

А. Б. Л а ч и х и н а, А. В. М а з и н

**МЕТОДИКА РАЦИОНАЛЬНОЙ НАСТРОЙКИ  
БАЗ ДАННЫХ НА ПРИМЕРЕ СИСТЕМЫ  
“АНАЛИТИК”**

*Рассмотрены основные этапы подготовительных работ для проектирования системы сбора статистики и организации настройки базы данных. Приведены примеры их реализации для настройки производительности СУБД Oracle 9i и диагностические данные о работе анализируемой базы.*

**E-mail: [ablaze@newmail.ru](mailto:ablaze@newmail.ru)**

**Ключевые слова:** мониторинг, настройка баз данных, производительность, целостность, транзакция, автоматизированный сбор диагностических данных.

Эффективность функционирования корпоративных информационных систем (ИС) зависит от качества функционирования баз данных (БД), на основе которых они построены. Качество функционирования БД определяется множеством требований таких, как высокое быстродействие, малое время отклика, безопасность и независимость данных, простота их обновления, доступность для многих пользователей и др. Требования эти противоречивы. Например, стремление к высокой производительности приводит к снижению функциональных возможностей, в том числе и мер по обеспечению безопасности БД. Использование аналитических моделей для оптимизации предъявляемых требований сложно и малоэффективно.

Для поддержания требуемых параметров БД необходим мониторинг системы, в результате которого может быть получена информация, позволяющая своевременно реагировать на изменения в предметной области и в требованиях пользователей путем настройки БД и внесения соответствующих изменений.

Цель настоящей работы — оптимизация требований к качеству функционирования БД путем мониторинга, позволяющего выявить узкие места с точки зрения производительности и других параметров БД в рамках всей ИС.

Для достижения этой цели необходимо выявить все факторы, которые могут повлиять на точность проведения мониторинга, выбрать параметры, по которым оценивается состояние БД, и отработать методику ее анализа. В настоящей работе изложены подходы к решению этих задач на примере настройки производительности СУБД Oracle 9i.

Большинство коммерческих СУБД предоставляют в распоряжение администратора БД набор утилит, предназначенных для наблюдения за функционированием системы и ее настройки.

Так, вместе с сервером Oracle поставляется пакет диагностики, состоящий из двух сценариев: `utlstat.sql` и `utlestat.sql`. Начиная с версии сервера 8.1.6, фирма Oracle предоставляет одновременно еще и другое средство под общим названием `Statspack` — более современный аналог сценариев `Estat/Bstat` [1].

Однако эти средства имеют ряд недостатков. Во-первых, это избыточность данных. На практике нет необходимости знать и анализировать каждый предоставляемый количественный показатель. Часть параметров может вообще не понадобиться или будет необходима только для уточнения конкретной видимой проблемы. Для пользователя достаточно бывает десяти процентов всего отчета. Избыточность данных приводит к тому, что в тексте отчета довольно трудно разобраться.

Во-вторых, для каждой БД статистика хранится непосредственно в самой базе. Это вызывает дополнительную нагрузку на производственную БД и приводит к искажению результатов статистики.

К недостаткам также можно отнести неудачную документацию, ложность некоторых предпосылок, например о повышении производительности [2], на которые опиралась данная система анализа, недостаточную полноту списка требуемых параметров.

Пакет `Statspack` предоставляет ряд параметров, таких как число логических чтений, число измененных блоков данных, число физических чтений, число физических записей, число разборов SQL-выражений (мягких и жестких), общее число сортировок (на диске и в памяти), основных событий ожидания, возникших в системе, и др.

Но данный пакет является универсальным средством анализа. Поэтому он не предоставляет таких важных для анализа работы системы параметров, как распределение загрузки систем ввода-вывода сервера за выбранный период времени; распределение по времени загрузки центрального процессора, загрузки сети; число подключенных к системе пользователей; число выполненных транзакций; значение коэффициентов использования памяти в различные моменты времени; распределение операций по таблицам БД; анализ выполняемых в системе SQL-операторов и др. И, конечно, пакет `Statspack` не может предоставить параметры, учитывающие потребности предприятия при конкретном типе оценки.

Поэтому предприятиям, работающим с крупными БД, приходится самим разрабатывать необходимые средства диагностики. Далее приведены некоторые общие подходы к проектированию диагностических средств и организации анализа БД.

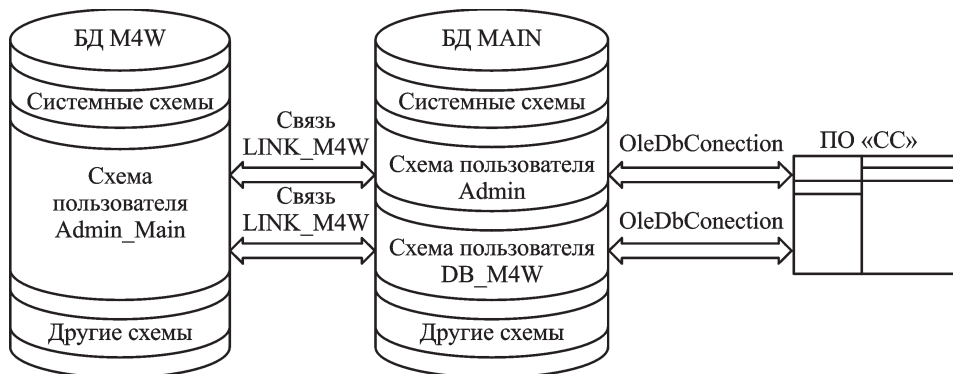
Проектированию системы сбора диагностических сведений и организации работ по проведению мониторинга должны предшествовать следующие этапы:

- 1) выявление причин анализа системы;
- 2) оценка и выделение существующих сложившихся подходов к анализу параметров системы;
- 3) выделение сложившихся проблем в работе ИС предприятия;
- 4) определение параметров, необходимых для оценки работы БД предприятия;
- 5) определение способов доступа к выбранным параметрам;
- 6) выбор инструментальных средств, используемых для разработки необходимого программного обеспечения (ПО);
- 7) разработка системы автоматизированного сбора и регистрации значений выбранных параметров (система сбора статистики).

На *первом* этапе необходимо выявить причины анализа системы, в зависимости от которых будет тот или иной способ проведения анализа. В качестве причин, побуждающих администратора к настройке, могут быть претензии пользователей, зависание оборудования, неустойчивая работа, высокая загрузка сервера, замена оборудования на более современное и др.

На *втором* этапе необходимо минимизировать дополнительную нагрузку на систему при организации анализа работы БД, что определяет архитектуру и схему подключения системы сбора статистики. Возможным вариантом решения этой проблемы является выполнение вычислений вне анализируемой БД и сервера, на котором она расположена. При таком подходе сервер Oracle используется как сервер приложений, в дополнение к которому необходимо разработать приложение, позволяющее просматривать собранные данные. Хранение собранных аналитических данных в одной специально выделенной базе упрощает работу администратора, особенно при анализе нескольких БД (от единиц до десятков).

Структурная схема системы сбора статистики, работающей по такой технологии, показана на рис. 1.



**Рис. 1. Структурная схема системы сбора статистики**

Если рассматривать представленную систему как трехзвенную архитектуру построения приложений, то программное обеспечение системы сбора статистики (ПО ССС) является клиентским. Центральная БД MAIN — это сервер приложений, через который происходит соединение клиента с анализируемой БД (БД M4W), представляющей третий уровень архитектуры. Сервер приложений отвечает также за выполнение основных расчетов и хранение собранных статистических данных.

Клиент посредством интерфейса, реализованного в пространстве имен OleDbConnection, подключается к центральной БД MAIN и передает команды, которые необходимо выполнить в анализируемой БД через связь LINK\_M4W. Ответ, полученный в результате выполнения запроса или команды, поступает по связи LINK\_M4W из анализируемой БД в центральную и затем по OleDbConnection передается клиентскому ПО, которое обрабатывает полученные данные и предоставляет их пользователю.

Для каждой анализируемой базы при ее регистрации в центральной БД автоматически создается схема пользователя DB\_M4W, в которой будут храниться все собранные данные. Через нее будет происходить соединение с анализируемой БД. Кроме этого, создаются табличное пространство (ТП) для хранения данных; набор процедур для получения диагностических данных; таблицы для хранения собранных параметров; связь с анализируемой БД (связь LINK\_M4W); схема администратора системы (схема Admin) для хранения ссылок и описаний анализируемых БД.

В анализируемой БД также необходимо создать схему пользователя (схему Admin\_Main), через которую будут взаимодействовать анализируемая и центральная базы. На этого пользователя в анализируемой БД будет зарегистрирована связь (db\_link) LINK\_M4W. Ему необходимо выделить права на создание сессии (привилегия CONNECT), права на получение данных из системных представлений (SELECT ANY DICTIONARY) и выполнение пакетов SYS.DBMS\_LOGMNR, SYS.DBMS\_SYSTEM (EXECUTE ON SYS.DBMS\_LOGMNR и EXECUTE ON SYS.DBMS\_SYSTEM).

Администратору предоставляются права на подключение к БД (CONNECT); права на создание и удаление пользователей в БД (CREATE USER, DROP USER), таблиц (CREATE TABLE, DROP TABLE), ТП (CREATE TABLESPACE, DROP TABLESPACE), процедур (CREATE PROCEDURE, DROP PROCEDURE); права на просмотр системных таблиц и представлений (SELECT ANY TABLE, SELECT ANY DICTIONARY); права на создание и удаление связей с БД (DROP PUBLIC DATABASE LINK, CREATE PUBLIC DATABASE LINK).

На *третьем* этапе необходимо выявить узкие места в организации работы БД в рамках всей ИС. Можно выделить следующие типичные негативные факторы в работе ИС:

- увеличение времени выполнения основных операций по выборке документов по сравнению с периодом, например полугодичной давности;

- увеличение времени выполнения операций по передаче данных. Например, в процессе развития системы объем передаваемых данных увеличился и временные затраты возросли с 3 до 16 ч. Ночная загрузка данных не укладывается в нерабочий период времени, что негативно влияет на производительность системы;

- увеличение времени выполнения основных операций в БД М4W. В связи с увеличением числа пользователей нагрузка на ИС возросла, необходимо определить наиболее востребованные ресурсы;

- высокий уровень дефрагментации таблиц БД на уровне блоков, что приводит к увеличению времени выборки данных. Необходимо определить для каждой из таблиц типы выполняемых над ней операций и перенастроить ее параметры в соответствии с полученными данными;

- неэффективное использование компонентов SGA. Как известно [3], сервер Oracle включает в себя три основные структуры памяти: SGA (System Global Area) — глобальную область системы, представляющую собой большой совместно используемый сегмент памяти, к которому обращаются все процессы Oracle; PGA (Process Global Area) — глобальную область процесса, недоступную другим процессам/потокам, и UGA (User Global Area) — глобальную область пользователя, связанную с сеансом (хранит состояние сеанса). Каждый экземпляр Oracle имеет одну SGA. Область SGA разбита на несколько пулов: Java-пул, большой пул, разделяемый пул, неопределенный (Null) пул. Неправильное распределение памяти сервера по компонентам SGA приводит к неэффективной работе экземпляра БД. Для выявления оптимальных параметров SGA необходимо разработать средство, позволяющее сравнить эффективность работы каждой из конфигураций SGA;

- аналогично предыдущему пункту необходимо провести настройку параметров использования сегментов отката и табличных пространств отката;

- необходимо предоставить разработчикам, администраторам и группе сопровождения средства, позволяющие анализировать эффективность выполняемых в БД запросов.

Для выявления причин возникновения указанных проблем и поиска средств борьбы с ними необходимы следующие данные, определяемые на *четвертом* этапе:

- 1) данные об использовании процессора на уровне БД;
- 2) число операций чтения/записи на жестком диске (ЖД);
- 3) число и размер передаваемых сетевых сообщений в БД;
- 4) число пользователей, работающих с ИС;
- 5) число выполняемых транзакций;
- 6) данные о событиях ожидания, возникающих в БД (в СУБД Oracle более 250 событий ожидания ресурсов, ответа пользователя и др.; в БД активными, имеющими не нулевое значение, могут быть не все события, обычно не более 100);

7) определение наиболее ресурсоемких и часто используемых SQL-запросов и их анализ (общая продолжительность обработки, продолжительность обработки на ЦПУ, объем данных, считанных из памяти и ЖД, возникшие события ожидания);

8) представление профиля сессий (общая продолжительность работы сессии, время реальной работы, число транзакций, распределение времени обслуживания и времени ожидания, физический, логический и сетевой ввод-вывод, потребление ресурсов);

9) выделение объектов в БД (таблиц, процедур, индексов и др.), к которым наиболее часто происходит обращение;

10) подсчет коэффициентов работы SGA (библиотечного, буферно-го кэша, словаря данных и др.) и сегментов отката;

11) данные по сбору статистики по объектам (когда последний раз собиралась, как много изменений было проведено над объектом и др.).

Весь набор параметров можно разделить на несколько групп в зависимости от целей, которые преследуются при их сборе и анализе.

Например, в первую группу можно включить с первого по шестой параметры, определяющие степень загрузки системы во времени.

Ко второй группе можно отнести статистику, позволяющую составить профиль сессии, выделить наиболее ресурсоемкие и продолжительные сессии (параметры 7 и 8), провести анализ наиболее “тяжелых” SQL-операторов. Необходимость сбора именно этих параметров объясняется правилом 80/20. Его суть в следующем: 80 % проблем с производительностью связано с плохо спроектированными или плохо реализованными операторами SQL и только 20 % — с параметрами БД [4].

Параметры третьей группы (параметры 10 и 11) оценивают систему с точки зрения организации вычислений, структуры БД, эффективности работы экземпляра и анализа различных коэффициентов, выполняют настройку использования защелок, фоновых процессов Oracle и другие внутренние настройки БД. Параметры 10 и 11 отвечают только за 20 % производительности БД, но именно они являются средствами администратора [5].

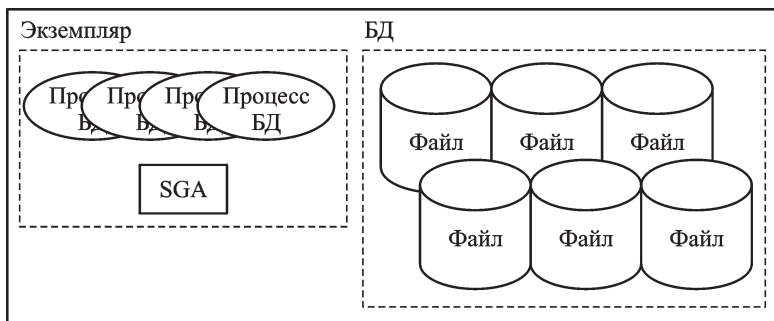


Рис. 2. Обобщенная структура сервера Oracle

К четвертой группе относятся параметры, определяющие наиболее востребованные ресурсы и объекты БД (параметры 9). Выявление их позволяет пересмотреть способы работы с ними, переконфигурировать используемое ПО, выявить ошибки в нем, правильно проводить “upgrade” и настройку сервера.

Выделив четыре группы параметров, можно определить четыре подхода к оценке работы системы, а именно: общий анализ системы, анализ SQL-операторов, выполняемых в системе, анализ работы экземпляра и анализ структуры БД. С экземпляром связаны файлы параметров. По этим файлам экземпляр при запуске определяет свои характеристики, например размер структур в памяти и местонахождение управляющих файлов.

На рис. 2 приведена упрощенная структура БД, работающей под управлением СУБД Oracle. Она включает в себя большую область памяти SGA, содержащую внутренние структуры данных, набор процессов, подключенных к этой области SGA. Серверные процессы выполняют запросы клиентов. Фоновые процессы выполняются при запуске экземпляра и решают различные задачи поддержки БД. Базу данных образуют файлы данных, паролей, журналы транзакций, управляющие и временные файлы. Единицами выделения пространства под объекты в БД Oracle являются: табличное пространство (ТП), сегмент, экстенд и блок. Блок — наименьшая единица выделения пространства в Oracle.

На **пятом** этапе необходимо определить способы доступа к выбранным параметрам. Например, в Oracle имеются, по меньшей мере, три различных способа получения данных [6]:

- SQL-запросы к фиксированным представлениям;
- прямой опрос сегментов разделяемой памяти;
- активирование в Oracle функции расширенной трассировки SQL, сохраняющей в трассировочном файле все хронометрические сведения о ходе выполнения команд для сессии Oracle.

SQL-запросы к V\$ представлениям позволяют получить данные о потреблении ресурсов, т.е. о числе обращений к различным ресур-

сам. Это относительно простой способ доступа к хронометрическим данным.

Прямой опрос сегментов разделяемой памяти дает ту же информацию, что и фиксированные представления, но позволяет вести выборку данных с высокой частотой, что весьма полезно при решении разнообразных проблем производительности. Однако доступ к разделяемой памяти и последующее хранение гигантских массивов данных требуют или больших трудозатрат, или расходов на специальный инструментарий.

Расширенная трассировка SQL также способна предоставить данные для надежной диагностики, не требуя при этом тех трудовых или финансовых ресурсов, которые характерны для способа с опросом памяти. Принципиальный недостаток расширенной трассировки в том, что она позволяет получить диагностические данные только для тех операций, о которых заранее известно, что они выполняются в неоптимальных условиях.

На *шестом* этапе осуществляется выбор инструментальных средств, используемых для разработки ПО. На выбор инструментальных средств влияют следующие факторы:

- работоспособность продукта с избранным сервером БД;
- наличие средств доступа к серверу БД;
- поддержка распределения вычислений между клиентом и сервером;
- взаимодействие с Windows и другими приложениями Windows;
- наличие интегрированного построителя отчетов, запросов и т.д.;
- возможность доступа к различным источникам данных в одном запросе;
- объектно-ориентированность инструментария, возможность создавать и применять новые объекты;
- степень развития визуальной среды проектирования;
- простота освоения языка программирования, средства отладки приложения;
- возможность переноса на другие платформы и среды;
- рейтинг фирмы-производителя, история продукта, сведения о его применении для аналогичных проектов.

Например, показанная на рис. 1 система сбора статистики обслуживает БД, работающие под управлением СУБД Oracle 9.2, функционирующие на серверах с ОС Windows 2003 Server или Windows XP. Для работы с системой необходимо сетевое подключение к серверу, на котором размещено ПО, и к серверу, на котором размещена анализируемая БД. Работа выполняется в веб-браузере Internet Explorer.

Основным достоинством веб-интерфейса является то, что для работы программы не требуется ее установка на компьютер-клиент, в



результате чего она может работать на любой платформе. Доступ к ПО возможен, даже если оно не установлено на компьютере. При обновлении ПО достаточно заменить его только на сервере.

Если оболочка и интерфейс системы будут созданы на основе технологии NET, то внутренний движок будет основан на языке PL/SQL, представляющем собой расширение языка SQL для работы с СУБД Oracle. Для обращения к объектам БД можно использовать технологию ADO.NET.

Для выполнения требуемых тестов в течение заданного интервала времени можно использовать пакет DBMS\_JOB [7].

И, наконец, на последнем, *седьмом* этапе разрабатывается система автоматизированного сбора и регистрируются значения выбранных параметров. Авторы настоящей работы принимали участие в проектировании системы сбора статистики “Аналитик” для одного из калужских предприятий, предназначенной для сбора указанных параметров БД, работающих под управлением СУБД Oracle 9.2i.

Система “Аналитик” включает в себя модуль управления, системную часть, утилиты и прикладную часть. Система “Аналитик” обеспечивает три режима работы: настройку, управление анализируемыми БД и собственно анализ БД. Описание системы и режимов ее функционирования в статье не приводится ввиду большого объема. Система позволяет получить следующие диагностические данные о работе анализируемой базы.

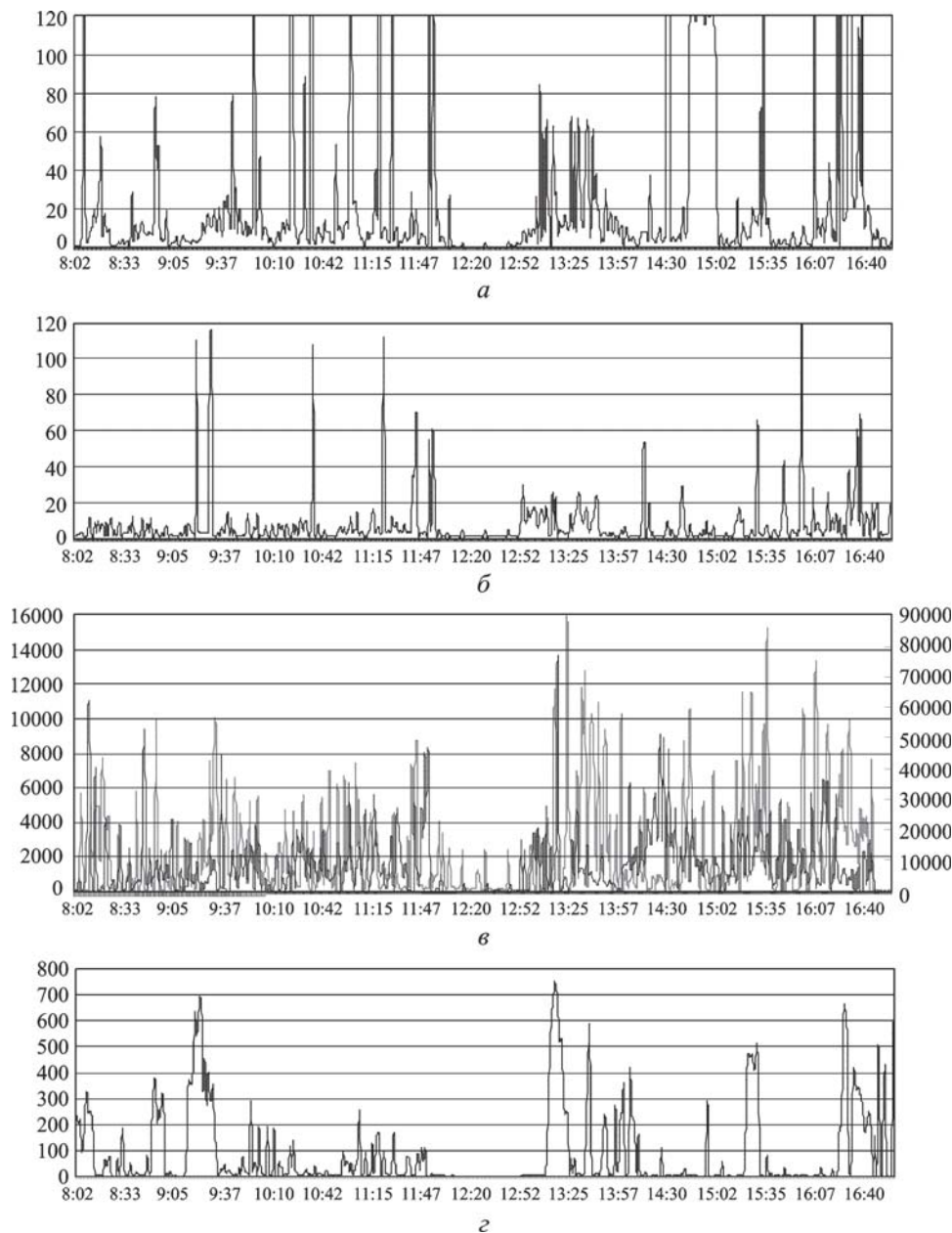
- Распределение нагрузки во времени на подсистему ввода-вывода ЖД; данные об использовании ЦП непосредственно БД; число подключенных к БД пользователей; число выполненных в БД транзакций; число байт, переданных от клиентского ПО в БД, число байт, переданных в обратном направлении от БД к клиенту; число байт, переданных удаленным базам и полученных от них;

- коэффициенты использования библиотечного пула, словарного и буферного кэша, ТП отката, сортировок, выполненных в БД. Эти коэффициенты можно получить на любом заранее заданном промежутке работы БД;

- события ожидания, распределенные по времени работы БД;
- профиль выбранной активной сессии и события ожидания, актуальные для данной сессии;
- профиль находящихся в библиотечном кэше SQL-операторов;
- трассировку сессии, просмотр файла трассировки и при необходимости его обработку утилитой tkprof;
- данные о дате сбора статистики по любому из объектов БД;
- данные о проценте использования индексов по всей системе;
- тип выполняемых операций над любой таблицей БД.

На рис. 3 показаны некоторые зависимости, полученные с помощью разработанной системы сбора статистики.

Построенные графики позволят определить время, когда система наиболее загружена, а также распределить выполнение пакетных заданий во времени (например, загрузку данных в БД, процедуры ре-



**Рис. 3. Диагностические данные о работе анализируемой БД:**

*a* — время работы ЦПУ, мс; *б* — число подключений к БД; *в* — число блоков данных, записанных (черные линии) на ЖД и считанных (серые линии) с ЖД; *г* — число выполненных транзакций

зервного копирования). Объем анализируемой базы около 20 Гбайт. Изменение конфигурации сервера должно быть серьезно мотивировано и подкреплено объективными данными о реальной нагрузке на сервер.

**Заключение.** Рассмотренные этапы позволяют выработать методики подготовительных работ, настройки производительности и других параметров БД, что существенно снижает трудоемкость этих процессов и облегчает работу администратора. Настройка БД устраняет необходимость приобретения дополнительного оборудования для поддержания требуемых параметров БД, снижает требования к конфигурации оборудования, уменьшает стоимость обслуживания всей системы. За счет настройки можно сократить время ответа, что повышает психологический комфорт пользователей и качество обслуживания клиентов.

В связи с появлением и распространением новых версий СУБД Oracle был проведен анализ возможности использования предложенной методики для оценки производительности в СУБД Oracle 10.2g. Для этого были рассмотрены изменения в структуре системных представлений в новых версиях СУБД.

Данные, предоставленные документацией на СУБД Oracle 10.2, показывают, что структура используемых системных представлений или осталась неизменной (например, представление V\$FILESTAT) или при добавлении новых полей сохранены ранее использовавшиеся поля для поддержки более старых версий. Например, в представление V\$SYSSTAT было добавлено новое поле STAT\_ID (идентификатор статистики). При этом сохранено ранее использовавшееся в аналогичных целях поле STATISTIC#, отмеченное в документации, как атрибут поддержки более ранних релизов (Note: Statistics numbers are not guaranteed to remain constant from one release to another [8]).

Анализ собранной статистики позволяет определить SQL-запросы, которые были выполнены наибольшее число раз; SQL-запросы, выполнение которых потребовало наибольшего числа физического или логического чтения в БД, и др. Кроме того, можно определить запросы, которые были выполнены не оптимально и требуют настройки. Анализ статистики за 25.03.10 показал, что следующие SQL-запросы были выполнены:

```
SELECT PODR_TYPE FROM M4ADMIN.PODRAZD WHERE PODR_ID
= :B1 — 26 207 901 раз;
SELECT PODR_PARENT FROM M4ADMIN.PODRAZD WHERE PODR_ID
= :B1 — 10 977 751 раз;
SELECT CHECKSUM FROM APPROVAL_Tech WHERE OB_ID = :B2
AND SIGN_TYPE = :B1 — 2 220 833 раз;
SELECT CHECK_SUM FROM CHK_SUM WHERE OB_ID = :B1 AND
TI_KIND = (SELECT TI_ID FROM TECHINFORM_KIND WHERE S_NAME
= :B2) — 2 132 798 раз.
```

Данные запросы выполняются за доли секунд, но частота их выполнения может привести к снижению производительности БД. Следовательно, необходимо перепроектировать ту часть программного обеспечения, в рамках которого проводится их выполнение, что потребует времени. Чтобы максимально снизить негативное влияние на БД этих типов запросов, необходимо провести анализ каждого из них и оптимизировать их выполнение путем настройки.

Далее приведен пример настройки первого запроса. План выполнения данного запроса следующий:

```
SELECT STATEMENT CHOOSE Cost: 2 Bytes: 7 Cardinality: 1  
TABLE ACCESS BY INDEX ROWID M4ADMIN.PODRAZD Cost: 2  
Bytes: 7 Cardinality: 1  
INDEX UNIQUE SCAN UNIQUE M4ADMIN.SYS_C003446 Cost: 1  
Cardinality: 1
```

Таблица PODRAZD содержит 157 записей и занимает 2 блока БД. Если бы оптимизатор БД выбрал в качестве метода доступа полный просмотр таблицы, то на это потребовалась бы одна операция чтения. Для индексного метода доступа потребуются три операции чтения: одна операция чтения для выборки корневого узла индекса SYS\_C003446, одна — для выбора листового узла индекса SYS\_C003446, одна — для доступа к таблице PODRAZD. Индекс SYS\_C003446 является первичным ключом таблицы, что не позволяет его удалить. Это позволяет утверждать, что оптимизатор БД выбирает неоптимальный план выполнения запроса. Изменив параметры статистики по данному индексу, например указав, что в индексе SYS\_C003446 содержатся 10 000 записей и из них только одна уникальная, можно сделать индекс мало пригодным для использования оптимизатором. Такая операция выполняется следующей командой:

```
DBMS_STATS.SET_INDEX_STATS  
(ownname=>'M4ADMIN',indname=>'SYS_C003446',numrows=>10000,  
numdist=>1);
```

В результате план выполнения данного SQL-запроса будет иметь следующий вид (и обеспечит оптимальный тип доступа):

```
SELECT STATEMENT CHOOSE Cost: 3 Bytes: 7 Cardinality: 1  
TABLE ACCESS FULL M4ADMIN.PODRAZD Cost: 3 Bytes: 7 Cardinality: 1
```

Аналогично проводится настройка и других запросов. Тестирование SQL-запросов, проведенное в новой версии СУБД, и анализ возвращаемых ими результатов позволяют отметить возможность использования предложенной методики.

Средства автоматического запуска и обработки результатов трассировки сессий, тестирование способов, используемых для выполнения

данной функции в новой версии СУБД, также показали свою работоспособность.

Технология организации хранения собранной статистической информации, а именно использование дополнительной системной базы, соединенной связью db\_link с анализируемыми БД, позволяет выполнять обращение как от старых версий к более новым версиям БД, так и в обратном направлении. Например, центральная база, работающая под управлением СУБД Oracle 9i (или 10.2g), может обслуживать БД версий 9 и 10.

Таким образом, разработанная методика подготовительных работ для настройки производительности на основе анализа сбора статистики применима в новых версиях СУБД Oracle.

## СПИСОК ЛИТЕРАТУРЫ

1. М и л л с а п К., Х о л ь т К. Oracle. Оптимизация производительности: Пер. с англ. – СПб.: Символ Плюс, 2006. – 464 с.
2. О п т и м и з а ц и я производительности Oracle8i: мифы и реальность / Гаджа Кришна Вайдияната // Oracle Magazine RE. – Май, 2002.
3. С е р в е р Oracle 9i. Справочное руководство по серверу.
4. Т о й Д. Настройка SQL. Для профессионалов. – СПб.: Питер, 2004. – 333 с.
5. О г а с л е 9 i DBA. Основы администрирования. – М.: НОУ “УКЦ ФОРС”, 2002.
6. А s k Т о м. [Электронный ресурс] – Режим доступа: <http://asktom.oracle.com/pls/asktom.html>, свободный.
7. К а й т Т. Oracle для профессионалов: Пер. с англ. – СПб.: ООО “ДиаСофтЮП”, 2005. – 656 с.
8. О г а с л е® Database Reference 10g Release 2 (10.2).

Статья поступила в редакцию 16.02.2010

Анастасия Борисовна Лачихина окончила Калужский филиал МГТУ им. Н.Э. Баумана в 1996 г. Старший преподаватель, аспирантка Калужского филиала МГТУ им. Н.Э. Баумана. Автор 15 научных работ в области информационных технологий.

A.V. Lachikhina graduated from the Kaluga Branch of the Bauman Moscow State Technical University in 1996. Senior teacher, post-graduate of the Kaluga Branch of the Bauman Moscow State Technical University. Author of 15 publications in the field of information technologies.



Анатолий Викторович Мазин родился в 1948 г., окончил Одесское высшее инженерное морское училище в 1972 г. Канд. техн. наук, доцент кафедры “Компьютерные системы и сети” Калужского филиала МГТУ им. Н.Э. Баумана. Автор 46 научных работ в области информационных технологий и технической диагностики.

A.V. Mazin (b. 1948) graduated from the Odessa Higher Engineering Marine School in 1972. Ph. D. (Eng.), assoc. professor of “Computer Systems and Networks” department of the Kaluga Branch of the Bauman Moscow State Technical University. Author of 46 publications in the field of information technologies and technical diagnostics.

