



данные на уровне хранения. Встроенный язык программирования Cache Object Script (COS) является развитием языка программирования MUMPS. Помимо COS, Cache предоставляет разработчикам API для использования объектного и SQL-доступа к одним и тем же данным.

В настоящее время сформулирована задача анализа использования модели «сущность-связь» в объектно-реляционных базах данных, включающая перечисленные вопросы, а также в качестве объектно-реляционных СУБД выбраны Cache и Oracle, на которых будет производиться анализ модели «сущность-связь».

### Литература

1. Дейт, К. Введение в системы баз данных. Шестое издание. [Текст] / К. Дейт. – Москва, Санкт-Петербург: изд-во Вильямс, 1999. – 846 с.
2. Коннолли, Т. Базы данных. Проектирование, реализация и сопровождение. Теория и практика. Третье издание. [Текст] / Т. Каннолли, К. Бегг. – Москва, Санкт-Петербург: изд-во Вильямс, 2003. – 1440 с.

Е.И. Чигарина, Ю.С. Чуркина

## ФОРМУЛИРОВКА И МЕТОДИКА РЕШЕНИЯ ЗАДАЧИ СРАВНИТЕЛЬНОГО АНАЛИЗА ИСПОЛЬЗОВАНИЯ СРЕДСТВ ГЕНЕРАЦИИ ОТЧЕТОВ В РЕЛЯЦИОННЫХ БАЗАХ ДАННЫХ

(Самарский национальный исследовательский университет  
имени академика С.П. Королева)

В настоящее время для облегчения и ускорения процессов, связанных с обработкой большого количества данных, создаются системы, основанные на реляционных СУБД. Такие системы эффективно справляются с задачами учета, контроля и хранения информации. Однако реляционная структура не позволяет решать задачи анализа и представления имеющейся информации, необходимой для принятия решений.

Для решения этих проблем применяются генераторы отчетов, которые позволяют создавать информативные и удобные отчеты любого уровня сложности. В связи с большим количеством представленных на рынке систем генерации отчетов, пользователю бывает достаточно сложно выбрать необходимый именно ему генератор отчетов, так как все они обладают различными возможностями по созданию отчетов, а также отличаются своей производительностью.

Главной задачей данной работы является проведение сравнительного анализа использования средств генерации отчетов в реляционных базах данных.

Генератор отчетов представляет собой программу или библиотеку, позволяющую представить информацию в удобочитаемом структурированном виде. Другими словами сделать из данных информацию (документ, отчет), которую можно распечатать или сохранить в различных электронных форматах. Средств-



ва генерирования отчетов могут использоваться как в составе программ, программных систем и комплексов, так и самостоятельно (для анализа имеющихся данных безотносительно формирующих их систем).

Отчет, с точки зрения информационных систем, представляет собой документ, содержимое которого динамически формируется на основе информации, содержащейся в базе данных. Это могут быть данные из одной или нескольких таблиц или запросов, а также данные из одной или нескольких записей. Кроме того, в таком документе часто присутствуют и другие составляющие, например, заголовок, находящийся в начале отчета, колонтитулы, повторяющиеся на каждой странице, области, в которых располагаются фактические или агрегатные данные.

Не стоит путать понятия отчета и запроса в СУБД, так как это совершенно разные объекты. В реляционных базах данных основным объектом хранения является таблица, поэтому все SQL-запросы представляют собой операции над таблицами. Это может быть запрос на выборку данных из базы, либо обращение к базе данных, которое приводит к изменению этих данных. В соответствии с тем, какие изменения происходят в базе данных, различают следующие типы запросов: на обновление, на добавление, на удаление, перекрестный запрос, создание таблиц. Наиболее часто используемыми являются запросы на выборку. В результате запроса на выборку создается временная результирующая таблица, в которой отображаются только нужные по условию запроса данные из базовых таблиц. Отчеты – это объекты, с помощью которых данные выдают на принтер в удобном и наглядном виде. Запрос выбирает данные из БД или изменяет их. Отчет же это текстовое или графическое представление выбранных данных.

Сравнение средств генерации отчетов будет происходить по критериям функциональных возможностей и по скорости выполнения запросов. Скорость выполнения запросов зависит от алгоритмов выборки данных и от объема обрабатываемых данных. Основной функцией любого генератора отчетов является выборка запрашиваемых параметров за определённый промежуток времени. Стандартным способом получения необходимых данных является создание запроса, соединяющего в себе результаты  $N$  подзапросов, каждый из которых возвращает один из параметров. Однако вычисление всех операций соединения оказывается узким местом в запросе при увеличении числа запрашиваемых параметров и временного интервала.

Среди критериев функциональных возможностей будем выделять следующие:

- возможность построения однотипных отчетов по шаблону;
- возможность построения отчетов любой сложности (нестандартных отчетов);
- поддержка различных видов СУБД;
- возможность предварительного просмотра печати готового отчета;
- возможность вывода на печать;
- возможность перевода отчета в различные форматы данных (Word, Excel, PDF, HTML );



- наличие средств создания вычисляемых полей и параметризации отчетов;
- наличие средств работы с графикой (вставка изображений, представление отчета в виде диаграмм и т.д.).

В настоящее время разработана автоматизированная информационная система, в базе данных которой хранится большой объем информации, которую необходимо представлять в различных видах отчетности. На основе данной системы и будет проводиться сравнительный анализ и тестирование различных средств генерации отчетов.

### Литература

1. Определение генератора отчетов [Электронный ресурс].– [https://ru.wikipedia.org/wiki/Генератор\\_отчетов](https://ru.wikipedia.org/wiki/Генератор_отчетов)
2. Классификация отчетов [Электронный ресурс].– <http://compress.ru/article.aspx?id=10013&iid=418>
3. Скорость выполнения запросов SQL [Электронный ресурс].– <http://ts-soft.ru/blog/sql-optimization-1>

А.С. Широканев

## ВЕКТОРНЫЙ АЛГОРИТМ ПАРАМЕТРИЧЕСКОЙ ИДЕНТИФИКАЦИИ КРИСТАЛЛИЧЕСКИХ РЕШЁТОК НА ОСНОВЕ ГРАДИЕНТНОГО МЕТОДА С ПОСТОЯННЫМ ШАГОМ

(Самарский университет)

### Введение

В современное время кристаллические наноструктуры используются во многих предметных областях: медицина, строительство, электротехника. Исследование структур кристаллов является неотъемлемо важной задачей. В частности, анализ кристаллических структур обеспечивается решением задачи параметрической идентификации кристаллических решёток [1-8]. С решением данной задачи связаны работы [1-6], в которых представлены существующие алгоритмы оценивания параметров моделей кристаллических решёток.

Важной характеристикой алгоритмов параметрической идентификации является точность идентификации, которая может быть увеличена различными модификациями известных алгоритмов. В работе [8] представлен алгоритм параметрической идентификации, который позволяет улучшить точность параметрической идентификации по сравнению с алгоритмами, представленными в работах [2, 3, 4]. Недостатком разработанного алгоритма является высокая вычислительная сложность.

Цель настоящего исследования заключается в разработке векторного алгоритма параметрической идентификации кристаллических решёток, позволяющего устранить недостаток высокой вычислительной сложности соответствующего последовательного алгоритма.