

## СРАВНИТЕЛЬНЫЙ АНАЛИЗ ВОЗМОЖНОСТЕЙ ДВУХ ВЕДУЩИХ ПРОМЫШЛЕННЫХ СУБД

А.С. Деревянко  
(представил д.т.н., проф. Л.М. Любчик)

Сравниваются реляционные возможности двух ведущих промышленных СУБД - Oracle 8i и IBM DB2 v.7. Сравнение проводилось решением одинаковых задач в двух указанных средах. Рассматриваются встроенные базовые типы данных, запросы, представления, триггеры, процедуры, управление транзакциями и управление доступом. Показывается, что достоинства рассмотренных СУБД не во всем симметричны, но "в среднем" равноценны.

В предлагаемой статье приводятся результаты сравнительного анализа СУБД Oracle 8i и IBM DB2 v.7. Сопоставление этих двух СУБД проведено при подготовке учебных материалов, в ходе которой одни и те же задачи решались с использованием того и другого продуктов. При этом обнаружилось выгодная специфика учебных задач, состоящая в том, что, во-первых, имелась возможность подобрать задачи, использующие то или иное рассматриваемое нами свойство «в чистом виде», а во-вторых, по необходимости была достигнута широта охвата, невозможная при решении прикладных задач.

По данным Gartner Group фирмы Oracle и IBM являются лидерами в производстве СУБД и их доля на этом рынке в 2000 году составила 33.8% и 30.1% соответственно ([http://www.gartnergroup.com/5\\_about/press\\_room/pr20010523b.html](http://www.gartnergroup.com/5_about/press_room/pr20010523b.html)). На отечественном рынке сложилась несколько иная картина. Здесь, наряду с Oracle, лидируют Informix и Microsoft. Однако, расширяющаяся экспансия DB2 на платформах Unix и Windows NT и покупка фирмой IBM в 2001 году бизнеса баз данных у Informix позволяют прогнозировать некоторое изменение этой картины в ближайшем будущем.

При жестких конкурентных отношениях, сложившихся между лидерами СУБД, наблюдается явный дефицит публикаций, в которых их возможности сопоставлялись бы объективно и без предвзятости. Нам известна единственная работа такого рода [1]. Однако, она рассматривает эти продукты (точнее – многофункциональные семейства продуктов, а не только СУБД) на «макроуровне», не рассматривая детали реализации отдельных возможностей. Мы же в данной работе ограничиваемся вопросами использования языка SQL, причем в рамках «традиционной» реляционной модели (например, [2]). Объектно - реляционным свойствам,

которые также значительно развиты в этих СУБД, планируется посвящать отдельную публикацию.

Обе рассматриваемые СУБД формально соответствуют стандарту ANSI SQL/92 Entry Level. Однако, они обладают отдельными возможностями, соответствующими более высокому уровню стандарта или вообще стандартом не предусмотренными.

**Описание данных.** DB2 поддерживает стандартный набор встроенных типов данных, принятый также и в большинстве промышленных СУБД. Oracle обеспечивает единое представление всех числовых типов собственным типом NUMBER(n,p), но при этом воспринимает и объявления со стандартными числовыми типами и автоматически переводит их в представление NUMBER. Единое представление числовых типов создает в Oracle возможность менять параметры их представления (n,p) в операторе ALTER TABLE.

DB2 поддерживает стандартные типы даты/времени: DATE, TIME, TIMESTAMP. В Oracle есть единственный тип – DATE, соответствующий стандартному TIMESTAMP, других типов Oracle «не понимает». Что касается средств работы с датой/временем, то в наших СУБД они существенно различны, функционально полны и не слишком просты в употреблении. Нам представляется наиболее интересной возможность арифметики даты/времени с использованием именованных временных единиц в DB2.

Преобразование типов в DB2 выполняется стандартными функциями, имена которых совпадают с именами целевых типов, и стандартной же операцией универсального преобразования CAST. В Oracle же существуют только три функции форматного преобразования – TO\_CHAR, TO\_NUMBER, TO\_DATE, которые функционально богаче стандартных, но сложнее в употреблении.

Ограничения целостности, задаваемые в описании данных, в обеих СУБД в целом одинаковы. DB2 предоставляет большие возможности выбора при определении действий, выполняемых для обеспечения ссылочной целостности при удалении / изменении строк.

**Запросы.** Естественно, обе рассматриваемые СУБД сохраняют реляционную полноту языка SQL, что позволяет сформулировать запрос на выборку «чего угодно». Дialect SQL DB2 по сравнению с диалектом Oracle обладает некоторыми дополнительными возможностями SQL (в основном – соответствующими стандарту), что позволяет формулировать запросы зачастую более компактные и более понятные. Среди таких преимуществ SQL DB2 наиболее существенными, на наш взгляд, являются следующие:

- возможность использования скалярных подзапросов в качестве столбцовых элементов фразы SELECT;
- условное выражение - CASE;

- отсутствие ограничений на использование подзапросов в предикатах сравнения (в частности, могут сравниваться и результаты двух подзапросов, что невозможно в Oracle).

Соединение таблиц в Oracle выполняется по предикату равенства значений в столбцах во фразе WHERE. В DB2 наряду с этим имеется операция JOIN, определяющая соединение уже во фразе FROM. Для выполнения внешних соединений в Oracle введена специальная операция (+), позволяющая определить левое или правое внешнее соединение, но не оба вместе. В DB2 расширение той же операции JOIN обеспечивает левое, правое, а также и полное внешнее соединение.

Обе СУБД поддерживают также иерархические запросы, выполнение которых, как показано в [2], выходит за рамки реляционной модели. В Oracle для этого введен особый синтаксис, включающий специальные ключевые слова CONNECT BY, START WITH, PRIOR. DB2 решает ту же задачу применением временных представлений с единственным новым ключевым словом WITH. Хотя иерархические запросы в DB2 получаются несколько менее компактными, чем в Oracle, на наш взгляд, они более понятны для программиста, знакомого с рекурсией. Кроме того, функциональность временных представлений не исчерпывается только иерархическими выборками. Они очень удобны также и как средство сокращения записи в сложных запросах.

**Представления.** В части создания представлений возможности обеих СУБД практически одинаковы. Они соответствуют стандартным возможностям оператора CREATE VIEW и различаются лишь рассмотренными выше различиями в формулировании тех запросов, которые определяют представление. Обе СУБД расширяют стандартные ограничения на изменяемость представлений, прежде всего тем, что разрешают изменять представления, в определении которых содержатся вложенные запросы. DB2 включает в число изменяемых также представления, в определении которых содержится объединение (UNION ALL) двух и более базовых таблиц. Oracle же позволяет изменять некоторые представления, содержащие в определении соединение таблиц в тех случаях, когда ключ представления является также и ключом базовой таблицы. Хотя эта возможность Oracle довольно ограничена и далеко не проста для понимания, ее важность бесспорна.

**Триггеры.** Стандартный синтаксис оператора CREATE TRIGGER обе СУБД дополняют своими модификациями.

Oracle дает возможность создавать триггеры не только для базовых таблиц, но и для представлений (как следствие более широких возможностей изменяемости представлений). Введенное в Oracle новое условие активизации INSTEAD OF (только для триггеров представлений) дает возможность выполнять некоторые изменения даже и для тех представлений, которые формально не являются изменяемыми. В Oracle имеется ряд ограничений на сложность дополнительного условия выполнения

триггера (фраза WHEN), которые, однако, компенсируются тем, что действия триггера задаются блоком операторов PL/SQL, что безусловно расширяет логические возможности триггера.

В DB2 существует ограничение на выполнение операторов модификации таблиц в BEFORE-триггерах, которое, однако, не представляется нам существенным ограничением на функциональность. В действиях триггера возможна ссылка не только на старые и новые значения изменяемой строки, как и в Oracle, но и на старые и новые значения изменяемой таблицы. Действия триггера представляются одним оператором SQL или блоком операторов, выполняющимся как атомарным. Список операторов, допустимых в триггере, расширяется операторами SET и SIGNAL SQLSTATE, но логические возможности действий триггера ограничены возможностями фразы WHEN (со сколь угодно сложным условием) и выражениями CASE.

**Процедурные расширения SQL.** На протяжении ряда лет безусловным преимуществом Oracle являлся процедурный язык PL/SQL с весьма широкой сферой применения. В DB2 v.7 также появляется процедурный язык, сфера применения которого в этой версии ограничивается только хранимыми процедурами. Хотя возможности процедурного языка DB2 пока существенно отстают от возможностей PL/SQL, в нем все же обеспечивается полный набор операторов процедурного программирования, объявление локальных переменных, работа с курсором. Принципиальным различием между ними представляется то, что PL/SQL часто позиционируется как самостоятельный язык программирования со встроенным SQL, а процедурный язык DB2 принципиально является расширением SQL и в базовых своих конструкциях (условия, выражения и т.п.) использует синтаксис и возможности SQL.

**Управление транзакциями.** Из четырех уровней изолированности, определяемых стандартом SQL, DB2 обеспечивает примерное или точное соответствие всем четырем (хотя их названия не совпадают со стандартными), а Oracle обеспечивает два стандартных уровня – READ COMMITTED и SERIALIZABLE. Однако, при формальном соблюдении требований стандарта, реализация их в двух СУБД совершенно различна, что приводит к различному поведению одинаковых транзакций в разных СУБД. В DB2 конфликт параллельного доступа всегда приводит к блокировке, в Oracle, как правило, читающая транзакция получает образ данных, зафиксированный на момент начала ее выполнения. DB2 применяет для обеспечения изолированности «классический» механизм блокировок, Oracle применяет многоверсионную модель. Различия в этих и других подходах к обеспечению изолированности давно обсуждается в литературе (например, [2]), но независимые авторы воздерживаются от прямых оценок. Обе СУБД одинаково реагируют на тупики, завершая одну из конфликтующих транзакций.

Отметим еще, что возможности управления изолированностью в Oracle можно считать несколько более гибкими за счет того, что уровень изолированности может устанавливаться не только для всего сеанса, но и для отдельной транзакции.

**Управление доступом.** Концепции управления доступом в двух рассматриваемых СУБД совершенно различны. Мы ограничимся лишь кратким изложением этих различий, избегая оценок. Oracle полностью берет на себя управление безопасностью. Следствием этого является то, что пользователи являются объектами базы данных. В DB2 управление доступом интегрировано с управлением безопасностью в операционной системе / сети, пользователи и группы пользователей регистрируются вне СУБД. Объектами СУБД являются лишь записи о привилегиях, предоставленных пользователям и группам. Отсюда появляется возможность возникновения "странной" ситуации: в среде управления (DB2 Command Center) можно создать нового пользователя и дать ему привилегии доступа к каким-то объектам, но этот пользователь не может пройти аутентификацию при соединении с базой данных, потому что он не зарегистрирован в системе.

Концепция Oracle представляет все управление привилегиями в практически одномерной модели. Существует только два типа привилегий (системные и объектные) и средства управления всеми привилегиями одного типа одинаковы (операторы GRANT / REVOKE). Впрочем и для двух типов привилегий они различаются только деталями.

В DB2 управление привилегиями более «типизировано». Для каждого класса объектов выделен свой набор привилегий. Для объектов верхнего уровня иерархии понятие привилегий заменяется понятием полномочий, управление которыми ведется только внешними по отношению к СУБД, средствами, а именно, включением пользователей в полномочные административные группы. Для неконтэйнерных объектов, типы привилегий хотя и различаются, но управление ими одинаково. Для этого используются те же операторы GRANT / REVOKE. Группирование привилегий ведется при помощи ролей (объектов базы данных) в Oracle и при помощи групп пользователей (управляемых вне СУБД) в DB2. В обоих случаях при этом обеспечивается одинаковая функциональность. В этой части следует также отметить принципиально различную трактовку термина экземпляр (instance) в двух СУБД. В Oracle экземпляр является только средой времени выполнения (набором процессов и областей памяти), выполняющей функции монитора одной базы данных. В DB2 экземпляр является хранимой конфигурацией, монитором нескольких созданных в нем баз данных. Эта возможность обеспечивает дополнительный уровень управления доступом, создавая иллюзию наличия нескольких СУБД в одной инсталляции.

В настоящей публикации не затрагивается вопрос о средствах разработки приложений, но упоминается о базовых инструментальных средствах в составе СУБД.

Среды управления – Oracle Navigator и DB2 Control Center практически одинаковы по возможностям и по эффективности функционирования. Среди несимметричных достоинств этих сред можно назвать возможность доступа к данным таблиц в Navigator и наличие функции SQL Assist в Control Center.

Среда интерактивного SQL DB2 Command Center безусловно богаче и комфортнее, чем Oracle SQL\*Plus. Однако, Command Center гораздо более ресурсоемкое приложение и при выполнении его на маломощных компьютерах могут возникать проблемы. В DB2 также имеется Command Line Processor и Command Window – интерактивные среды SQL без сервиза, но с минимальными требованиями к ресурсам.

Выводы из проведенных нами исследований в основном совпадают со сделанными в [1] применительно к другой области. Возможности двух рассмотренных нами СУБД во многом несимметричны, но их достоинства и недостатки компенсируют друг друга. Функциональные возможности DB2 и Oracle «в среднем» эквивалентны и не могут служить единственным основанием для выбора того или другого продукта. Можно с уверенностью утверждать, что любые заявления об абсолютном превосходстве одного продукта над другим являются безосновательными и у разработчиков информационных систем имеется возможность выбора.

Автор благодарит фирму Insart за предоставленное лицензионное программное обеспечение.

## ЛИТЕРАТУРА

1. Bischoff J. DB2 V.5 and Oracle 8. What They've Got, What They've Not // DB2 Magazine. – 1997. – V.2. – № 2. – P. 33 - 39.
2. Дейт К. Введение в системы баз данных. – К.: Диалектика, 1998. – 784 с.
3. Беренсон Х., Бернштейн Ф., Грей Д. И др. Критика уровней изолированности в стандарте ANSI SQL // Системы управления базами данных. – 1998. – № 2. – С. 68 - 79.

*Поступила в редколлегию 15.10.2001*

---