

- «модель данных», отвечающая за хранение и кэширование данных;
- «менеджер соединений», выполняющий соединение с сервером, формирование запросов и обработку результатов;
- «менеджер взаимодействий», обеспечивающий взаимодействие между подсистемами и определяющий логику работы приложения.

Кроме того, можно выделить подсистему навигации и визуализации данных.

Ряд стандартных компонент Flash заменен на более эффективные с точки зрения данной задачи, разработано несколько дополнительных компонент.

Приложение является универсальным и может быть использовано для решения разнообразных задач. В данный момент система используется для выбора интересующего региона при поиске объектов недвижимости в штате Нью-Йорк. Система разработана в рамках проекта компании ЗАО «Энтерра».

### **Ядровая модель данных**

*А.С. Покидов*

*ЗАО «Энтерра», г. Барнаул*

Особенность многих современных систем хранения данных заключается в том, что они не всегда могут полностью решить задачу по целостности данных, по отношениям между данными, переключая часть решения на внешние системы. Это означает необходимость для каждой внешней системы, работающей с системой хранения данных реализовывать одно и то же решение. Внешние системы могут быть самыми разнообразными и написанными на разных языках программирования, что не позволяет использовать одни и те же библиотеки.

Несмотря на то, что реляционные СУБД существуют много времени, они остаются наиболее популярными для решения прикладных задач. Большой интерес представляют объектно-ориентированные СУБД. Однако объектно-ориентированная модель имеет существенный недостаток – ООСУБД требуют больше вычислительных ресурсов, чем реляционные, что не позволяет применять их там, где требования по расходуемым ресурсам играют существенную роль.

Работа направлена на создание эффективной модели данных позволяющей решать широкий спектр прикладных задач. Ее основой является проведенное автором математическое исследование хранения данных в табличном виде, которое показало, что реляционная модель

не использует в полной мере потенциал табличной формы описания сущностей, т.к. не включает в себя целый ряд возможностей, функций, которые могли бы в значительной мере облегчить решение множества задач.

Идея ядровой модели – это идея расширения реляционной модели с возможностью дальнейшего развития ядровой модели данных.

При разработке автором новой модели данных акцент смещен в сторону связей между сущностями, этот фактор, а также стремление к однообразию привело к новой идее построения структуры базы данных и использованию для этого гиперграфов. Основная идея формирования структуры ядровой СУБД – «идти наоборот».

В реляционных СУБД сначала определяются таблицы для хранения данных об экземплярах сущностей, а потом уже связи между таблицами. Первым шагом в ядровой модели данных является выделение связей. Второй шаг – на основе связей и дополнительных описаний производится создание структур хранения данных об экземплярах сущностей.

Результатом построения структуры является гиперграф. Отдельные его части несут определенную смысловую нагрузку. Например, к вершинам гиперграфа «привязаны» типы данных, таким образом, вершина становится не просто узлом гиперграфа, а «ядром», которое представляет тип и связи между сущностями по данному типу. Гиперребро также можно назвать «узлом» гиперграфа, и также как и вершина, гиперребро имеет смысловую нагрузку, но в отличие от вершины смысловой нагрузкой является сущность, что позволяет говорить о гиперребре как о «ядре» сущности.

В данный момент разрабатывается СУБД, которая реализует ядровую модель данных. Разрабатываемая СУБД – это комбинация реляционной СУБД (для хранения данных в табличном виде без связей) и оболочки, поддерживающей ядровую модель данных.

## **Разработка информационной системы тестирования студентов в бийском технологическом институте**

***Н.В. Попов***

*БТИ (филиал) АлтГТУ, г. Бийск*

В настоящее время особую популярность приобретают средства и методы автоматизированной оценки знаний. В последнее время в качестве метода оценки знаний важную роль играет применение компьютерного тестирования, которое позволяет автоматизировать процесс