

- весь ЖЦ ПО (сохраняет модель приложения в исполняемом файле, на этапе выполнения приложения использует эту модель для управления бизнес-уровнем, для контроля целостности объектного пространства, для управления взаимодействием бизнес-уровня с уровнем данных и графическим интерфейсом).
2. Концепцию MDA реализует средство разработки ПО ИС Bold for Delphi.
  3. Использование Bold for Delphi является целесообразным и оптимальным вариантом для автоматизации разработки приложений и информационных систем.

## РАЗРАБОТКА ОБЪЕКТНО-РЕЛЯЦИОННОГО ПРЕОБРАЗОВАТЕЛЯ ДЛЯ ЯЗЫКА ПРОГРАММИРОВАНИЯ PERL

Зубань С.А.

Одной из главных задач, стоящих перед разработчиком сложных интернет-систем является проектирование и работа с базой данных. В настоящее время наиболее распространёнными СУБД являются *реляционные базы данных*. Отличительной особенностью реляционной модели является минимальная избыточность при представлении данных (при правильном проектировании базы данных). Но реляционная модель имеет и ряд отрицательных характеристик. Одной из них является абсолютно разная структура представления данных в такой базе в сравнении с объектами объектно-ориентированного языка программирования, что делает задачу работы с базой данных довольно сложной.

Для решения этой проблемы существуют *объектно-реляционные преобразователи*, которые позволяют максимально абстрагироваться от того факта, что данные хранятся в реляционной базе данных и оперировать уже исключительно объектами объектно-ориентированного языка программирования.

Представляет интерес использование ОРП в многопользовательских интернет-приложениях, например, в системах дистанционного

обучения. К сожалению, с увеличением количества студентов и объема учебных материалов проявляется один из основных недостатков ОРП - низкая производительность, обусловленная дополнительными действиями по формированию реляционных запросов на основе объектной модели и по преобразованию полученных данных из реляционной в объектную модель. Задачей данной работы является разработка ОРП повышенного быстродействия с сохранением основной функциональности.

Один из возможных путей для повышения быстродействие ОРП является *кэширование полученных данных*, и не менее важным — *кэширование сформированных запросов*. Логично предположить, что большинство операций, выполняемых с базой данных практически одинаковы и все запросы можно разбить на максимально возможное число частей и хранить эти части в кэше. При повторном выполнении операции используется запрос хранящийся в кэше, что позволяет существенно сэкономить время работы преобразователя.

Главной сложностью в таком подходе является выбор метода кэширования, т.е. информации, на основе которой будет происходить разбиение SQL-запроса на части. В качестве ключа кэша в разрабатываемом преобразователе используется информация об объекте и выполняемом действии.

Сам кэш реализуется с помощью хеш-таблиц, которые являются встроенным типом данных языка perl и все операции с хешем выполняются за время  $O(1)$ , что делает эту структуру данных идеальной для представления кэша SQL-запросов.

### Литература

1. Tangram Orthogonal Persistence. - <http://www.tangram-persistence.org/>
2. <http://www.alzabo.org/>
3. Всемирная библиотека модулей для языка perl. - <http://www.perl.com/CPAN/>
4. Perl Object-Oriented Persistence - <http://poop.sourceforge.net/>