

Изоморфизмы и кроссплатформенные приложения

А.В. Клетинин

В процессе разработки программных продуктов часто возникает необходимость обеспечить работоспособность программ на широком спектре программных и аппаратных платформ. В частности, вполне естественным выглядит требование работоспособности программ на широком спектре систем управления базами данных. И использование алгебраических конструкций позволяет выполнить это требование.

В данной работе рассматривается модель, позволяющая создавать программы, способные работать на самых различных системах управления базами данных. Она была успешно реализована в процессе разработки программного продукта IBM QMF for WebSphere, и показала свою перспективность и полезность для разработчиков.

В основе модели лежит алгебраическое понятие изоморфизма систем. Для удовлетворения требования работоспособности на широком спектре платформ в модели вводится набор систем объектов, которые оказываются изоморфными алгебрами одной и той же сигнатуры. С точки зрения программы это дает возможность при помощи выбирающей функции легко переключаться с использования функциональности одной платформы, на использование функциональности другой платформы.

Приведем наиболее важные следствия применения предлагаемой модели.

1. Инструкции SQL оказываются отделены от остального кода программы, что упрощает разработку кода, его дальнейшее сопровождение.
2. Обеспечивается возможность автоматизированного тестирования запросов SQL (как на предмет синтаксической корректности, так и на предмет правильности работы).
3. Обеспечивается простота процесса адаптации программы к различным СУБД
4. Обеспечивается эффективное использование дополнительных возможностей, предоставляемых СУБД, для выполнения SQL-запросов.

Данная модель должна быть интересна разработчикам ПО, поскольку ее применение позволяет упростить код, сократить затраты на разработку и поддержку программных продуктов, ориентированных на широкий спектр платформ, повысить эффективность труда специалистов, работающих в рамках проекта. А относительная самостоятельность и достаточная гибкость модели позволяет думать о перспективности создания независимого программного продукта (средства разработки), позволяющего получать реализации данной модели для конкретных языков программирования и конкретных систем запросов.

Уральский государственный университет
E-mail: **Aleksandr.Klepinin@usu.ru**

11 октября 2004