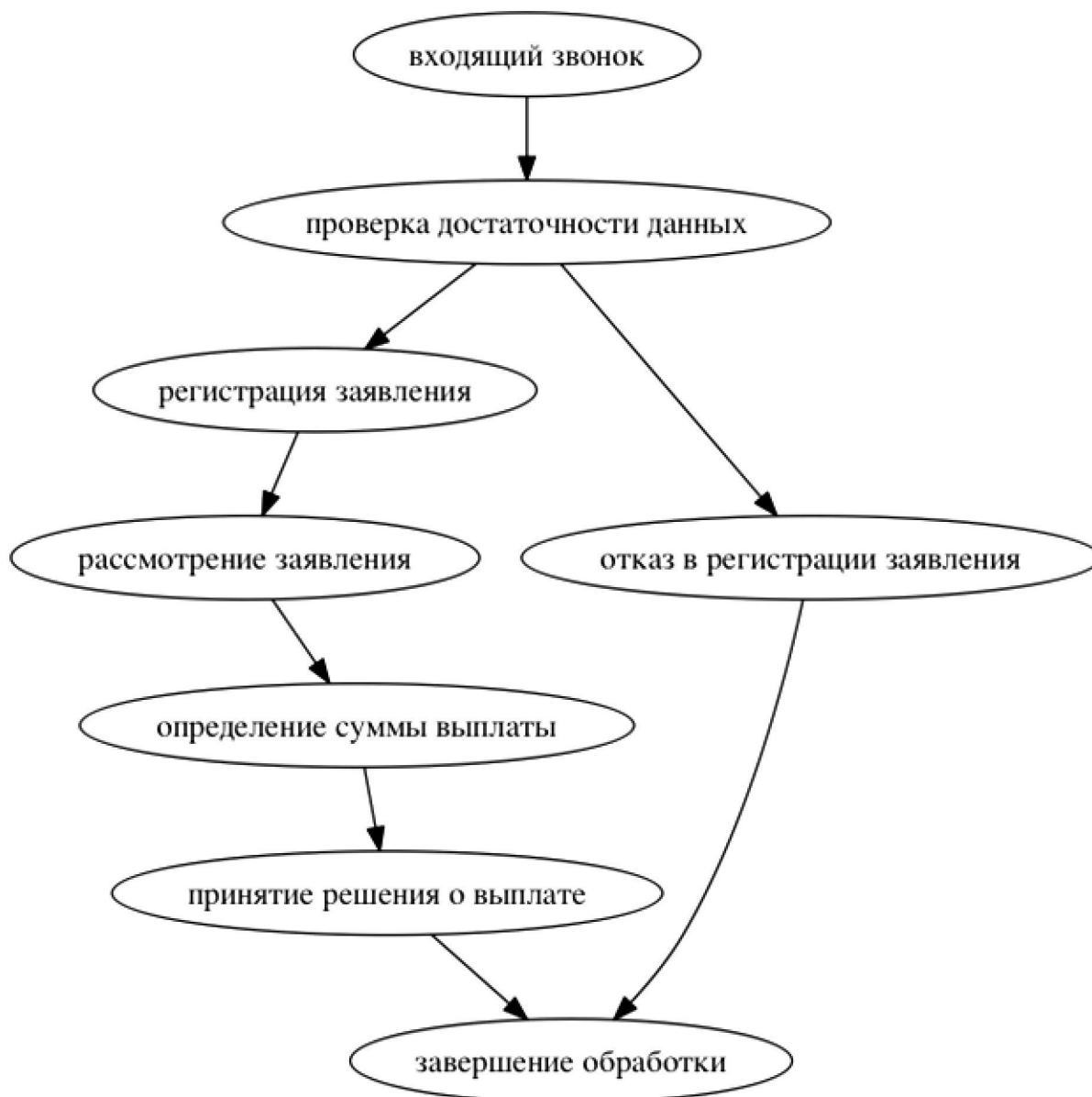


Рис. 2. Пример модели процесса в страховой компании с двумя вариантами подпроцессов

Отображение логической многомерной модели данных в физическую схему базы данных существенно зависит от используемой OLAP-системы. Структура хранения данных в MOLAP-системе однозначно соответствует логической многомерной модели. Для ROLAP-систем можно построить несколько вариантов физической реляционной схемы, соответствующих схемам «звезды» и «снежинки» с разной степенью нормализации измерений.

Наиболее прямолинейное применение подхода OLAP для хранения одного или нескольких ЖС, соответствующих одному и тому же конкретному процессу, состоит в объединении данных ЖС в одной денормализованной реляционной таблице [6]. Каждый кортеж такой таблицы хранит расширенную запись об одном событии: значения всех атрибутов события, значения всех атрибутов той трассы и того процесса, к которым принадлежит собы-

тие. При таком подходе физическая схема БД не может быть универсальной, а жестко привязывается к структуре конкретного ЖС.

Для достижения большей гибкости в описываемой разработке процессы, трассы и события хранятся в отдельных таблицах (FACT, TRACE, EVENT соответственно). Каждая таблица содержит атрибуты, которыми обязательно обладают соответствующие сущности (например, название и момент времени для событий). Для атрибутов, не являющихся обязательными, используем парадигму Сущность – Атрибут – Значение (Entity – Attribute – Value) и создадим таблицы свойств трасс (TPROPERTY) и свойств событий (EPROPERTY). Таблицы свойств содержат в себе пары «ключ – значение»; указание на тип данных значения; и внешний ключ, ссылающийся на таблицу трасс или событий.

Физическая схема базы данных приведена на рис. 3.

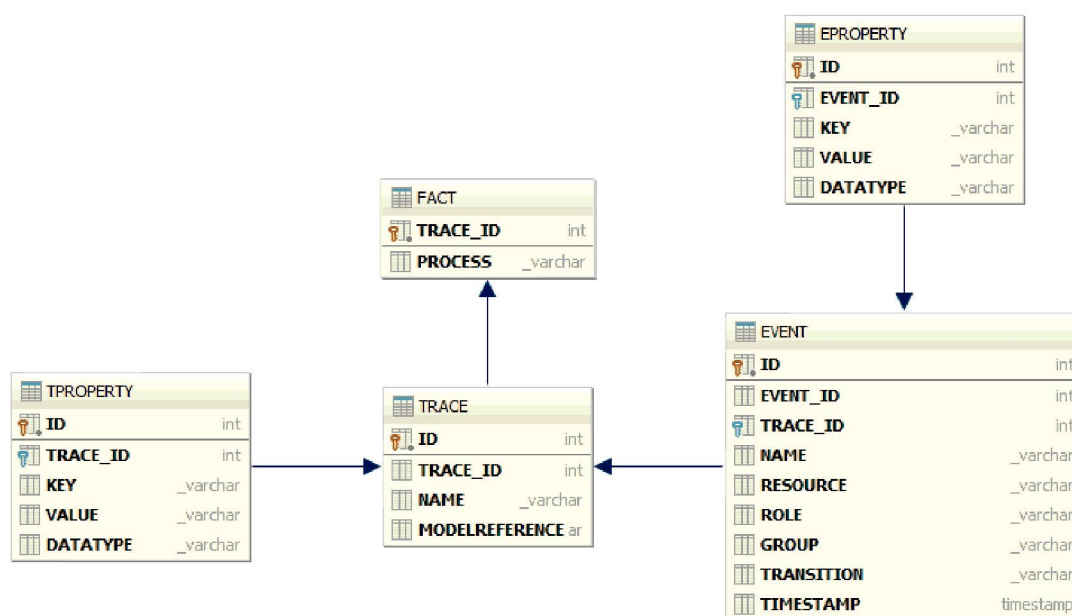


Рис. 3. Физическая схема базы данных

**Построение и выполнение аналитических запросов к хранилищу.** Запросы к хранилищу строятся с использованием визуального редактора запросов, позволяющего выбрать поле интересующего измерения; выбрать значение поля для создания подзапроса; выбрать условие для поля (=, <>); задать отношение подзапроса к родительской части запроса (INTERSECT, UNION); создать вложенные запросы. После выбора атрибута трассы или события пользователю становится доступным задание ограничений на выборку событий и трасс путем указания оператора сравнения и одного из значений этого атрибута. После выбора значения атрибута в окне конструктора запросов появляется новый элемент с указанной операцией и выбранным значением. Для элемента конструктора пользователь может выделять подзапросы из «большого» запроса. Еще одна операция, доступная для пользователя, – это выбор пересечения либо объединения множеств полученных процессов в результате выполнения подзапроса.

После выполнения составленного запроса в основную рабочую область интерфейса программы добавляется модель процесса в виде направленного графа с кратким описанием.

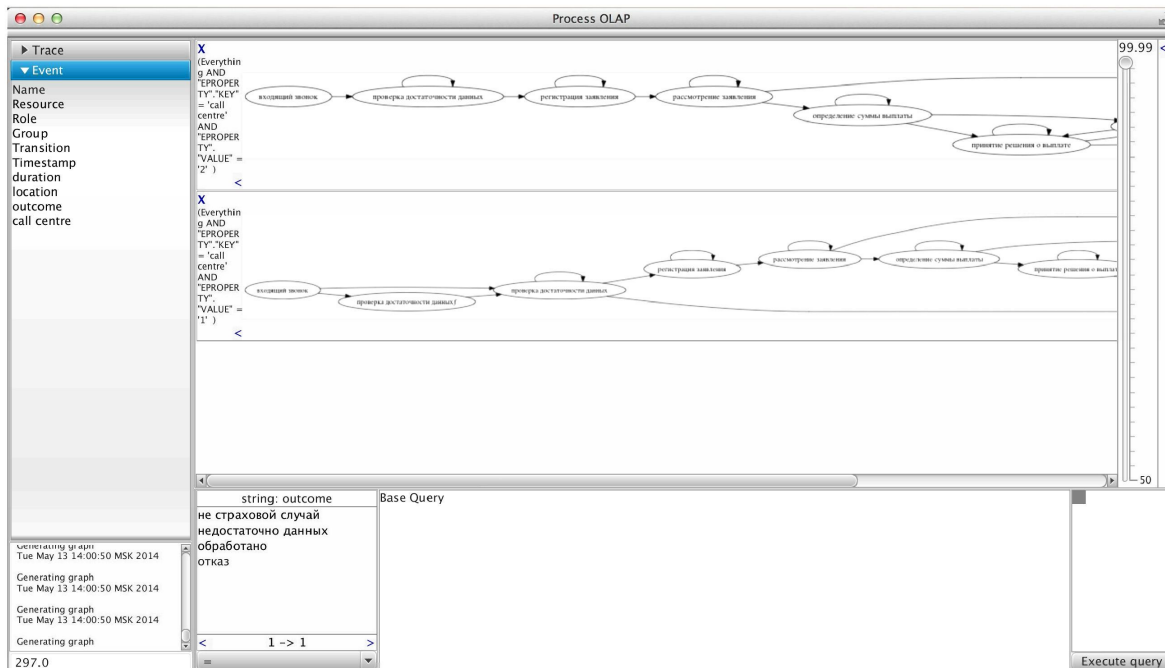


Рис. 4. Вид интерфейса программы с двумя моделями подпроцессов на максимальном уровне детализации

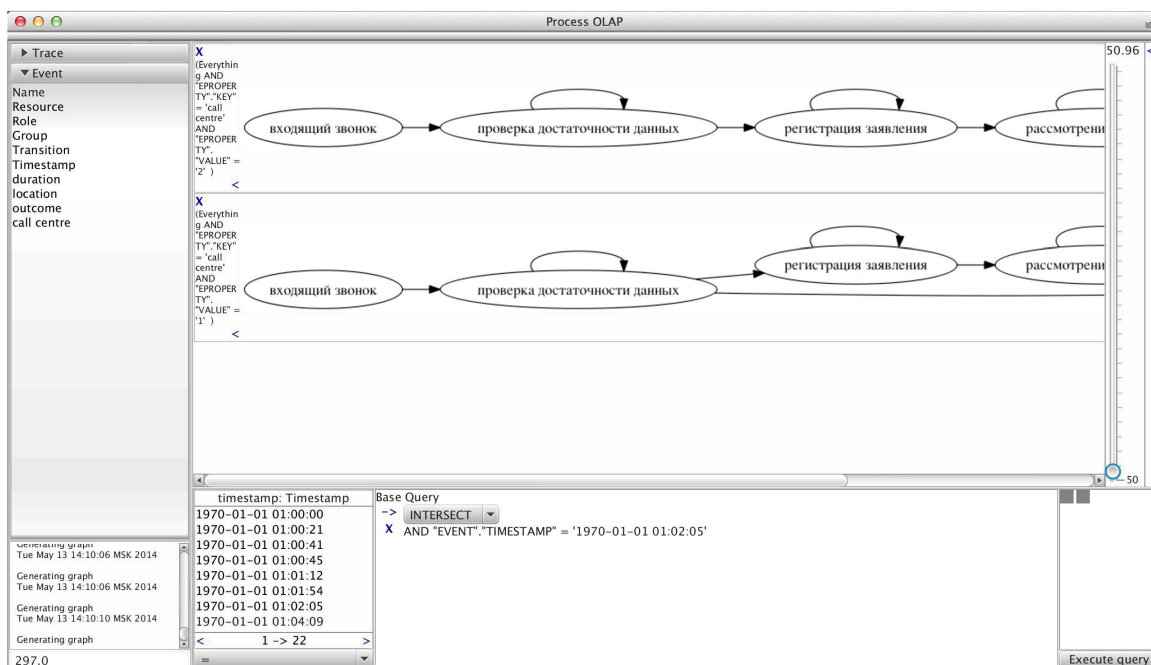


Рис. 5. Вид интерфейса программы с двумя моделями подпроцессов на уровне детализации 0,5



Во время работы с приложением можно менять уровень детализации моделей. При выполнении запроса БД возвращает набор трасс и событий, которые подходят под заданное условие. Затем полученный набор трасс сортируется по убыванию частоты их встречаемости в ЖС. Уровень детализации задается положительным вещественным числом, меньшим единицы, и может быть интерпретирован как доля трасс, которую должна отразить модель. Если уровень детализации максимален (равен единице), модель содержит все трассы, хотя бы единожды встретившиеся в ЖС. Если уровень детализации меньше максимального, то наиболее редко встречающиеся трассы отбрасываются и модель строится без них. На рис. 4 и 5 приведен внешний вид интерфейса программы, отображающей две модели подпроцессов на двух уровнях детализации.

**Экспериментальное исследование.** Для вычислительных экспериментов был использован ЖС, содержащий 35000 событий в 4514 трассах. Из полного журнала были выделены поджурналы, содержащие 150, 300, 700, 1500, 3000, 12000 и 24000 событий. Для полного журнала и каждого из поджурналов была выполнена загрузка в БД, а затем произведена выгрузка для получения срезов журнала. Вычислительные эксперименты были выполнены на MacBook Pro Retina Late 2013, оснащенный процессором Intel Core i7 с тактовой частотой 2800 МГц, оперативной памятью 16 Гб, SSD-накопителем объемом 512 Гб, под управлением операционной системы OS X 10.9.2 с использованием Java 7 SE. Система ProM с пакетом, реализующим подход HOLAP [6], была получена из системы управления версиями исходного кода, поддерживаемой Техническим университетом Эйндховена (<https://svn.win.tue.nl/repos/prom/Packages/TatianaMamaliga>).

Результаты эксперимента приведены на рис. 6 и 7.

Длительность загрузки ЖС при использовании HOLAP-подхода [6] измерялась для каждого поджурнала в трех различных конфигурациях: HOLAP 2D – гиперкуб строился для двух измерений (колл-центр и класс события), HOLAP 3D – гиперкуб строился для трех измерений (момент времени, колл-центр и класс события), HOLAP 4D – гиперкуб строился для четырех измерений (момент времени, длительность, колл-центр и класс события). При использовании ROLAP-подхода длительность загрузки измерялась в единственной конфигурации: в базу данных переносились все события и все измерения.

Эксперимент продемонстрировал, что длительность загрузки ЖС практически не зависит от количества измерений и от способа реализации многомерной модели данных. Действительно, во всех конфигурациях при загрузке каждого события выполняется один и тот же набор действий, не зависящий ни от количества измерений, ни от способа сохранения данных.

Длительность выгрузки данных из БД также измерялась для разных конфигураций: 1D – одно ограничение: выборка событий, относящихся к заданному колл-центру; 3D – три ограничения на значения измерений: выборка событий заданного класса и длительности, относящихся к заданному колл-центру. Длительность выгрузки, как продемонстрировал эксперимент, мало зависела от количества ограничений.

Наиболее важным результатом этого эксперимента стало подтверждение существенно более быстрой (до трех десятичных порядков) выгрузки данных из БД, организованной по принципу ROLAP. Проигрыш HOLAP в скорости, по нашему мнению, связан с неоптимизированным доступом ко множеству событий, хранящихся в реляционной БД: многомерный куб хранит списки идентификаторов событий в оперативной памяти, однако каждое событие извлекается из реляционной БД по отдельности. При использовании подхода ROLAP для извлечения событий используется SQL-запрос, содержащий условия выборки, при выполнении которого оптимизатор запросов может принять решение об использовании индексов, многоблочных чтений и других средств повышения вычислительной эффективности.

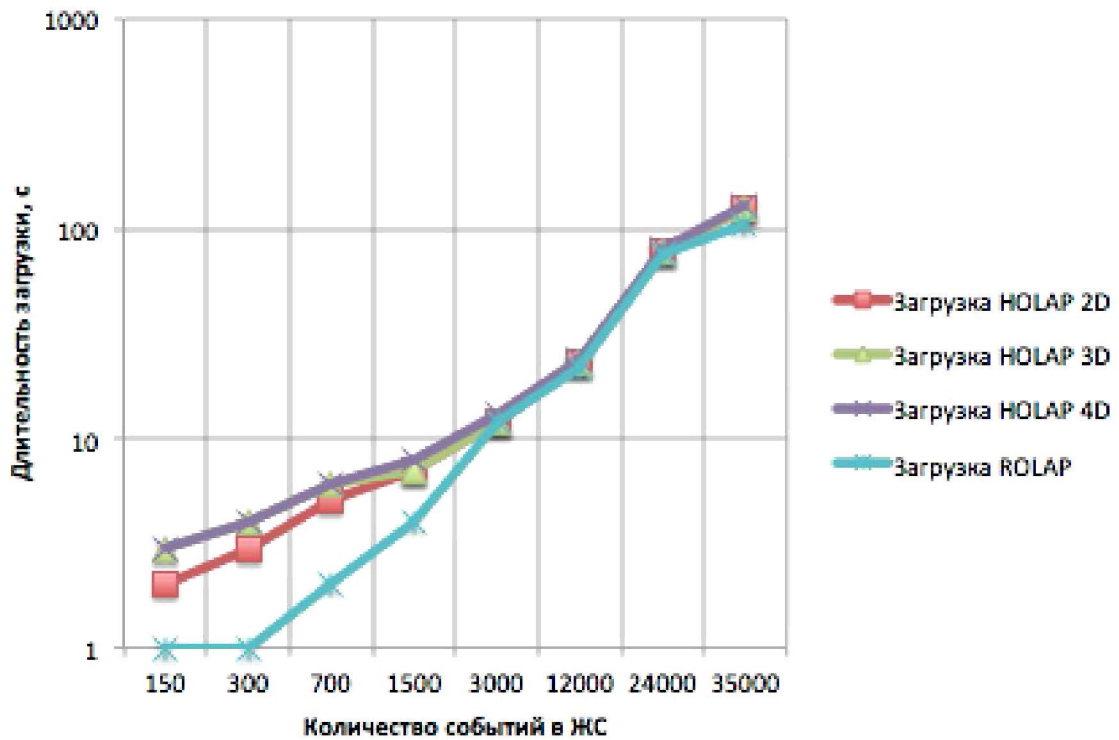


Рис. 6. Длительность загрузки ЖС в БД

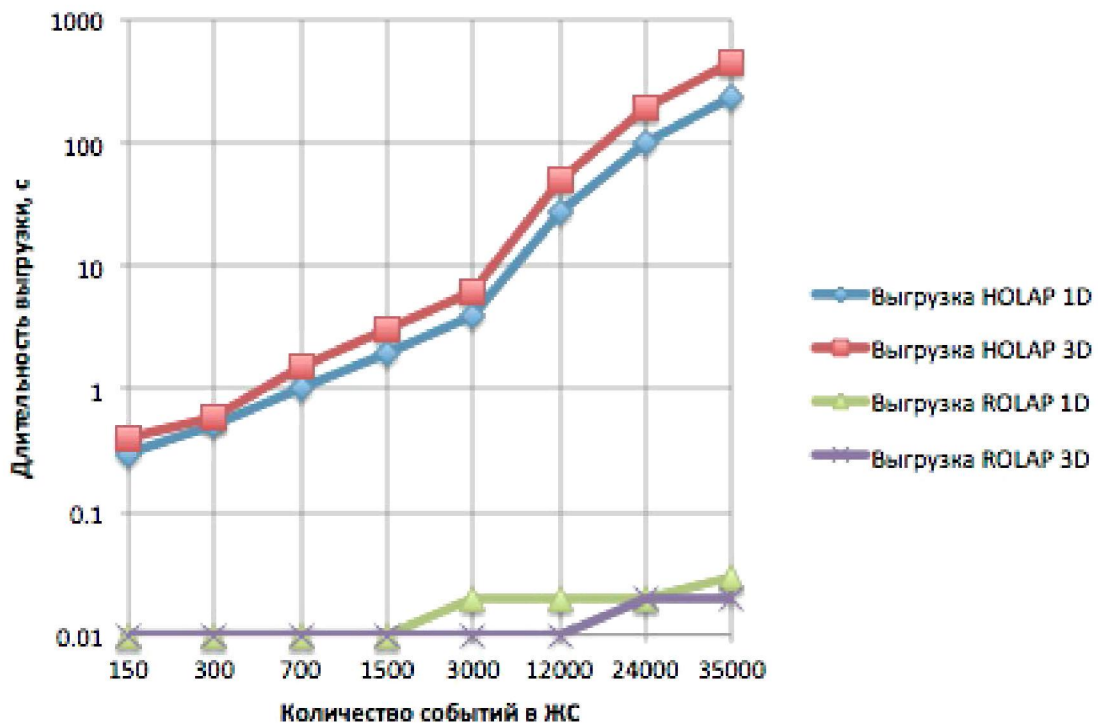


Рис. 7. Длительность выгрузки ЖС из БД

**Заключение.** В статье описан метод многомерного хранения журналов событий для извлечения и анализа процессов, основанный на подходе ROLAP. Этот метод позволяет оперативно визуализировать модели подпроцессов или одного процесса в едином интерфейсе. Предложенный метод реализован в виде программного продукта на языке Scala с использованием СУБД PostgreSQL 9.2. Хранение ЖС в БД обеспечивает возможность оперативного анализа данных и построения нескольких моделей с задаваемыми пользователем условиями отбора событий и/или трасс без повторных обращений к файлу журнала.

Полученные экспериментальные данные демонстрируют высокую скорость получения ЖС из БД и приемлемую скорость их загрузки в БД.

#### Список литературы

1. Брумштейн Ю. М. Анализ объектов и методов выбора оптимальных совокупностей решений для задач планирования в условиях ресурсных ограничений и рисков / Ю. М. Брумштейн, Д. А. Тарков, И. А. Дюдиков // Прикаспийский журнал: управление и высокие технологии. – 2013. – № 3. – С. 169–179.
2. Крэнке Д. Теория и практика построения баз данных / Д. Крэнке. – 9-е изд. – Санкт-Петербург : Питер, 2004. – 864 с.
3. Мальвина А. С. Автоматизация, диспетчеризация и информатизация высокотехнологичных медучреждений как средство повышения эффективности их работы / А. С. Мальвина, Ю. М. Брумштейн, Е. В. Скляренко, А. Б. Кузьмина // Прикаспийский журнал: управление и высокие технологии. – 2014. – № 1. – С. 122–138.
4. Eder J. A Data Warehouse for Event Logs / J. Eder, G. E. Olivotto, W. Gruber // Proceedings of First International Conference on Engineering and Deployment of Cooperative Information Systems (EDCIS 2002). Lecture Notes in Computer Science. – China, Beijing, 2002. – Vol. 2480. – P. 1–15.
5. La Rosa M. Congurable Multi-Perspective Business Process Models / M. La Rosa, M. Dumas, A. ter Hofstede, J. Mendling // Information Systems. – 2011. – № 36 (2). – P. 313–340.
6. Mamaliga T. Realizing a Process Cube Allowing for the Comparison of Event Data. Master's thesis / T. Mamaliga. – Eindhoven : Eindhoven University of Technology, 2013.
7. Ribeiro J. T. S. Event Cube: Another Perspective on Business Processes / J. T. S. Ribeiro, A. J. M. M. Weijters // Proceedings of 19th International Conference on Cooperative Information Systems. Lecture Notes in Computer Science. – Berlin : Springer-Verlag, 2011. – Vol. 7044. – P. 274–283.
8. van der Aalst W. M. P. Process Cubes: Slicing, Dicing, Rolling Up and Drilling Down Event Data for Process Mining / W. M. P. van der Aalst // Proceedings of Asia Pacific Conference on Business Process Management (AP-BPM 2013). Lecture Notes in Business Information Processing. – Berlin : Springer-Verlag, 2013. – Vol. 159. – P. 1–22.
9. van der Aalst W. M. P. Process Mining: Discovery, Conformance and Enhancement of Business Processes / W. M. P. van der Aalst. – Springer, 2010. – 349 p.
10. XES. – Available at: <http://www.xes-standard.org> (accessed 01.12.2013).

#### References

1. Brumshteyn Yu. M., Tarkov D. A., Dyudikov I. A. Analiz obektov i metodov vybora optimalnykh sovokupnostey resheniy dlya zadach planirovaniya v usloviyakh resursnykh ogranicheniy i riskov [The analysis of objects and methods of optimum choice for decision sets in conditions of resource restrictions and risks]. *Prikaspiyskiy zhurnal: upravlenie i vysokie tekhnologii* [Caspian Journal: Management and High Technologies], 2013, no. 3, pp. 169–179.
2. Krenke D. *Teoriya i praktika postroeniya baz dannyykh* [Database Processing: Fundamentals, Design and Implementation], 9<sup>th</sup> ed. Saint Peterburg, Piter, 2004. 864 p.
3. Malvina A. S., Brumshteyn Yu. M., Sklyarenko Ye. V., Kuzmina A. B. Avtomatizatsiya, dispetcherizatsiya i informatizatsiya vysokotekhnologichnykh meduchrezhdeniy kak sredstvo povysheniya effektivnosti ikh raboty [Automation, scheduling and informatization of high-tech medical facilities as a means of improving their performance]. *Prikaspiyskiy zhurnal: upravlenie i vysokie tekhnologii* [Caspian Journal: Management and High Technologies], 2014, no. 1, pp. 122–138.

4. Eder J., Olivotto G. E., Gruber W. A Data Warehouse for Event Logs. *Proceedings of First International Conference on Engineering and Deployment of Cooperative Information Systems (EDCIS 2002). Lecture Notes in Computer Science*. China, Beijing, 2002, vol. 2480, pp. 1–15.
5. La Rosa M., Dumas M., ter Hofstede A., Mendling J. Congurable Multi-Perspective Business Process Models. *Information Systems*, 2011, no. 36 (2), pp. 313–340.
6. Mamaliga. T. *Realizing a Process Cube Allowing for the Comparison of Event Data. Master's thesis*. Eindhoven University of Technology, 2013.
7. Ribeiro J. T. S., Weijters A. J. M. M. Event Cube: Another Perspective on Business Processes. *Proceedings of 19th International Conference on Cooperative Information Systems. Lecture Notes in Computer Science*. Berlin, Springer-Verlag, 2011, vol. 7044, pp. 274–283.
8. van der Aalst W. M. P. Process Cubes: Slicing, Dicing, Rolling Up and Drilling Down Event Data for Process Mining. *Proceedings of Asia Pacific Conference on Business Process Management (AP-BPM 2013). Lecture Notes in Business Information Processing*, Berlin, Springer-Verlag, 2013, vol. 159, pp. 1–22.
9. van der Aalst W. M. P. *Process Mining: Discovery, Conformance and Enhancement of Business Processes*. Springer, 2010. 349 p.
10. XES. Available at: <http://www.xes-standard.org> (accessed 1 December 2013).

УДК 621.391.677: 519.711.3

## **ПОСТРОЕНИЕ ГЕОМЕТРИЧЕСКОЙ МОДЕЛИ ИЗЛУЧЕНИЯ ЗЕРКАЛЬНОЙ ПАРАБОЛИЧЕСКОЙ АНТЕННЫ**

*Шишов Максим Владимирович*, аспирант, Пензенский государственный университет, 440026, Российская Федерация, г. Пенза, ул. Красная, 40, e-mail: [admin@r4f.su](mailto:admin@r4f.su)

*Юрков Николай Кондратьевич*, доктор технических наук, Пензенский государственный университет, 440026, Российская Федерация, г. Пенза, ул. Красная, 40, e-mail: [yurkov\\_nk@mail.ru](mailto:yurkov_nk@mail.ru)

*Якимов Александр Николаевич*, доктор технических наук, Пензенский государственный университет, 440026, Российская Федерация, г. Пенза, ул. Красная, 40, e-mail: [admin@r4f.su](mailto:admin@r4f.su)

Для исследования излучения зеркальной параболической антенны (ЗПА) методом математического моделирования необходимо построить её геометрическую модель, заменяющую излучающую поверхность антенны многогранником. При этом поле излучения модели антенны представляется как суперпозиция полей, формируемых токами, протекающими по граням модели с учетом амплитуд и фаз волн, распространяющихся от этих граней. Отличие многогранника от параболоида приводит к возникновению фазовых ошибок, минимизация которых является важной задачей. Авторами предложен алгоритм построения геометрической модели излучающей поверхности ЗПА, основанный на методе триангуляции Делоне, позволяющем произвести дискретизацию излучающей поверхности антенны в виде совокупности равносторонних плоских треугольных конечных элементов; получить совокупности узловых точек, образующих эти конечные элементы и, соответственно, координаты узлов. Этих данных оказывается достаточно для проведения расчета как полей, формируемых конечными элементами излучающей поверхности антенны, так и для нахождения их суперпозиции с учетом пространственной ориентации этих элементов и векторного характера электромагнитного поля. Дана оценка погрешности представленной модели. Предложенный алгоритм, основанный на методе триангуляции Делоне, обеспечивает выполнение требования к максимально допустимой фазовой ошибке при минимальных вычислительных затратах. Это позволяет использовать модель не только при расчете характеристик излучения ЗПА, но и для оценки влияния внешних воздействий на конструкцию антенны конечно-элементным методом.

**Ключевые слова:** излучающая поверхность, модель, антенна, триангуляция, конечный элемент, пространственная ориентация, суперпозиция, электромагнитное поле, фазовая ошибка, погрешность