

Oracle Database 11g: администрирование, семинар I

Том I • Справочное руководство

D50102RU20

Выпуск 2.0

Август 2010 г.

D79468

ORACLE®

Авторы

Deirdre Matishak
Mark Fuller

Рецензенты и технические эксперты

Maria Billings
Herbert Bradbury
Yanti Chang
Timothy Chien
Andy Fotunak
Gerlinde Frenzen
Steve Friedberg
Joel Goodman
Vimala Jacob
Dominique Jeunot
Pete Jones
Fukue Kawabe
Donna Keesling
Sean Kim
Achiel Langers
Gwen Lazenby
Essi Parast
Randy Richeson
Joe Roch
Hilda Simon
Ira Singer
Jim Spiller
Supithran Thananayagam
Branislav Valny
Manju Varrier

Редакторы

Raj Kumar
Daniel Milne

Графический дизайнер

Rajiv Chandrabhanu

Издатели

Jobi Varghese
Veena Narasimhan

© Oracle и филиалы, 2009, 2010. Все права защищены.

Отказ от ответственности

Этот документ содержит частную информацию и защищен авторскими правами и другими законами об интеллектуальной собственности.

Вы можете копировать и печатать этот документ исключительно для ваших собственных потребностей в рамках курса обучения Oracle. Этот документ не может быть модифицирован или изменен каким бы то ни было способом. За исключением добросовестного использования в рамках законов об авторских правах, вы не можете использовать, предоставлять, загружать, копировать, печатать, отображать, выполнять, воспроизводить, публиковать, лицензировать, размещать, передавать и распространять этот документ полностью или частично без явного разрешения корпорации Oracle.

Информация, содержащаяся в данном документе, может быть изменена без уведомления. При обнаружении проблем в документе направляйте отчеты об этом в письменной форме по адресу: Oracle University, 500 Oracle Parkway, Redwood Shores, California 94065 USA. Отсутствие ошибок в этом документе не гарантируется.

Замечание об ограниченных правах

В случае передачи этой документации Правительству США или использования ее от имени Правительства США применяется следующее замечание:

ПРАВА ПРАВИТЕЛЬСТВА США

Права Правительства США на использование, модификацию, воспроизведение, выпуск, выполнение, отображение и разглашение этих обучающих материалов ограничены условиями действующего лицензионного соглашения Oracle и/или действующим контрактом с Правительством США.

Замечание о торговых марках

Oracle и Java являются зарегистрированными товарными знаками корпорации Oracle и/или ее филиалов. Другие названия могут быть торговыми марками соответствующих владельцев.

Содержание

I Введение

- Цели курса I-2
- Предлагаемый план I-3
- Продукты и службы Oracle I-4
- Oracle Database 11g: “g” – это сетка (grid) I-5
- Инфраструктура распределенных вычислений для одного экземпляра I-7

1. Архитектура базы данных Oracle

- Цели 1-2
- База данных Oracle 1-3
- Подключение к серверу 1-4
- Архитектура сервера баз данных Oracle: обзор 1-6
- Экземпляр: конфигурации баз данных 1-7
- Подключение к экземпляру базы данных 1-8
- Структуры памяти базы данных Oracle 1-9
- Разделяемый пул 1-11
- Буферный кэш базы данных 1-13
- Буфер журнала повторов 1-14
- Большой пул 1-15 Пул Java и пул потоков 1-16
- Глобальная программная область (PGA) 1-17
- Экспресс-тест 1-18
- Архитектура процессов 1-20
- Структуры процессов 1-21
- Процесс записи в базу данных (DBWn) 1-23
- Процесс записи в журнал (LGWR) 1-25
- Процесс создания контрольной точки (CKPT) 1-27
- Процесс системного монитора (SMON) 1-28
- Процесс монитора процессов (PMON) 1-29
- Процесс восстановления 1-30
- Процессы архивации (ARCn) 1-31
- Последовательность запуска процессов 1-32
- Архитектура хранения БД 1-33
- Логические и физические структуры БД 1-35

Сегменты, экстенды и блоки 1-37
Табличные пространства и файлы данных 1-38
Табличные пространства SYSTEM и SYSAUX 1-39
Автоматическое управление хранением 1-40
Компоненты хранения ASM 1-41
Взаимодействие с базой данных Oracle: память, процессы и хранение 1-42
Экспресс-тест 1-44
Заключение 1-46
Упражнение 1: обзор 1-47

2 Установка программного обеспечения Oracle

Цели 2-2
Задачи администратора БД Oracle 2-3
Средства администрирования базы данных Oracle 2-4
Планирование установки 2-6
Установка инфраструктуры Oracle Grid Infrastructure и базы данных Oracle:
системные требования 2-8
Подготовка операционной системы 2-9
Настройка переменных среды 2-10
Проверка требований к системе 2-11
Oracle Universal Installer (OUI) 2-12
Пример: сценарий установки 2-13
Часть 1. Установка инфраструктуры распределенных вычислений Oracle Grid
Infrastructure для автономного сервера 2-14
Выбор языков продукта 2-15
Создание группы дисков ASM 2-16
Установка паролей ASM 2-17
Определение привилегированных групп операционной системы 2-18
Указание расположения установки 2-19
Создание каталога Inventory 2-20
Выполнение предварительных проверок 2-21
Проверка сводных данных установки 2-22
Отслеживание хода установки 2-23
Выполнение сценариев конфигурации root 2-24
Запуск помощников Configuration Assistant 2-25
Завершение установки 2-26
Настройка группы дисков FRA 2-27
Экспресс-тест 2-28
Часть 2. Установка программного обеспечения базы данных Oracle 2-30
Выбор типа установки 2-31
Выбор вариантов установки Grid 2-32
Выбор параметров языка 2-33

Выбор выпуска базы данных 2-34
Указание расположения установки 2-35
Выбор групп операционной системы 2-36
Выполнение предварительных проверок 2-37
Страница сводки установки 2-38
Страница установки продукта 2-39
Установка готова 2-40
Параметр установки: бесшумный режим 2-41
Экспресс-тест 2-42
Заключение 2-44
Обзор упражнения 2: подготовка среды базы данных 2-45

3 Создание базы данных Oracle с помощью DBCA

Цели 3-2
Проектирование базы данных 3-3
Базы данных: примеры 3-4
Выбор соответствующего набора символов 3-5
Как используются наборы символов? 3-7
Возможные проблемы 3-8
Database Configuration Assistant (DBCA) 3-9
Создание базы данных с помощью DBCA 3-10
Сводка создания базы данных 3-16
Управление паролями 3-17
Создание шаблона структуры базы данных 3-18
Удаление базы данных с помощью DBCA 3-19
Выполнение дополнительных задач с помощью DBCA 3-21
Экспресс-тест 3-22
Заключение 3-24
Обзор упражнения 3: использование DBCA 3-25

4 Управление экземпляром базы данных

Цели 4-2
Структура управления 4-3
Запуск и остановка средства управления базой данных Database Control 4-4
Oracle Enterprise Manager 4-5
Домашняя страница базы данных 4-7
Другие инструменты Oracle 4-8
Использование SQL*Plus 4-9
Вызов SQL*Plus из сценария оболочки 4-10
Вызов SQL-сценария из SQL*Plus 4-11
Файлы параметров инициализации 4-12

Упрощенные параметры инициализации 4-14
Параметры инициализации: примеры 4-15
Использование SQL*Plus для отображения параметров 4-19
Изменение значений параметров инициализации 4-21
Изменение значений параметров: примеры 4-23
Экспресс-тест 4-24
Запуск и остановка базы данных: учетные данные 4-26
Запуск экземпляра базы данных Oracle 4-27
Запуск экземпляра базы данных Oracle: NOMOUNT 4-28
Запуск экземпляра базы данных Oracle: MOUNT 4-29
Запуск экземпляра базы данных Oracle: OPEN 4-30
Параметры запуска: примеры 4-31
Остановка экземпляра базы данных Oracle 4-32
Режимы остановки 4-33
Параметры остановки 4-34
Параметры остановки: примеры 4-37
Просмотр журнала предупреждений 4-38
Использование файлов трассировки 4-40
Динамические представления производительности 4-42
Динамические представления производительности: примеры использования 4-43
Динамические представления производительности: рекомендации 4-44
Словарь данных: обзор 4-45
Представления словаря данных 4-46
Словарь данных: примеры использования 4-48
Экспресс-тест 4-49
Заключение 4-51
Обзор упражнения 4: управление экземпляром Oracle 4-52

5 Управление экземпляром ASM

Цели 5-2
Преимущества ASM для администраторов 5-3
Экземпляр ASM 5-4
Компоненты ASM: экземпляр ASM — основные процессы 5-6
Параметры инициализации экземпляра ASM 5-7
Взаимодействие между экземплярами базы данных и ASM 5-9
Экземпляр ASM: динамические представления производительности 5-10
Системные полномочия ASM 5-11
Использование Enterprise Manager для управления пользователями ASM 5-12
Запуск и остановка экземпляров ASM с помощью SQL*Plus 5-13
Запуск и остановка экземпляров ASM с помощью srvctl 5-15

Запуск и остановка экземпляров ASM с помощью asmcmd	5-16
Обзор дисковых групп	5-17
Диски ASM	5-18
Выделенные блоки	5-19
Файлы ASM	5-20
Карты экстенгов	5-21
Детализация размещения	5-22
Детализированное разнесение	5-23
Группы отказа ASM	5-25
Пример разнесения и зеркалирования	5-26
Пример отказа	5-27
Управление дисковыми группами	5-28
Создание и удаление дисковых групп с помощью SQL*Plus	5-29
Добавление дисков в дисковые группы	5-30
Прочие команды ALTER	5-31
Управление ASM с помощью Enterprise Manager	5-32
Совместимость дисковой группы ASM	5-33
Атрибуты дисковой группы ASM	5-35
Использование диспетчера Enterprise Manager для редактирования атрибутов дисковой группы	5-36
Извлечение метаданных ASM	5-37
Обзор быстрой зеркальной повторной синхронизации ASM	5-38
Экспресс-тест	5-39
Заключение	5-41
Обзор упражнения 5: управление экземпляром ASM	5-42

6 Настройка конфигурации сетевой среды Oracle

Цели	6-2
Oracle Net Services	6-3
Прослушиватель Oracle Net	6-4
Установление сетевых соединений	6-5
Установление соединения	6-6
Сеансы пользователей	6-7
Средства настройки и управления сетью Oracle	6-8
Служебная программа управления прослушивателями	6-10
Синтаксис служебной программы управления прослушивателями	6-11
Использование SRVCTL для запуска и остановки прослушивателя	6-13
Домашняя страница прослушивателя	6-14
Страница администрирования Net Services	6-15
Создание прослушивателя	6-16
Добавление адресов прослушивателей	6-17
Регистрация служб базы данных	6-18

- Методы разрешения имен 6-20
- Easy Connect 6-21
- Локальное разрешение имен 6-22
- Разрешение имен на сервере каталогов 6-23
- Внешнее разрешение имен 6-24
- Настройка псевдонимов служб 6-25
- Дополнительные параметры соединения 6-26
- Проверка связности сети Oracle Net 6-28
- Сеансы пользователей: выделенный серверный процесс 6-29
- Сеансы пользователей: разделяемые серверные процессы 6-30
- SGA и PGA 6-31
- Разделяемый сервер: пул соединений 6-32
- Когда не стоит использовать разделяемый сервер 6-33
- Настройка соединения между базами данных 6-34
- Соединение с другой БД 6-35
- Экспресс-тест 6-36
- Заключение 6-38
- Обзор упражнения 6: работа с сетевыми компонентами Oracle 6-39

7 Управление структурами хранения базы данных

- Цели 7-2
- Хранение табличных данных 7-3
- Блок базы данных: содержимое 7-4
- Структура хранилища 7-5
- Создание нового табличного пространства 7-6
- Хранилище для табличных пространств 7-8
- Табличные пространства в предварительно настроенной базе данных 7-10
- Изменение табличного пространства 7-12
- Действия над табличными пространствами 7-14
- Удаление табличных пространств 7-16
- Просмотр сведений о табличном пространстве 7-17
- Просмотр содержимого табличного пространства 7-18
- Файлы, управляемые Oracle (OMF) 7-19
- Увеличение базы данных 7-21
- Экспресс-тест 7-22
- Заключение 7-24
- Обзор упражнения 7: управление структурами хранения базы данных 7-25

8 Администрирование защиты пользователей

- Цели 8-2
- Формуляры пользователей БД 8-3

Предопределенные формуляры администратора 8-5
Создание пользователя 8-6
Аутентификация пользователей 8-7
Аутентификация администраторов 8-9
Разблокирование формуляра пользователя и сброс пароля 8-10
Полномочия 8-11
Системные полномочия 8-12
Объектные полномочия 8-14
Аннулирование системных полномочий с параметром ADMIN OPTION 8-15
Аннулирование объектных полномочий с параметром GRANT OPTION 8-16
Преимущества использования ролей 8-17
Назначение полномочий ролям и назначение ролей пользователям 8-18
Предопределенные роли 8-19
Создание роли 8-20
Защита ролей 8-21
Назначение ролей пользователям 8-22
Экспресс-тест 8-23
Профили и пользователи 8-25
Возможности защиты паролем 8-27
Создание профиля с паролем 8-29
Стандартная функция проверки пароля: VERIFY_FUNCTION_11G 8-30
Назначение квот пользователям 8-31
Применение принципа меньших полномочий 8-33
Защита формуляров с большими полномочиями 8-35
Экспресс-тест 8-36
Заключение 8-38
Обзор упражнения 8: администрирование пользователей 8-39

9 Управление параллельным доступом к данным

Цели 9-2
Блокировки 9-3
Механизм блокирования 9-4
Параллелизм при работе с данными 9-5
DML-блокировки 9-7
Механизм обслуживания очереди 9-8
Конфликты блокировок 9-9
Возможные причины конфликтов блокировок 9-10
Выявление конфликтов блокировок 9-11
Устранение конфликтов блокировок 9-12
Устранение конфликтов блокировок с помощью SQL 9-13

Взаимные блокировки 9-14
Экспресс-тест 9-15
Заключение 9-17
Обзор упражнения 9: управление данными и параллельным доступом 9-18

10 Управление данными для отмены

Цели 10-2
Данные отмены операций 10-3
Транзакции и данные отмены операций 10-5
Хранение данных отмены операций 10-6
Данные отмены операций и данные повторов 10-7
Управление данными отмены операций 10-8
Настройка времени хранения данных отмены операций 10-9
Гарантированное хранение данных отмены операций 10-11
Приведение табличного пространства отмены операций к фиксированному размеру 10-12
Общие сведения о данных отмены операций 10-13
Использование помощника Undo Advisor 10-14
Просмотр системной активности 10-15
Экспресс-тест 10-16
Заключение 10-18
Обзор упражнения 10: управление сегментами отмены операций 10-19

11 Внедрение аудита базы данных Oracle

Цели 11-2
Разделение ответственности 11-3
Безопасность базы данных 11-4
Мониторинг совместимости 11-6
Стандартный аудит БД 11-7
Настройка журнала аудита 11-8
Унифицированные журналы аудита 11-9
Определение параметров аудита 11-10
Аудит по умолчанию 11-11
Страница аудита в Enterprise Manager 11-12
Использование и обслуживание информации аудита 11-13
Аудит на основе значений 11-14
Детализированный аудит 11-16
Политика FGA 11-17
Аудит DML-операторов: вопросы 11-19
Руководящие принципы FGA 11-20
Аудит SYSDBA 11-21
Обслуживание журнала аудита 11-22

Oracle Audit Vault	11-23
Экспресс-тест	11-24
Заключение	11-26
Обзор упражнения 11: внедрение защиты базы данных Oracle	11-27

12. Обслуживание базы данных

Цели	12-2
Обслуживание базы данных	12-3
Просмотр хронологии предупреждений	12-4
Терминология	12-5
Оптимизатор Oracle: обзор	12-6
Статистика оптимизатора	12-7
Использование страницы управления статистикой оптимизатора	12-8
Ручной сбор статистики оптимизатора	12-9
Предпочтения для выполнения сбора статистики	12-11
Автоматический репозиторий рабочей нагрузки (AWR)	12-13
Инфраструктура AWR	12-14
Базовые конфигурации AWR	12-15
Диспетчер Enterprise Manager и репозиторий AWR	12-16
Управление репозиторием AWR	12-17
Уровни статистики	12-18
Монитор автоматической диагностики баз данных (ADDM)	12-19
Выводы монитора ADDM	12-20
Рекомендации монитора ADDM	12-21
Структура консультационной системы	12-22
Диспетчер Enterprise Manager и помощники	12-24
Пакет DBMS_ADVISOR	12-25
Экспресс-тест	12-26
Автоматические задачи обслуживания	12-27
Конфигурация автоматических задач обслуживания	12-29
Предупреждения сервера	12-30
Установка пороговых значений	12-31
Создание и тестирование предупреждения	12-32
Уведомления о предупреждениях	12-33
Реагирование на предупреждения	12-35
Типы предупреждений и их очистка	12-36
Экспресс-тест	12-37
Заключение	12-38
Обзор упражнения 12: упреждающее обслуживание	12-39

13. Управление производительностью

Цели 13-2

Контроль производительности 13-3

Страница производительности Enterprise Manager 13-4

Детализация до определенной категории событий ожидания 13-5

Страница производительности: производительность 13-6

Контроль производительности: основные сеансы 13-7

Контроль производительности: основные службы 13-8

Управление компонентами памяти 13-9

Включение автоматического управления памятью (AMM) 13-10

Включение автоматического управления разделяемой памятью (ASMM) 13-11

Помощник по автоматическому выделению разделяемой памяти 13-12

Динамическая статистика производительности 13-13

Представления для устранения неполадок и настройки 13-15

Недопустимые и неиспользуемые объекты 13-16

Экспресс-тест 13-18

Заключение 13-20

14. Основные понятия резервного копирования и восстановления

Цели 14-2

Задачи администратора 14-3

Категории сбоев 14-5

Сбои операторов 14-6

Сбои пользовательских процессов 14-7

Сбои сети 14-8

Ошибка пользователя 14-9

Технология Flashback 14-10

Сбои экземпляра 14-12

Процедура восстановления экземпляра: процесс контрольной точки (СКРТ) 14-13

Процедура восстановления экземпляра: файлы журналов повторов и функция записи в журнал 14-14

Процедура восстановления экземпляра 14-15

Этапы восстановления экземпляра 14-16

Настройка восстановления экземпляра 14-17

Использование помощника по MTTR 14-18

Сбои носителей 14-19

Настройка возможности восстановления 14-20

Настройка области быстрого восстановления 14-21

Мультиплексирование управляющих файлов 14-22

Файлы журнала повторов 14-24

Мультиплексирование журнала повторов 14-25

Архивные файлы журналов 14-26
Процесс архиватора (ARCn) 14-27
Архивный файл журнала: именование и места назначения 14-28
Включение режима ARCHIVELOG 14-29
Экспресс-тест 14-30
Заключение 14-32
Обзор упражнения 14: настройка возможности восстановления 14-33

15. Выполнение резервного копирования базы данных

Цели 15-2
Решения для резервного копирования: обзор 15-3
Служебная программа Oracle Secure Backup 15-4
Управляемое пользователем резервное копирование 15-5
Терминология 15-6
Диспетчер восстановления (RMAN) 15-8
Настройка параметров резервного копирования 15-9
Планирование резервного копирования: стратегия 15-11
Планирование резервного копирования: параметры 15-12
Планирование резервного копирования: настройки 15-13
Планирование резервного копирования: график 15-14
Планирование резервного копирования: анализ 15-15
Резервное копирование управляющего файла в файл трассировки 15-16
Управление резервными копиями 15-17
Просмотр отчетов о резервном копировании 15-18
Контроль области быстрого восстановления 15-19
Использование командной строки RMAN 15-20
Экспресс-тест 15-21
Заключение 15-22
Обзор упражнения 15: создание резервных копий базы данных 15-23

16. Выполнение восстановления базы данных

Цели 16-2
Открытие базы данных 16-3
Поддержание базы данных в открытом состоянии 16-5
Помощник по восстановлению данных Data Recovery Advisor 16-6
Потеря управляющего файла 16-8
Потеря файла журнала повторов 16-9
Потеря файла данных в режиме NOARCHIVELOG 16-11
Потеря некритического файла данных в режиме ARCHIVELOG 16-12

Потеря критического для системы файла данных в режиме ARCHIVELOG 16-13
Сбои данных: примеры 16-14
Помощник по восстановлению данных Data Recovery Advisor 16-15
Оценка сбоев данных 16-16
Сбои данных 16-17
Получение перечня сбоев данных 16-18
Рекомендации по восстановлению 16-19
Устранение сбоев 16-20
Представления помощника по восстановлению данных Data Recovery Advisor 16-21
Экспресс-тест 16-22
Заключение 16-24
Обзор упражнения 16: выполнение восстановления базы данных 16-25

17. Перемещение данных

Цели 17-2
Перемещение данных: общая архитектура 17-3
Насос данных Oracle: обзор 17-4
Насос данных Oracle: преимущества 17-5
Объекты каталога для насоса данных 17-7
Создание объектов каталога 17-8
Клиенты Data Pump Export и Data Pump Import: обзор 17-9
Служебная программа Data Pump: интерфейсы и режимы 17-10
Экспорт Data Pump Export с помощью средства управления Database Control 17-11
Пример экспорта с помощью средства Data Pump Export: основные параметры 17-12
Пример экспорта с помощью средства Data Pump Export:
расширенные параметры 17-13
Пример экспорта с помощью средства Data Pump Export: файлы 17-14
Пример экспорта с помощью средства Data Pump Export: график 17-16
Пример экспорта с помощью средства Data Pump Export: анализ 17-17
Пример импорта с помощью средства Data Pump Import: impdp 17-18
Data Pump Import: преобразования 17-19
Использование диспетчера Enterprise Manager для мониторинга заданий насоса
данных 17-20
Миграция в режиме совместимости насоса данных с прежними системами 17-21
Режим совместимости насоса данных с прежними системами 17-22
Управление расположениями файлов 17-24
Загрузчик SQL*Loader: обзор 17-25
Загрузка данных с помощью загрузчика SQL*Loader 17-27
Управляющий файл загрузчика SQL*Loader 17-28
Способы загрузки 17-30
Внешние таблицы 17-31
Преимущества внешних таблиц 17-32
Определение внешних таблиц с помощью ORACLE_LOADER 17-33

Заполнение внешних таблиц с помощью ORACLE_DATAPUMP 17-34
Использование внешних таблиц 17-35
Словарь данных 17-36
Экспресс-тест 17-37
Заключение 17-39
Обзор упражнения 17: перемещение данных 17-40

18. Работа со службой поддержки

Цели 18-2
Использование рабочего интерфейса поддержки 18-3
Просмотр предупреждений о критических ошибках в диспетчере Enterprise Manager 18-4
Просмотр сведений о проблеме 18-5
Просмотр сведений об инциденте: файлы дампов 18-6
Просмотр сведений об инциденте: результаты проверки 18-7
Создание запроса на обслуживание 18-8
Упаковка и загрузка диагностических данных в центр поддержки Oracle 18-9
Отслеживание запросов на обслуживание и устранение неполадок 18-10
Закрытие инцидентов и проблем 18-12
Конфигурация пакета инцидента 18-13
Рабочий интерфейс поддержки Enterprise Manager для ASM 18-14
Работа со службой поддержки Oracle 18-15
Интеграция службы поддержки My Oracle Support 18-16
Использование службы поддержки My Oracle Support 18-17
Исследование проблемы 18-19
Регистрация запросов на обслуживание 18-21
Управление исправлениями 18-23
Применение выпуска исправлений 18-24
Использование помощника по исправлениям 18-25
Использование мастера исправлений 18-26
Применение исправления 18-27
Подготовка исправления к установке 18-28
Интерактивные исправления: обзор 18-29
Установка интерактивного исправления 18-30
Преимущества интерактивных исправлений 18-31
Традиционные исправления и интерактивные исправления 18-32
Интерактивные исправления: вопросы 18-33
Экспресс-тест 18-35
Заключение 18-36
Обзор упражнения 18: использование средств диспетчера EM для предупреждений и исправлений 18-37

Приложение А. Упражнения и решения

Приложение В. Основные команды Linux и vi

Приложение С. Синтаксис операторов SQL

Приложение D. Фоновые процессы Oracle

Приложение Е. Сокращения и термины

Приложение F. Oracle Restart

- Цели F-2
- Oracle Restart F-3
- Запуск процесса Oracle Restart F-5
- Управление Oracle Restart F-6
- Выбор правильной служебной программы SRVCTL F-8
- Конфигурация Oracle Restart F-9
- Использование служебной программы SRVCTL F-10
- Получение справки по служебной программе SRVCTL F-11
- Запуск компонентов с помощью служебной программы SRVCTL F-12
- Остановка компонентов с помощью служебной программы SRVCTL F-13
- Просмотр статуса компонентов F-14
- Просмотр конфигурации Oracle Restart для компонента F-15
- Добавление компонентов в конфигурацию Oracle Restart вручную F-16
- Экспресс-тест F-17
- Заключение F-18

Приложение G. Продолжение обучения и дополнительные материалы

- Что дальше? G-2
- Ресурсы для продолжения обучения G-3
- Oracle University G-4
- Продолжение обучения G-5
- Специальные области баз данных G-6
- Oracle Real Application Clusters G-7
- Oracle Data Guard G-8
- Обзор потоков G-9
- Oracle Technology Network G-11
- Защита G-12
- Примеры Oracle by Example G-13
- Oracle Magazine G-14

Сообщество Oracle Applications G-15
Техническая поддержка: My Oracle Support G-16
Страница продукта Oracle Database G-17
Спасибо! G-18

I

Введение

ORACLE

© Oracle, 2009. Все права защищены.

Цели

Изучив материал этого курса, вы освоите следующие темы:

- установка, создание и администрирование базы данных Oracle Database 11g версия 2;
- **настройка базы данных для приложений;**
- использование основных процедур мониторинга;
- применение стратегии резервного копирования и восстановления;
- перемещение данных между базами данных и файлами.

ORACLE

© Oracle, 2009. Все права защищены.

Цели

В данном курсе вы узнаете, как установить программное обеспечение Oracle Database 11g Enterprise Edition версии 2 и создать новую базу данных, а также научитесь администрировать базы данных.

Кроме того, вы научитесь настраивать базу данных для поддержки приложения, создавать пользователей, определять структуры хранения и настраивать параметры безопасности.

В курсе используется симуляция приложения. В то же время вы сможете осуществить все основные операции, необходимые для настоящего приложения.

Администрирование базы данных не завершается после ее настройки. Вы также узнаете, **как защитить ее при помощи разработанной стратегии резервного копирования и восстановления**, а также как осуществлять мониторинг базы данных для обеспечения ее эффективной работы.

Предлагаемый план

День	Уроки	День	Уроки
1	1. Архитектура базы данных Oracle 2. Подготовка среды базы данных 3. Создание базы данных Oracle 4. Управление экземплярами баз данных	3	9. Управление параллелизмом при работе с данными 10. Управление данными отмены операций 11. Внедрение аудита базы данных Oracle
		4	12. Обслуживание базы данных 13. Управление производительностью 14. Принципы резервного копирования и восстановления
2	5. Управление экземплярами ASM 6. Настройка сетевой среды Oracle 7. Управление структурами хранения базы данных 8. Администрирование защиты пользователей	5	15. Создание резервных копий базы данных 16. Восстановление базы данных 17. Перемещение данных 18. Работа со службой поддержки

ORACLE

© Oracle, 2009. Все права защищены.

Продукты и службы Oracle

- **Oracle Database**
- Oracle WebLogic Application Server
- Oracle Applications
- Oracle Collaboration Suite
- Oracle Developer Suite
- Oracle Services



ORACLE®

ORACLE®

I - 4

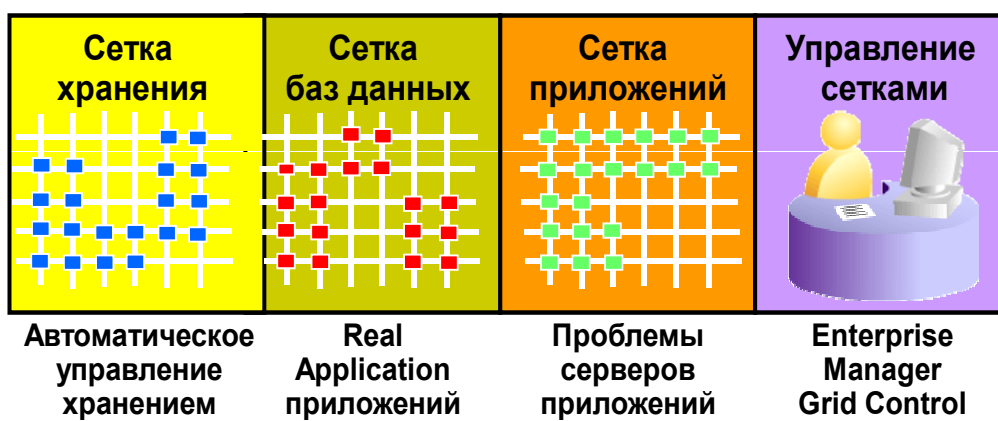
© Oracle, 2009. Все права защищены.

Продукты и службы Oracle

- **База данных Oracle** – это первая база данных, разработанная для корпоративных распределенных вычислений как наиболее гибкий и экономически выгодный способ управления информацией и приложениями.
- **Платформа Oracle Java 2, Enterprise Edition** – это сертифицированный сервер, объединяющий все необходимые средства для разработки и развертывания Web-приложений. Сервер приложений позволяет развертывать порталы для электронной коммерции, Web-службы и такие транзакционные приложения, как PL/SQL, Oracle Forms и приложения на основе Java EE.
- **Приложения Oracle:** Oracle E-Business Suite – это полный набор бизнес-приложений для автоматизации процессов в организации и управления ими.
- **Oracle Collaboration Suite:** Oracle Collaboration Suite – это единая интегрированная система управления всеми коммуникационными данными в организации: голосовыми данными, электронной почтой, факсимильными данными, беспроводной связью, календарной информацией и файлами.
- **Oracle Developer Suite:** Oracle Developer Suite – это полнофункциональная интегрированная среда, объединяющая средства разработки приложений и бизнес-аналитики.
- **Службы Oracle**, такие как Oracle Consulting и Oracle University, позволяют вам получить необходимый профессиональный опыт для проектов Oracle. Ссылки на множество полезных ресурсов находятся в приложении «Следующие шаги: продолжение обучения».

Oracle Database 11g: “g” - это сетка (grid)

- Open Grid Forum (OGF)
- Инфраструктура распределенных вычислений Oracle:
 - низкая стоимость;
 - **высокое качество услуг;**
 - простота управления.



ORACLE

I - 5

© Oracle, 2009. Все права защищены.

Oracle Database 11g: “g” – это сетка (grid)

Open Grid Forum (OGF) – это орган стандартизации, который разрабатывает стандарты распределенных вычислений. Он состоит из нескольких комитетов и рабочих групп, отвечающих за различные аспекты распределенных вычислений. В эти комитеты и рабочие группы входят представители научной и образовательной сферы, а также представители возрастающего числа коммерческих компаний. Веб-сайт OGF находится по адресу <http://www.ogf.org>.

Корпорацией Oracle создано программное обеспечение с инфраструктурой распределенных вычислений, выравнивающее нагрузку между серверами и позволяющее управлять этими серверами как единой системой. Использование распределенных вычислений позволяет достичь того же высокого уровня надежности, что и при использовании больших ЭВМ, благодаря кластеризации всех компонентов. Однако, в отличие от больших ЭВМ и UNIX-серверов с симметричной многопроцессорной обработкой (SMP), систему распределенных вычислений можно создать на основе технологий открытых систем, например на процессорах Intel и ОС Linux, что значительно сократит затраты.

Технология распределенных вычислений Oracle включает в себя следующие компоненты:

- автоматическое управление хранением (ASM)
- Real Application Clusters (RAC)
- кластеры сервера приложений
- Enterprise Manager Grid Control

Oracle Database 11g: “g” – это сетка (grid) (продолжение)

Технология автоматического управления хранением (ASM) позволяет распределить данные БД между всеми дисками, создать и обслуживать распределенное хранилище, а также обеспечивает высокую производительность операций ввода-вывода при минимальных затратах на обслуживание. При добавлении или удалении дисков ASM автоматически перераспределяет данные. (Для управления файловой системой не нужен диспетчер логических томов). Доступность данных можно повысить с помощью дополнительного зеркального дублирования дисков, кроме того, добавлять и удалять диски можно в оперативном режиме. См. урок «Управление структурами хранения базы данных». Технология Oracle **Real Application Clusters** позволяет запускать и масштабировать загрузку любых приложений на кластере из серверов, а также обладает следующими возможностями:

- **Интегрированное кластерное программное обеспечение:** включает в себя средства для подключения, обмена сообщениями, блокирования, управления и восстановления кластеров. Данная функция доступна на любых платформах с поддержкой Oracle Database 10g.
- **Автоматическое управление рабочей нагрузкой:** возможно определение правил автоматического распределения вычислительных ресурсов для каждой из служб как при обычном режиме работы, так и на случай сбоя. Эти правила могут динамически изменяться в зависимости от потребностей бизнеса. Динамическое распределение ресурсов в сетке баз данных является уникальным свойством технологии Oracle RAC.
- **Автоматическое уведомление о событиях в среднем ярусе:** в случае изменения конфигурации кластера промежуточные узлы могут мгновенно адаптироваться к сбою экземпляра или появлению нового экземпляра. Это позволяет конечным пользователям даже в случае сбоя экземпляра продолжить работу без задержек, обычно вызываемых перебоями в работе сети. В случае появления нового экземпляра промежуточные узлы могут немедленно начать выравнивать нагрузку подключений к этому экземпляру. Драйверы Oracle Database 10g Java Database Connectivity (JDBC) содержат функцию «Failover с быстрым соединением», которая может включаться автоматически для управления подобными ситуациями.

Oracle WebLogic Application Grid работает с любым сервером приложений, в т.ч. Oracle **WebLogic Server**, IBM **WebSphere Application Server** и **JBoss Application Server**, а также в исключительно сетевой среде без сервера приложений. Oracle WebLogic Application Grid позволяет масштабировать приложения и обеспечивает максимальную производительность в соответствии с ожиданиями. Возможность предоставления необходимых ресурсов по требованию позволяет Oracle WebLogic Application Grid линейно масштабировать промежуточную инфраструктуру с нескольких серверов до нескольких тысяч. Решение для распределенных вычислений в памяти облегчает доступ к часто используемым данным. За счет **сетки вычисления могут выполняться параллельно, что повышает производительность** приложения.

Компонент **Enterprise Manager Grid Control** управляет операциями на уровне сеток, включая управление всем стеком программного обеспечения, обеспечение пользователей, клонирование баз данных и управление исправлениями. Он позволяет следить за производительностью всех приложений с точки зрения конечных пользователей. Компонент Grid Control отслеживает производительность и доступность распределенной инфраструктуры как единой системы, а не отдельных запоминающих устройств, баз данных и серверов приложений. Все узлы оборудования, базы данных и серверы приложений можно сгруппировать в единые логические сущности и управлять группой адресатов как единым целым.

Примечание. В данном курсе Enterprise Manager Database Console используется для управления одной базой данных одновременно.

Инфраструктура распределенных вычислений для одного экземпляра

Функция инфраструктуры распределенных вычислений для одного экземпляра представлена в базе данных Oracle Database 11g (версия 2).

- Она устанавливается с носителя clusterware отдельно от программного обеспечения базы данных Oracle
- Включает в себя функцию автоматического управления хранением (ASM)
- Содержит решение для повышения доступности Oracle Restart для некластеризованных баз данных
 - позволяет отслеживать и перезапускать следующие компоненты:
 - экземпляры баз данных;
 - приемник Oracle Net;
 - службы баз данных;
 - экземпляр автоматического управления хранением (ASM);
 - группы дисков ASM;
 - службы уведомлений Oracle (ONS/eONS) для Data Guard.

ORACLE

© Oracle, 2009. Все права защищены.

Инфраструктура распределенных вычислений для одного экземпляра

Функция инфраструктуры распределенных вычислений для одного экземпляра представлена в базе данных Oracle Database 11g (версия 2). Она устанавливается с носителя clusterware отдельно от программного обеспечения базы данных Oracle и включает в себя функцию автоматического управления хранением Oracle Automatic Storage Management, а также новую функцию Oracle Restart.

Oracle Restart позволяет повысить доступность базы данных Oracle. Этот компонент реализует решение для высокой доступности в некластеризованных средах (средах с одним экземпляром). В среде Oracle Real Application Cluster (Oracle RAC) функция автоматического перезапуска компонентов обеспечивается Oracle Clusterware. Oracle Restart позволяет отслеживать работоспособность следующих компонентов, а также автоматически перезапускать их:

- экземпляры баз данных;
- приемник Oracle Net;
- службы баз данных;
- экземпляр ASM
- группы дисков ASM;
- службы уведомлений Oracle (ONS/eONS) для Data Guard.

Oracle Restart обеспечивает запуск компонентов в порядке, соответствующем зависимостям компонентов. При необходимости остановки компонента эта функция вначале корректно останавливает работу всех зависимых компонентов. Oracle Restart выполняется из домашнего каталога инфраструктуры распределенных вычислений Oracle Grid Infrastructure, который устанавливается отдельно от домашних каталогов баз данных Oracle.

1

Архитектура базы данных Oracle

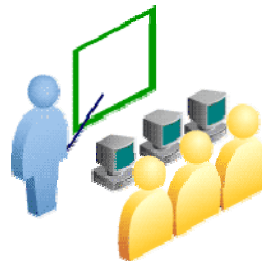
ORACLE

© Oracle, 2009. Все права защищены.

Цели

Изучив материал этого занятия, вы освоите следующие темы:

- список основных компонентов архитектуры базы **данных Oracle**;
- структуры памяти;
- фоновые процессы;
- разница между логическими и физическими структурами хранения;
- компоненты хранения ASM.



ORACLE

© Oracle, 2009. Все права защищены.

Цели

В этом уроке подробно описывается архитектура базы данных Oracle. Вы узнаете о физических и логических структурах, а также о различных компонентах.

Oracle Database

Система управления реляционными базами данных (СУРБД) Oracle обеспечивает открытый, всесторонний и интегрированный подход к управлению информацией.



ORACLE

© Oracle, 2009. Все права защищены.

Oracle Database

База данных – это набор данных, представляющий собой единое целое. Задачей базы данных является хранение и извлечение связанных данных.

Система управления реляционными базами данных Oracle (СУРБД) надежно управляет большим количеством данных в многопользовательской среде, предоставляя нескольким пользователям параллельный доступ к одним и тем же данным. При этом сохраняется высокая производительность работы. В то же время СУРБД предотвращает несанкционированный доступ и предоставляет надежные инструменты для восстановления базы данных после сбоя.

Подключение к серверу



Пример многоярусной архитектуры

ORACLE

© Oracle, 2009. Все права защищены.

Подключение к серверу

Пользователь базы данных может подключиться к серверу Oracle одним из трех способов:

- Пользователь осуществляет вход в операционную систему с экземпляром Oracle, а затем запускает приложение или средство для обращения к базе данных в этой системе. Устанавливается канал связи с помощью механизмов межпроцессного взаимодействия, доступных в операционной системе хоста.
- Пользователь запускает приложение или средство на локальном компьютере и подключается через сеть к компьютеру с базой данных Oracle. В такой конфигурации (ее называют *клиент-серверной*) для взаимодействия между пользователем и сервером используется сетевое программное обеспечение. Система базы данных с клиент-серверной архитектурой состоит из двух частей: внешней (клиентская часть) и внутренней (серверная часть). Для взаимодействия пользователя и сервера Oracle используется сетевое программное обеспечение.
 - Клиент представляет собой приложение базы данных, инициирующее запрос на выполнение операции на сервере баз данных. Оно запрашивает, обрабатывает и отображает управляемые сервером данные. Клиентская рабочая станция может быть для этого специальным образом оптимизирована. Например, клиент может не нуждаться в большом объеме дискового пространства или активно пользоваться графическими возможностями. Как правило, клиент размещается на отдельном компьютере, а не на сервере базы данных. С одним сервером может одновременно работать множество клиентов.

Подключение к серверу (продолжение)

- На сервере запускается программное обеспечение Oracle Database и выполняются функции, необходимые для обеспечения параллельного общего доступа к данным. Сервер получает и обрабатывает запросы, получаемые от клиентских приложений. Компьютер, управляющий сервером, можно оптимизировать для выполнения определенных задач. Например, серверный компьютер может иметь большой объем дискового пространства и быстрые процессоры.
- Пользователь обращается к серверу приложений с помощью программы (например, Web-браузера) на локальном компьютере (клиенте). После этого сервер приложений обращается к серверу базы данных по запросу клиента.

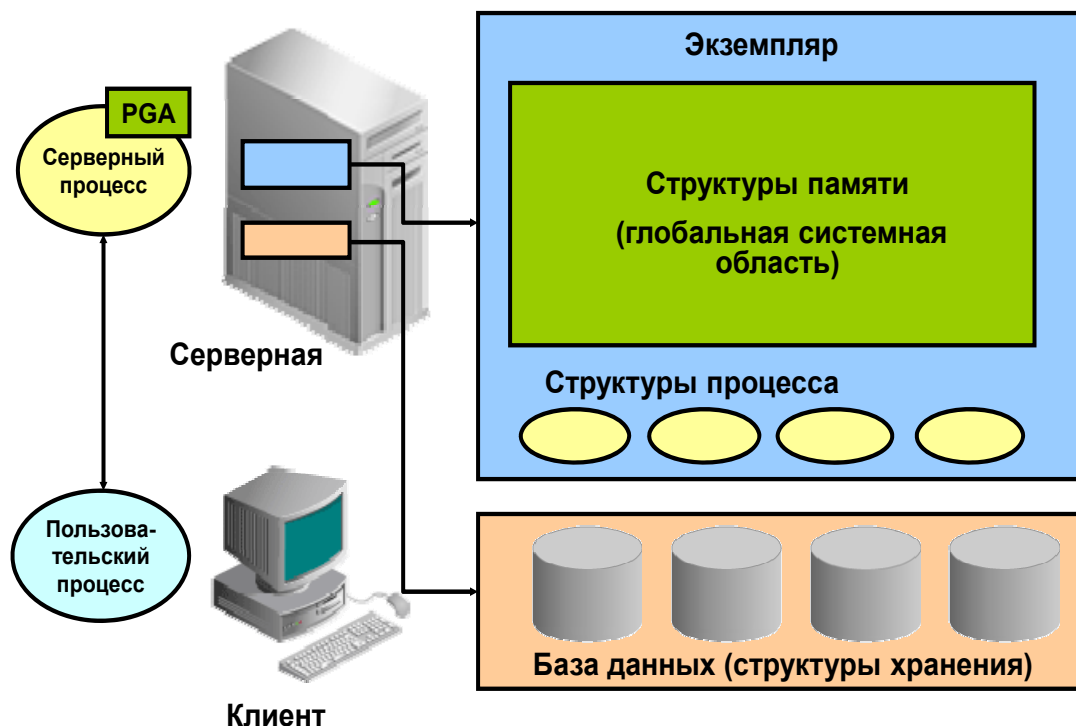
Традиционная многоярусная архитектура содержит такие компоненты:

- Процесс клиента или инициатора, который начинает операцию.
- Один или несколько серверов приложений, выполняющие разные части операции. Сервер приложений содержит значительную часть логики приложения и обеспечивает клиенту доступ к данным, а также выполняет обработку некоторых запросов, благодаря чему снижается нагрузка на сервер баз данных. Сервер приложений может выполнять роль интерфейса между клиентами и множеством серверов баз данных, а также может обеспечивать дополнительную безопасность.
- Конечный сервер или сервер баз данных, на котором хранится большая часть используемых в операции данных.

Такая архитектура позволяет использовать сервер приложений для следующих целей:

- проверка реквизитов клиента (например, Web-браузера);
- подключение к серверу баз данных Oracle;
- выполнение операции по запросу клиента.

Архитектура сервера баз данных Oracle: обзор



ORACLE

© Oracle, 2009. Все права защищены.

Архитектура сервера баз данных Oracle

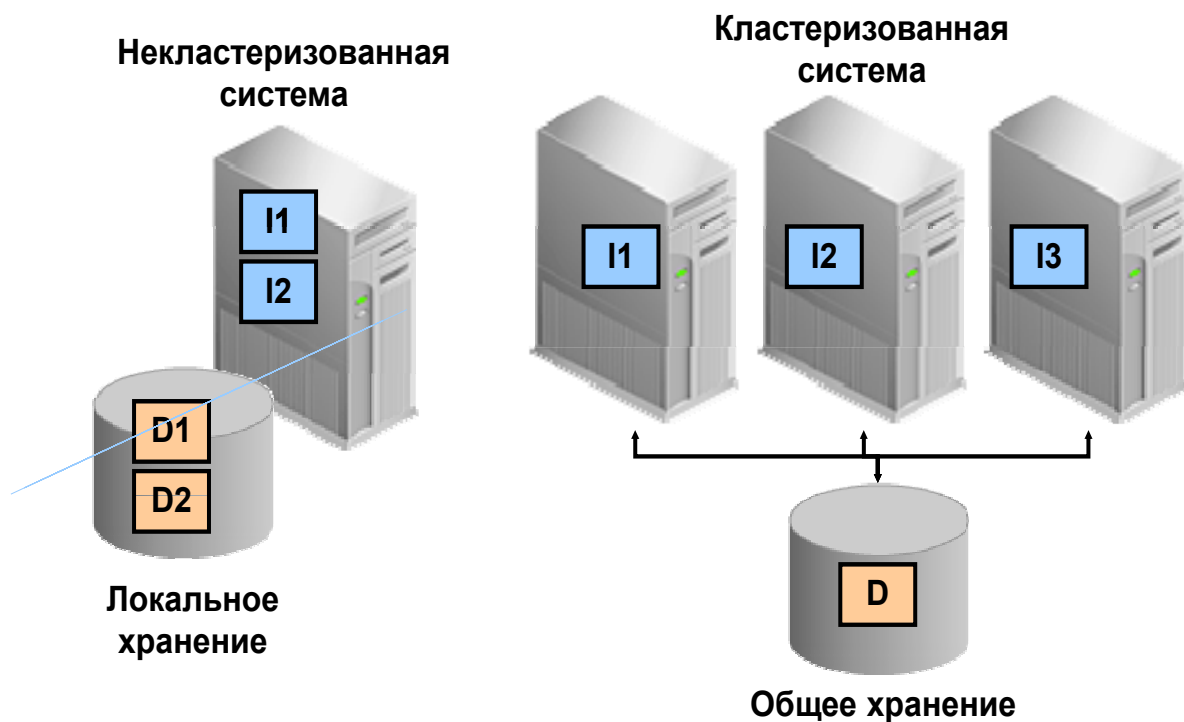
Архитектура сервера баз данных Oracle Database состоит из трех основных структур: структур памяти, процессов и хранения. Стандартная система базы данных Oracle состоит из базы данных Oracle и экземпляра базы данных.

База данных состоит как из физических, так и из логических структур. Так как физические и логические структуры разделены, физическими хранилищами данных можно управлять, не влияя на логические структуры хранения.

Экземпляр состоит из структур памяти и фоновых процессов, связанных с этим экземпляром. При каждом запуске экземпляра выделяется совместно используемая область памяти, которая называется глобальной системной областью (System Global Area, SGA), а также запускаются фоновые процессы. Процессы – это задания, которые выполняются в памяти компьютеров. Процесс можно описать как "управляющий поток" или механизм операционной системы, который может выполнять наборы действий. После запуска экземпляра базы данных программное обеспечение Oracle ассоциирует экземпляр с конкретной базой данных. Этот процесс называется *монтированием базы данных*. После этого база данных будет готовой к открытию, что сделает ее доступной для авторизованных пользователей.

Примечание. Система автоматического управления хранением Oracle Automatic Storage Management (ASM) использует экземпляры для компонентов памяти и процессов, но не связывает их с определенной базой данных.

Экземпляр: конфигурации баз данных



ORACLE

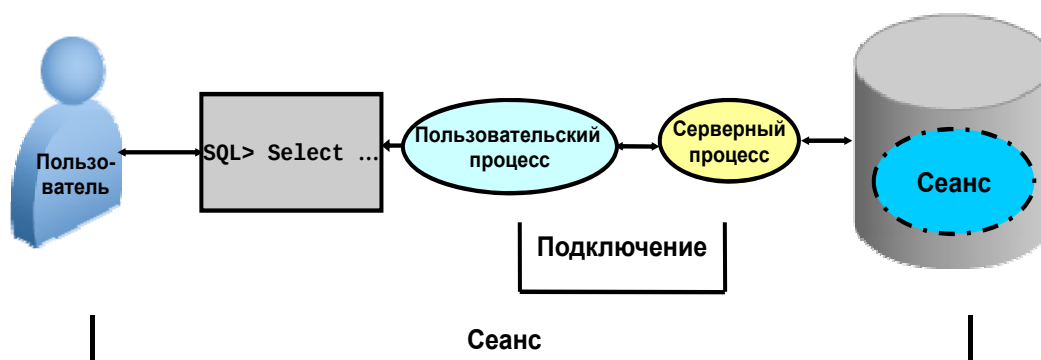
© Oracle, 2009. Все права защищены.

Экземпляр: конфигурации баз данных

Каждый экземпляр связан только с одной базой данных. При наличии нескольких баз данных на сервере для каждой из них имеется собственный экземпляр. Общий доступ к одному экземпляру невозможен. База данных Real Applications Cluster (RAC) обычно имеет несколько экземпляров одной и той же общей базы на различных серверах. В этой модели с каждым экземпляром RAC связана одна и та же база. Это соответствует требованию о том, что с экземпляром может быть связана максимум одна база данных.

Подключение к экземпляру базы данных

- Подключение: взаимодействие пользовательского процесса и экземпляра
- Сеанс: конкретное подключение пользователя к экземпляру посредством пользовательского процесса



ORACLE

© Oracle, 2009. Все права защищены.

Подключение к экземпляру базы данных

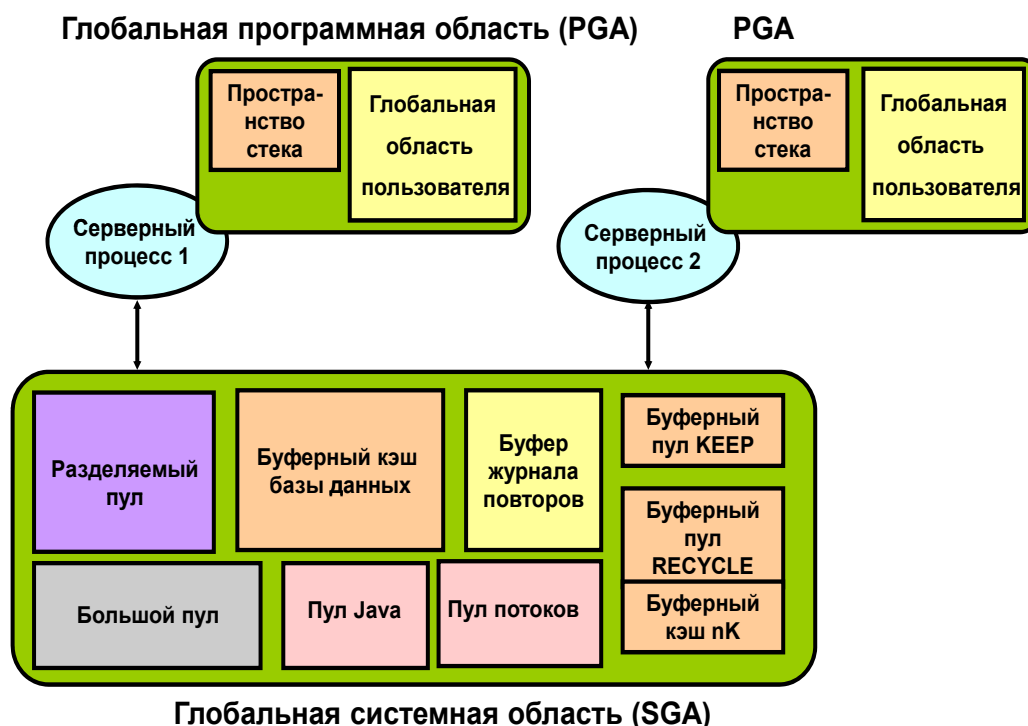
Подключения и сеансы тесно связаны с пользовательскими процессами, но при этом они имеют весьма различные значения.

Подключение – это канал связи между пользовательским процессом и экземпляром базы данных Oracle. Канал связи устанавливается с помощью доступного механизма межпроцессного взаимодействия (на компьютере, на котором запущены как пользовательский процесс, так и база данных Oracle) или же с помощью сетевого программного обеспечения (когда на разных компьютерах, взаимодействующих через сеть, запущено приложение базы данных и база данных Oracle).

Сеанс представляет собой состояние подключения текущего пользователя к экземпляру базы данных. Например, когда пользователь запускает приложение SQL*Plus, он должен предоставить действительное имя пользователя и пароль, после чего для данного пользователя будет открыт сеанс. Сеанс длится с момента подключения пользователя и до момента отключения или выхода из приложения базы данных.

Для одного пользователя базы данных Oracle может быть создано множество параллельных сеансов с одним и тем же именем пользователя. Например, пользователь с именем/паролем HR/HR может подключиться к одному экземпляру базы данных Oracle несколько раз.

Структуры памяти базы данных Oracle



ORACLE

© Oracle, 2009. Все права защищены.

Структуры памяти базы данных Oracle

База данных Oracle создает и использует структуры памяти для различных целей.

Например, в памяти хранится код выполняемой программы, данные, разделяемые среди пользователей, и частные области данных для каждого подключенного пользователя.

С экземпляром связаны две базовые структуры памяти:

- **Глобальная системная область (SGA):** группа разделяемых структур памяти (компонентов SGA), которые содержат данные и управляющую информацию для одного экземпляра базы данных Oracle. SGA является общей для всех фоновых и серверных процессов. Примерами данных, хранимых в SGA, являются кэшированные блоки данных и разделяемые области SQL.
- **Глобальная программная область (PGA):** области памяти, которые содержат данные и управляющую информацию для фонового или серверного процесса. PGA – это неразделяемая область памяти, создаваемая базой данных Oracle при запуске фонового или серверного процесса. К PGA может обращаться только серверный процесс. Каждый фоновый или серверный процесс имеет свою область PGA.

Структуры памяти базы данных Oracle (продолжение)

SGA – это область памяти, которая содержит данные и управляющую информацию для экземпляра. В SGA входят следующие структуры данных:

- **Разделяемый пул:** кэширует различные построения, которые можно разделять между пользователями.
- **Буферный кэш базы данных:** кэширует блоки данных, извлеченные из базы данных.
- **Пул буфера КЕЕР:** специализированный тип кэша буфера базы данных, который настроен на длительное хранение блоков данных в памяти.
- **Пул буфера RECYCLE:** специализированный тип кэша буфера базы данных, который настроен на повторное использование или удаление блоков данных из памяти.
- **Кэш буфера пК:** один из нескольких специализированных кэшей буфера базы данных, который настроен на хранение блоков данных с размером, отличным от размеров блоков по умолчанию.
- **Буфер журнала повторов:** кэширует данные для операций повторов (используемых для восстановления экземпляра), пока они не будут записаны в физические файлы журнала повторов, хранимые на диске.
- **Большой пул:** необязательная область, позволяющая выделять большие объемы памяти для определенных ресурсоемких процессов, таких как операции резервного копирования и восстановления, а также серверные процессы ввода/вывода.
- **Пул Java:** используется для всего сеансового Java-кода и данных в виртуальной машине Java (JVM).
- **Пул потоков:** используется в Oracle Streams для хранения данных, необходимых для операций захвата и применения.

При запуске экземпляра с помощью Enterprise Manager или SQL*Plus будет отображен объем памяти, выделенной для SGA.

Глобальная программная область (PGA) – это область памяти, которая содержит данные и управляющую информацию для каждого серверного процесса. Серверный процесс Oracle обслуживает запросы клиента. Каждому серверному процессу соответствует собственная частная область PGA, создаваемая при запуске серверного процесса. Доступ к PGA имеет только соответствующий серверный процесс. Кроме того, операции чтения и записи в PGA осуществляются только кодом Oracle, действующим по запросу PGA. PGA делится на две основных области: пространство стека и глобальную пользовательскую область (UGA).

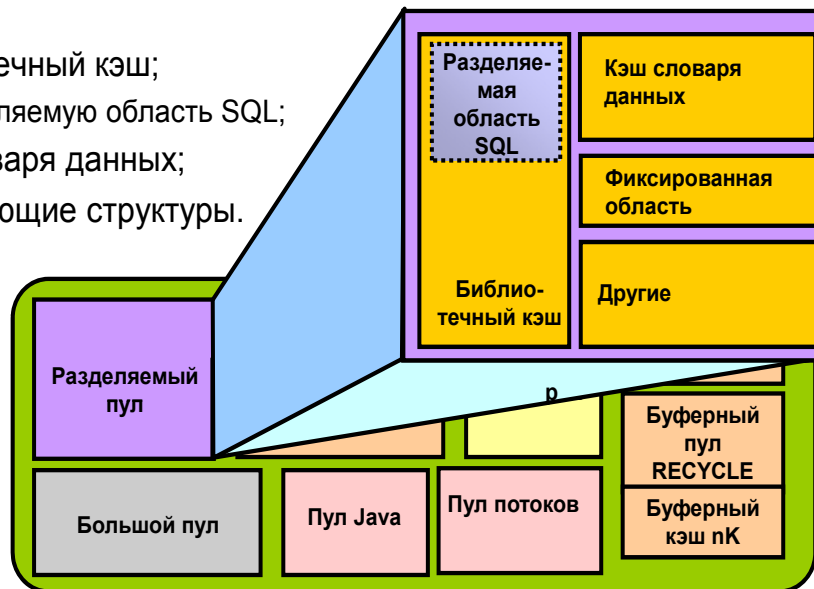
Благодаря динамической инфраструктуре SGA размеры буферного кэша базы данных, разделяемого пула, большого пула, пула Java и пула потоков можно изменять без остановки работы экземпляра.

База данных Oracle использует параметры инициализации для создания и управления структурами памяти. Самый простой способ управления памятью – позволить базе данных автоматически осуществлять управление и настройку памяти. Для этого (на большинстве платформ) необходимо установить параметр инициализации планируемого объема памяти (MEMORY_TARGET) и параметр инициализации максимального объема памяти (MEMORY_MAX_TARGET).

Разделяемый пул

- Представляет собой область SGA
- Содержит:

- библиотечный кэш;
- разделяемую область SQL;
- кэш словаря данных;
- управляющие структуры.



Глобальная системная область (SGA)

ORACLE

© Oracle, 2009. Все права защищены.

Разделяемый пул

Область разделяемого пула в SGA содержит библиотечный кэш, кэш словаря данных, кэш результатов обработки SQL-запросов, кэш результатов обработки функций PL/SQL, буферы для сообщений о параллельном выполнении и управляющие структуры.

Словарь данных представляет собой набор таблиц и представлений базы данных, содержащих связанную информацию о базе данных, ее структурах и пользователях. База данных Oracle регулярно обращается к словарию данных при синтаксическом разборе SQL-операторов. Такие обращения предельно важны для функционирования базы данных Oracle.

База данных Oracle настолько часто обращается к словарию данных, что для хранения его **данных выделены две специальные области памяти. Первая область называется кэшем словаря данных.** Ее также называют кэшем строк, поскольку она хранит данные в виде строк, а не буферов (в которых хранятся целые блоки данных). Другая область памяти для хранения данных словаря называется *библиотечным кэшем*. Все процессы пользователей базы данных Oracle совместно пользуются этими двумя кэшами для доступа к информации из словаря данных.

База данных Oracle отображает каждый обрабатываемый SQL-оператор в разделяемой области SQL (в то время как частная область SQL расположена в PGA). База данных Oracle отслеживает ситуации, когда два пользователя выполняют одинаковый SQL-оператор, а затем повторно использует для них разделяемую область SQL.

Разделяемый пул (продолжение)

Разделяемая область SQL содержит дерево синтаксического разбора и план выполнения заданного SQL-оператора. База данных Oracle экономит память, используя одну разделяемую область SQL для SQL-операторов, выполняемых несколько раз, что обычно случается, когда много пользователей работают с одним приложением.

При синтаксическом разборе нового SQL-оператора база данных Oracle выделяет память в разделяемом пуле для хранения разделяемой области SQL. Размер этой памяти зависит от сложности оператора.

База данных Oracle обрабатывает программные блоки PL/SQL (процедуры, функции, пакеты, анонимные блоки и триггеры базы данных) практически так же, как и отдельные SQL-операторы. База данных Oracle выделяет совместно используемую область для хранения синтаксически разобранной и скомпилированной формы программного блока. База данных Oracle выделяет частную область для хранения значений конкретного сеанса, который запускает программный блок, включая локальные и глобальные переменные, переменные пакетов (также называемые экземплярами пакета), а также буферы для выполнения SQL. Если к программному блоку обращаются несколько пользователей, для всех их используется единая разделяемая область, при этом у каждого пользователя есть своя отдельная копия частной области SQL, в которой хранятся значения переменных сеанса этого пользователя.

Отдельные SQL-операторы, содержащиеся в программном блоке PL/SQL, обрабатываются аналогично другим SQL-операторам. Несмотря на то, что SQL-операторы происходят из программного блока PL/SQL, они используют разделяемую область для хранения их синтаксически разобранных представлений и частную область для каждого сеанса, запускающего оператор.

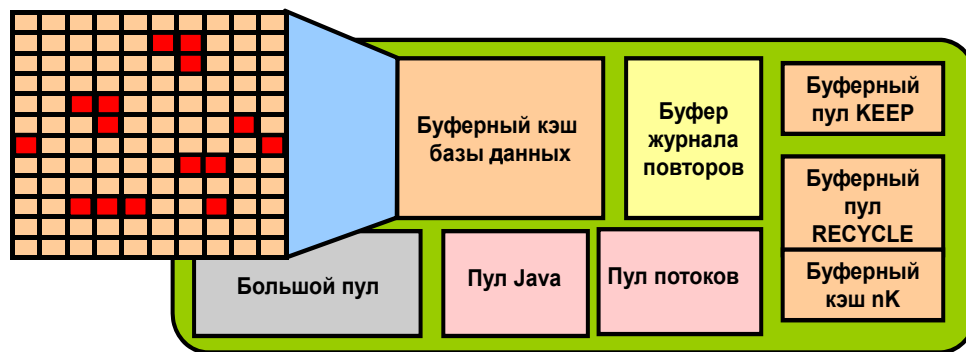
Кэш результатов обработки SQL-запросов и кэш результатов обработки функций PL/SQL являются нововведением в продукте Oracle Database 11g. Они используют одну и ту же архитектуру, отображаются посредством динамических представлений производительности (V\$) и администрируются с помощью одного и того же пакета.

Результаты запросов и их фрагменты могут кэшироваться в памяти в кэше результатов обработки SQL-запросов. После этого база данных сможет использовать кэшированные результаты для будущего выполнения таких же запросов и их фрагментов. Поскольку извлечение результатов из кэша результатов обработки SQL-запросов выполняется значительно быстрее, чем повторная обработка запроса, при выполнении часто используемых запросов можно достичь существенного повышения производительности, если их результаты находятся в кэше.

Функция PL/SQL иногда используется для возврата результата вычисления, в котором входными параметрами являются один или несколько параметризованных запросов, вызываемых функцией. В некоторых случаях такие запросы обращаются к данным, изменяемым довольно редко по сравнению с частотой вызова функции. В исходном тексте SQL-функции можно использовать синтаксические конструкции, чтобы указать, что результат следует сохранить в кэше результатов обработки PL/SQL-функций, а также убедиться в том, что кэш будет очищен, когда в таблицах из списка таблиц попадет DML. Фиксированная область разделяемого пула представляет нагрузку на SGA при запуске. Она крайне мала по сравнению со стандартным разделяемым пулом или областью SGA.

Буферный кэш базы данных

- Является частью SGA
- Хранит копии блоков данных, считанных из файлов данных
- Совместно используется всеми одновременно работающими пользователями



Глобальная системная область (SGA)

ORACLE

© Oracle, 2009. Все права защищены.

Буферный кэш базы данных

Буферный кэш базы данных — это часть области SGA, которая хранит образы блоков, считанных из файлов данных или динамически построенных в соответствии с моделью целостности чтения. Все пользователи, одновременно подключенные к экземпляру, получают общий доступ к буферному кэшу базы данных.

Когда процессу пользователя базы данных Oracle в первый раз требуется определенный фрагмент данных, он ищет нужные данные в буферном кэше. Если процесс найдет данные в кэше (попадание в кэш), он сможет считать их непосредственно из памяти. Если же процесс не сможет найти данные в кэше (отсутствие в кэше), он будет вынужден скопировать блок данных из файла данных, размещенного на диске, в буфер кэша, и только после этого сможет обратиться к данным. Доступ к данным, попавшим в кэш, осуществляется значительно быстрее, чем при их отсутствии в кэше.

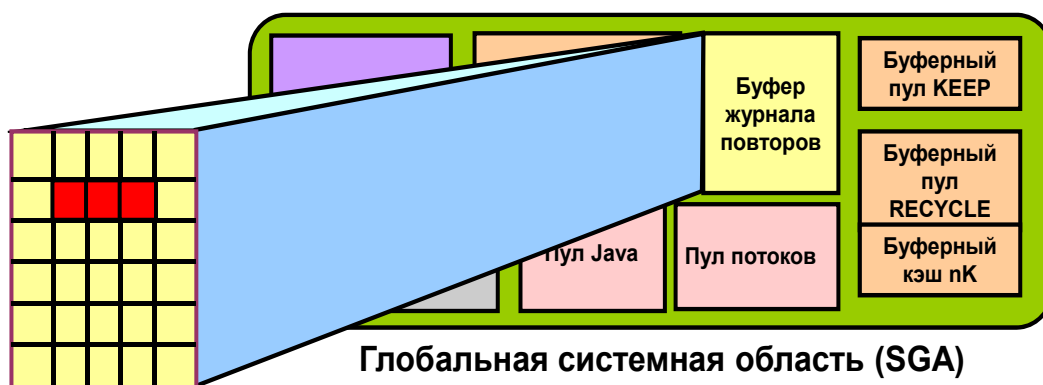
Буферы кэша управляются сложным алгоритмом, использующим комбинации списков страниц, которые дольше всего не использовались (LRU), а также счетчик обращений. Список LRU обеспечивает хранение наиболее часто используемых блоков в памяти для снижения интенсивности обращения к диску.

Буферные пулы KEEP и RECYCLE используются для специальной настройки буферного пула. Пул KEEP предназначен для хранения буферов в памяти дольше срока, предусмотренного для списка LRU. Пул RECYCLE предназначен для удаления буферов из памяти раньше срока, предусмотренного для списка LRU.

Можно настроить дополнительные буферные кэши для хранения блоков с размером, отличным от размера по умолчанию.

Буфер журнала повторов

- Представляет собой циклический буфер в SGA
- Хранит информацию об изменениях базы данных
- Содержит записи повторов, в которых хранится **информация для повторного применения изменений**, внесенных такими операциями, как DML и DDL



ORACLE

© Oracle, 2009. Все права защищены.

Буфер журнала повторов

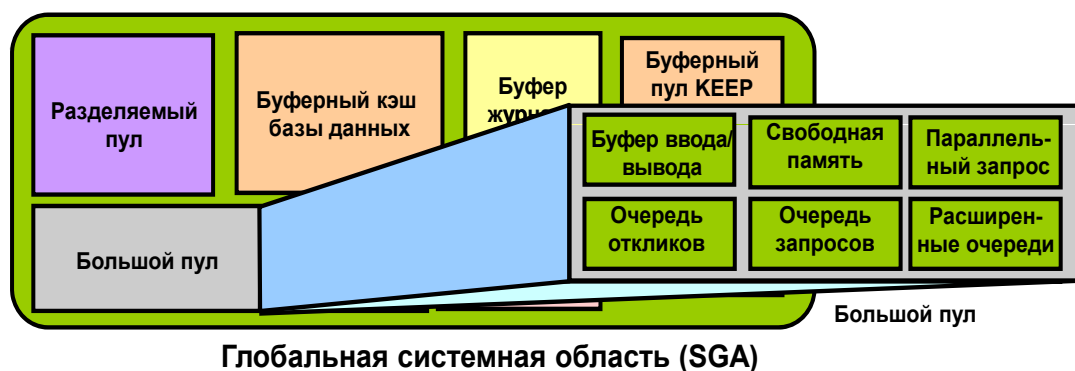
Буфер журнала повторов представляет собой циклический буфер в SGA, в котором хранится информация об изменениях, внесенных в базу данных. Эта информация хранится в записях повторов. Записи повторов содержат информацию, необходимую для воссоздания (или повторения) изменений, внесенных в базу данных операциями DML, DDL или внутренними операциями. Записи повторов используются для восстановления базы данных в случае необходимости.

По мере изменений буферного кэша сервером создаются записи повторов, которые записываются в буфер журнала повторов области SGA. Записи повторов создаются непрерывно и располагаются последовательно в буфере. Фоновый процесс записи производит запись буфера журнала повторов на диск в активный файл журнала повторов (или группу файлов).

Большой пул

Предоставляет большие области памяти для:

- памяти сеансов для разделяемого сервера и интерфейса Oracle XA;
- серверных процессов ввода/вывода;
- операций резервного копирования и восстановления базы данных Oracle.



ORACLE

© Oracle, 2009. Все права защищены.

Большой пул

Администратор базы данных может настроить необязательную область памяти, которая называется *большим пулом*, чтобы обеспечить возможность выделять большие области памяти для:

- памяти сеансов для разделяемого сервера и интерфейса Oracle XA (используется в том случае, если транзакции взаимодействуют с несколькими базами данных);
- серверных процессов ввода/вывода;
- операций резервного копирования и восстановления базы данных Oracle;
- операций параллельного запроса;
- хранения таблицы памяти расширенных очередей.

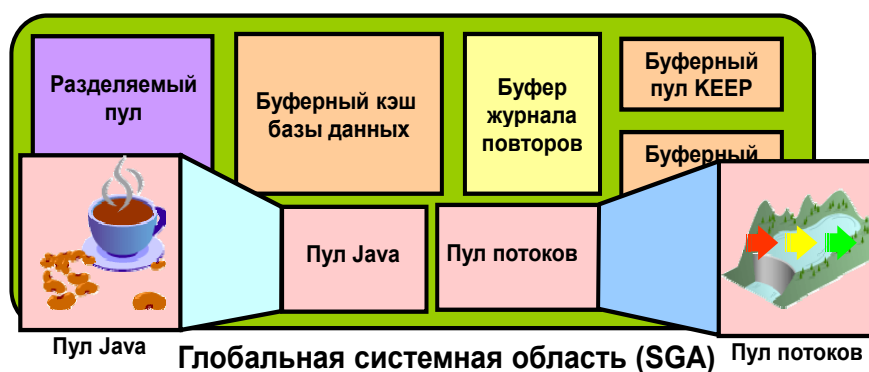
Благодаря возможности выделения памяти сеансов из большого пула для разделяемого сервера, интерфейса Oracle XA или буферов параллельных запросов база данных Oracle сможет использовать разделяемый пул в основном для кэширования разделяемых областей SQL и избежать избыточной нагрузки, возникающей из-за сжатия кэша разделяемых областей SQL.

Кроме того, память для операций резервного копирования и восстановления базы данных Oracle, серверных процессов ввода/вывода и параллельных буферов выделяется в буферах, которые имеют размер порядка нескольких сотен килобайт. Большой пул в большей степени способен удовлетворять запросы на выделение больших областей памяти, чем разделяемый пул.

Управление большим пулом не осуществляется через наименее часто использовавшихся объектов (LRU).

Пул Java и пул потоков

- Память пула Java используется для хранения всего принадлежащего сеансу Java-кода и данных в JVM.
- Память пула потоков используется только в Oracle Streams для:
 - хранения сообщений о буферизованных запросах;
 - обеспечения памятью процессов Oracle Streams.



ORACLE

© Oracle, 2009. Все права защищены.

Пул Java и пул потоков

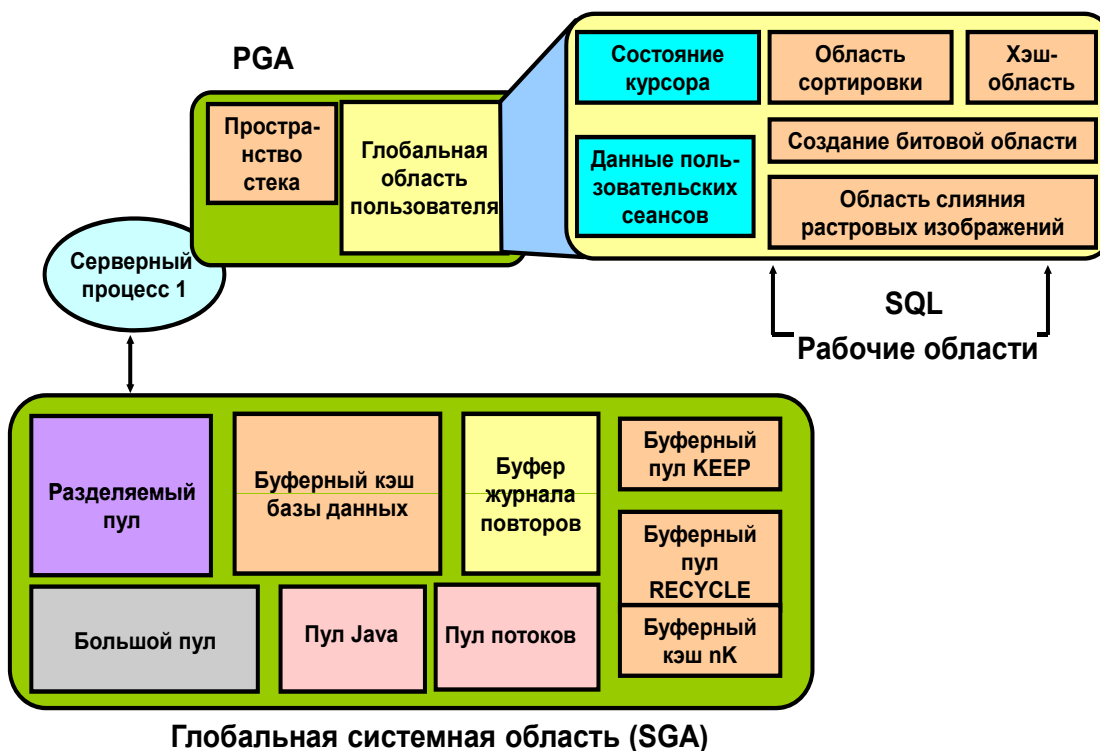
Память пула Java используется для хранения всего принадлежащего сеансу Java-кода и данных в JVM. Память пула Java используется различными способами в зависимости от режима, в котором работает база данных Oracle.

Пул потоков используется исключительно в Oracle Streams. Пул потоков хранит сообщения о буферизованных запросах и предоставляет память для процессов применения и захвата Oracle Streams.

Если не производилась настройка пула потоков, его размер будет начинаться с нулевой отметки. Размер пула меняется динамически в зависимости от потребностей Oracle Streams.

Примечание. Подробное обсуждение Oracle Streams и программирования на Java выходит за рамки данного руководства.

Глобальная программная область (PGA)



ORACLE

© Oracle, 2009. Все права защищены.

Глобальная программная область (PGA)

Глобальная область программ (PGA) – это закрытая область памяти, содержащая данные и управляющую информацию для серверного процесса. Каждый серверный процесс имеет собственную глобальную системную область PGA. Доступ к ней предоставляется исключительно этому серверному процессу, а операции чтения для этой области выполняются только кодом Oracle, действующим от имени этого процесса. Эта область недоступна для кода разработчика.

Каждая область PGA содержит пространство стека. В среде с выделенными серверами для каждого пользователя, подключающегося к экземпляру базы данных, имеется отдельный серверный процесс. Для этого типа подключения область PGA содержит подкатегорию памяти — так называемую глобальную пользовательскую область (UGA). Область UGA состоит из следующих компонентов:

- область курсора для хранения информации выполнения в курсорах;
- область хранения данных пользовательских сеансов для управления информацией сеанса;
- рабочие области SQL для обработки операторов SQL, состоящие из следующих компонентов:
 - область сортировки для функций упорядочивания данных (например, `ORDER BY` и `GROUP BY`);
 - область хэша для объединений аблиц по хэшам;
 - область создания битовых образов, используемая при создании битовых индексов для информационных хранилищ;
 - область слияния битовых образов для определения выполнения плана битовых индексов.

В среде с общими серверами серверный процесс совместно используется несколькими клиентами. В данной модели область UGA перемещается в область SGA (разделяемый пул или большой пул (при наличии)), а в области PGA остается только пространство стека.

Экспресс-тест

Как называется область памяти, которая содержит данные и управляющую информацию для фонового или серверного процесса?

1. разделяемый пул
2. PGA
3. буферный кэш
4. данные пользовательского сеанса

ORACLE

© Oracle, 2009. Все права защищены.

Ответ: 2

Экспресс-тест

Что считывается из файлов данных в буферный кэш базы данных?

1. строки
2. **изменения**
3. блоки
4. SQL

ORACLE

© Oracle, 2009. Все права защищены.

Ответ: 3

Архитектура процессов

- Пользовательский процесс
 - приложение или средство для подключения к базе данных Oracle
- Процессы базы данных
 - Серверный процесс: подключается к экземпляру Oracle и запускается при установлении сеанса пользователем
 - Фоновые процессы: запускаются при запуске экземпляра Oracle
- Демон / процессы приложений
 - сетевые приемники
 - демоны инфраструктуры распределенных вычислений

ORACLE

© Oracle, 2009. Все права защищены.

Архитектура процессов

Процессы в системе баз данных Oracle можно разделить на три основные группы:

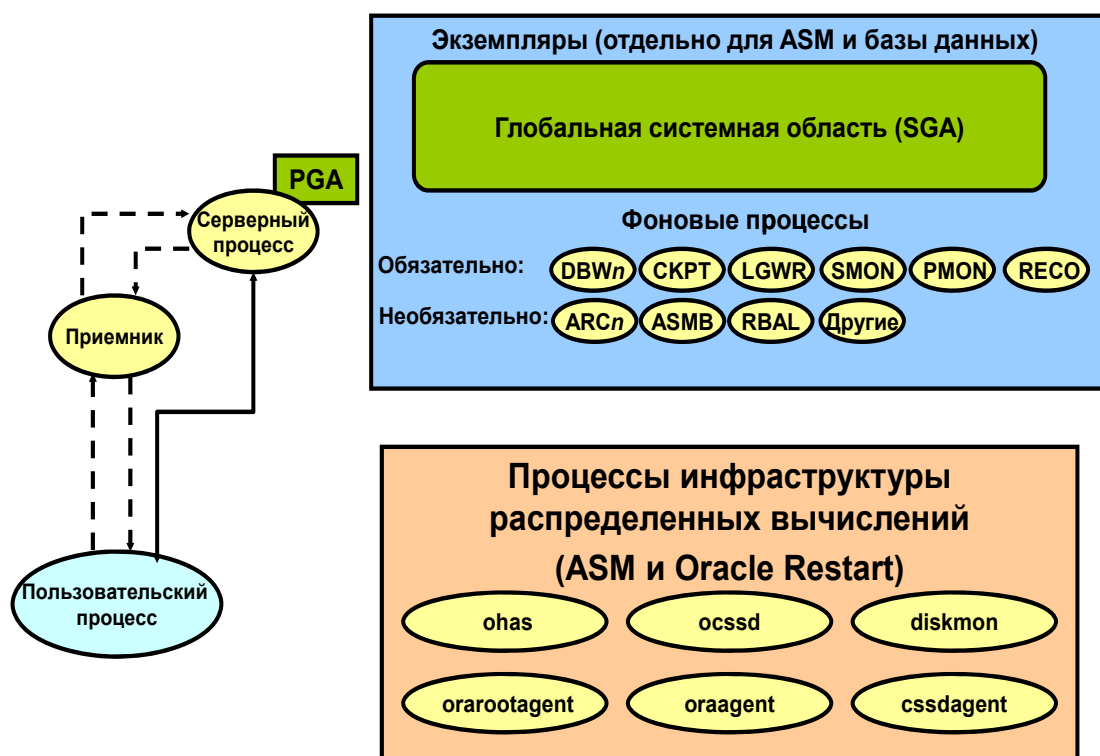
- пользовательские процессы, выполняющие код приложения или инструмента Oracle
- процессы базы данных Oracle, выполняющие код сервера баз данных Oracle (включая серверные и фоновые процессы)
- демоны Oracle и процессы приложений, не относящиеся к конкретной базе данных

Когда пользователь запускает приложение или инструмент Oracle, например SQL*Plus, база данных Oracle создает *пользовательский процесс* для ссылки на приложение пользователя.

Пользовательский процесс не обязательно должен присутствовать на компьютере сервера баз данных. База данных Oracle также создает *серверный процесс* для выполнения команд пользовательского процесса. Кроме того, сервер Oracle также порождает набор *фоновых процессов* для экземпляра, которые взаимодействуют друг с другом и с операционной системой, управляя структурами памяти, выполняя асинхронные операции ввода/вывода для записи данных на диск и другие задачи. Структура процессов различна для каждой конфигурации базы данных Oracle в зависимости от используемой операционной системы и настроек базы данных Oracle. Код для подключенных пользователей можно настроить как выделенный или разделяемый сервер.

- **Выделенный сервер:** для каждого сеанса приложение базы данных запускается пользовательским процессом, который обслуживается выделенным серверным процессом, исполняющим код сервера баз данных Oracle.
- **Разделяемый сервер:** устраняет необходимость в выделенном серверном процессе для каждого подключения. Диспетчер направляет множество входящих сетевых запросов на установку сеансов пулу процессов разделяемого сервера. Любой запрос клиента обслуживается процессом разделяемого сервера.

Структуры процесса



ORACLE

© Oracle, 2009. Все права защищены.

Структуры процесса

Серверные процессы

База данных Oracle создает серверные процессы для управления запросами на подключение к экземпляру, которые исходят от процессов пользователей. Пользовательский процесс представляет приложение или средство для подключения к базе данных Oracle. Он может находиться на компьютере базы данных Oracle или на удаленном клиенте с использованием сети для доступа к базе данных Oracle. Пользовательский процесс вначале взаимодействует с процессом приемника, который создает серверный процесс в выделенной среде.

Серверные процессы, созданные по запросу приложений каждого пользователя, могут выполнять одну или несколько функций:

- выполнять синтаксический разбор и обрабатывать SQL-операторы, переданные через приложение;
- считывать нужные блоки данных из файлов данных на диске в разделяемые буферы базы данных SGA (если блоки пока не присутствуют в SGA);
- возвращать результаты в такой форме, в которой приложение сможет обрабатывать данные.

Фоновые процессы

Чтобы максимально повысить производительность и обслуживать множество пользователей, многопроцессная система баз данных Oracle использует некоторые дополнительные процессы базы данных Oracle, которые называются *фоновыми процессами*. Экземпляр базы данных Oracle может иметь множество фоновых процессов.

Структуры процессов (продолжение)

Фоновые процессы обычно происходят в средах, отличных от RAC и ASM. К фоновым процессам относятся:

- процесс записи в базу данных (DBWn);
- процесс записи в журнал (LGWR);
- процесс создания контрольной точки (CKPT);
- процесс системного монитора (SMON);
- процесс монитора процессов (PMON);
- процесс восстановления (RECO);
- координатор очередей заданий (CJQ0);
- подчиненные процессы заданий (Jnnn);
- процессы архивации (ARcn);
- процессы монитора очередей (QMNn).

В более сложных конфигурациях (например, RAC) могут присутствовать и другие фоновые процессы. Для получения дополнительных сведений о фоновых процессах см. представление V\$BGPROCESS.

Некоторые фоновые процессы запускаются автоматически при запуске экземпляра, а другие запускаются по требованию.

Другие структуры процессов не относятся к определенной базе данных, а могут совместно использоваться несколькими базами данных на одном сервере. В эту категорию попадает инфраструктура распределенных вычислений Grid Infrastructure и сетевые процессы.

Процессы Oracle Grid Infrastructure в системах Linux и UNIX:

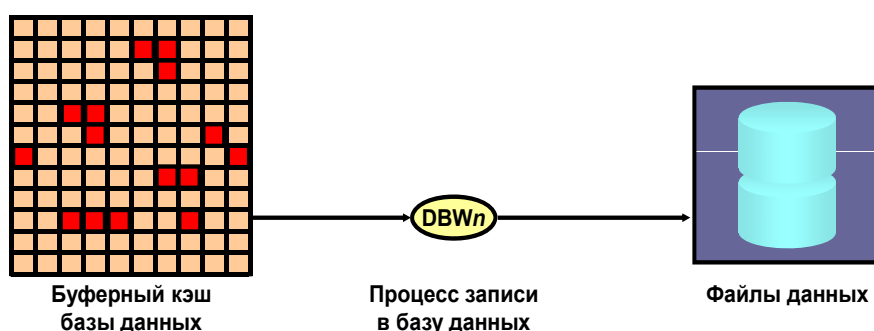
- **ohasd**: демон службы высокой доступности Oracle High Availability Service, запускающий процессы Oracle Clusterware;
- **ocssd**: демон службы синхронизации кластеров Cluster Synchronization Service;
- **diskmon**: демон отслеживания диска Disk Monitor, выполняющий закрепление ввода и вывода для сервера HP Oracle Exadata Storage Server;
- **cssdagent**: запускает, останавливает и проверяет статус демона CSS ocssd;
- **oraagent**: расширение clusterware для поддержки специфических требований и комплексных ресурсов Oracle;
- **orarootagent**: специальный процесс агента Oracle, управляющий ресурсами пользователя root, например сетью.

Примечание. Подробный список фоновых процессов см. в приложении *Фоновые процессы Oracle* в этом курсе или в руководстве *Справочник по базам данных*.

Процесс записи в базу данных (DBWn)

Записывает измененные (заполненные) буферы из буферного кэша базы данных на диск:

- асинхронно во время выполнения другой обработки
- для приближения контрольной точки



ORACLE

© Oracle, 2009. Все права защищены.

Процесс записи в базу данных (DBWn)

Процесс записи в базу данных (DBWn) записывает содержимое буферов в файлы данных. Процессы DBWn осуществляют запись измененных (заполненных) буферов из буферного кэша базы данных на диск. Хотя наличие одного процесса записи в базу данных (DBW0) достаточно для большинства систем, можно настроить дополнительные процессы (DBW1 - DBW9 и DBWa - DBWz) для повышения производительности при записи, если в системе выполняется много изменений данных. Дополнительные процессы DBWn бесполезны в однопроцессорных системах.

Если буфер в буферном кэше базы данных изменяется, он помечается как заполненный и добавляется в начало очереди контрольных точек, отсортированной по номеру SCN. Таким образом, порядок буферов совпадает с порядком операций повтора, записанных в журналы повторов для таких измененных буферов. Когда количество доступных буферов в буферном кэше становится меньше внутреннего порогового значения (при котором серверным процессам становится трудно получать доступные буферы), DBWn записывает редко используемые буферы в файлы данных с конца списка LRU, чтобы процессы при необходимости заменяли буферы. DBWn также выполняет запись с конца очереди контрольных точек для продвижения очереди.

Процесс записи в базу данных (DBWn) (продолжение)

SGA содержит структуру памяти, в которой находится журнальный байтовый адрес (RBA) позиции в потоке повторов, с которой следует начать восстановление в случае отказа экземпляра. Данная структура служит указателем на повтор и записывается в контрольный файл процессом CKPT один раз в три секунды. Поскольку процесс DBWn записывает заполненные буферы в порядке следования SCN, а журнал также формируется по порядку номеров SCN, то каждый раз, когда процесс DBWn записывает заполненные буферы из списка LRUW, он также перемещает указатель в структуре памяти SGA. Таким образом, при восстановлении экземпляра (если это необходимо) чтение начинается с приблизительно правильного размещения, что позволяет избежать лишних операций ввода-вывода. Это называется *инкрементной установкой контрольных точек*.

Примечание. Процесс DBWn может выполнять запись и в других ситуациях (например, если табличные пространства становятся доступными только для чтения или переводятся в автономный режим). В таких случаях инкрементная установка контрольной точки не происходит, так как в базу данных записываются заполненные буферы, принадлежащие только соответствующим файлам данных, вне зависимости от порядка SCN.

Алгоритм LRU хранит часто используемые блоки в буферном кэше для минимизации количества операций доступа к диску. Для продления срока хранения блока в памяти в таблицах можно установить параметр CACHE.

Параметр инициализации DB_WRITER_PROCESSES задает количество процессов DBWn. Максимальное количество процессов DBWn равно 36. Если данный параметр не указан пользователем при запуске, Oracle Database определяет значение параметра DB_WRITER_PROCESSES в зависимости от количества центральных процессоров и групп процессоров.

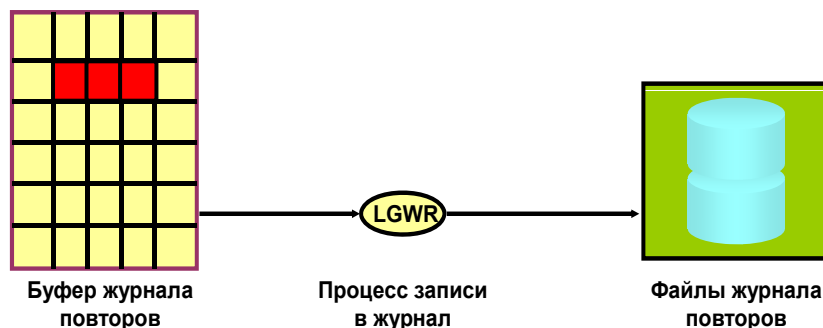
Процесс DBWn записывает заполненные буферы на диск при таких условиях:

- Когда серверному процессу после сканирования порогового количества буферов не удастся найти свободный многократно используемый буфер, он подает процессу DBWn сигнал начала записи. Процесс DBWn записывает заполненные буферы на диск в асинхронном режиме во время выполнения других вычислений.
- Процесс DBWn записывает буферы, чтобы передвинуть контрольную точку, которая является позицией в потоке (журнале) повторов и с которой начинается восстановление экземпляра. Эта позиция определяется наиболее старым заполненным буфером в буферном кэше.

Во всех случаях процесс DBWn осуществляет пакетную (многоблочную) запись, что повышает продуктивность. Количество блоков, записываемых при многоблочной записи, зависит от используемой операционной системы.

Процесс записи в журнал (LGWR)

- Записывает буфер журнала повторов в файл журнала повторов на диске
- Запись осуществляется:
 - когда пользовательский процесс фиксирует транзакцию;
 - когда буфер журнала повторов заполняется на одну треть;
 - перед тем как процесс DBWn записывает измененные буферы на диск;
 - каждые 3 секунды.



ORACLE

© Oracle, 2009. Все права защищены.

Процесс записи в журнал (LGWR)

Процесс записи в журнал (LGWR) отвечает за управление буферами журналов повторов, записывая элементы буферов журнала повторов на диск в файл журнала повторов. Процесс LGWR записывает все записи повторов, скопированные в буфер с момента последней записи на диск.

Буфер журнала повторов является циклическим. После того как процесс LGWR сохранит записи повторов из буфера в файл журнала повторов, серверные процессы смогут копировать новые записи на место записей в буфере журнала повторов, которые уже записаны на диск. Обычно LGWR выполняет запись достаточно быстро, чтобы обеспечить в буфере свободное место для новых записей даже при интенсивном обращении к журналу повторов. Процесс LGWR записывает на диск одну непрерывную часть буфера.

Процесс LGWR выполняет запись:

- когда пользовательский процесс фиксирует транзакцию;
- когда буфер журнала повторов заполняется на одну треть;
- перед тем как процесс DBWn записывает измененные буферы на диск (при необходимости);
- каждые три секунды;

Процесс записи в журнал (LGWR) (продолжение)

Прежде чем процесс DBWn начнет запись измененного буфера, на диск должны быть сохранены все записи журнала повторов, связанные с изменениями буфера (протокол упреждающей записи). Если процесс DBWn обнаруживает, что некоторые записи журнала повторов не сохранены, он сигнализирует процессу LGWR, чтобы тот сохранил эти записи на диск, и ждет завершения записи буфера журнала повторов, прежде чем приступить к записи буферов данных. Процесс LGWR выполняет запись в текущую группу журналов. Если один из файлов в группе поврежден или недоступен, процесс LGWR продолжает запись в остальные файлы группы и записывает сообщение об ошибке в файл трассировки LGWR и системный журнал предупреждений. Если в группе повреждены все файлы или группа недоступна, так как не была заархивирована, процесс LGWR не сможет продолжить работу.

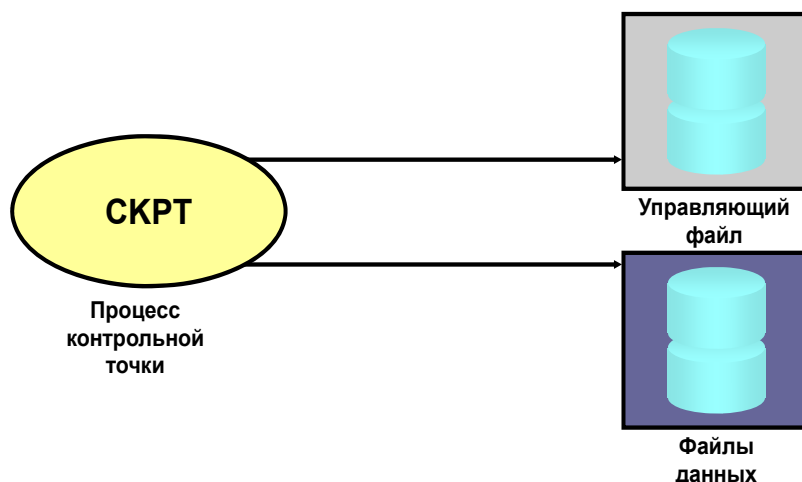
Когда пользователь выдает оператор **COMMIT**, процесс LGWR помещает запись фиксации транзакции в буфер журнала повторов. После этого LGWR немедленно записывает на диск буфер журнала повторов и записи повторов из транзакции. Внесение соответствующих изменений в блоки данных задерживаются до момента наиболее эффективной записи. Это называется механизмом *быстрой фиксации транзакции*. Простейшая операция записи повтора, которая содержит запись о фиксации транзакции, является единственным событием, определяющим, была ли зафиксирована транзакция. Oracle Database возвращает код успешной фиксации транзакции, хотя буферы данных еще не записаны на диск.

Если необходимо дополнительное пространство в буфере, процесс LGWR иногда сохраняет записи журнала повторов перед фиксацией транзакции. Эти записи становятся постоянными только в том случае, если транзакция впоследствии фиксируется. Когда пользователь фиксирует транзакцию, ей присваивается системный номер изменения (SCN), который записывается базой данных Oracle в журнал повторов вместе с записями повторов транзакций. Номера SCN записываются в журнал повторов для синхронизации операций восстановления в кластерах Real Application Clusters и распределенных базах данных.

В моменты интенсивной нагрузки процесс LGWR может записывать файл журнала повторов при помощи групповой фиксации транзакций. Предположим, что пользователь фиксирует транзакцию. Процесс LGWR должен записать на диск записи повторов из этой транзакции. В это время другие какие-либо пользователи выдают оператор **COMMIT**. Однако процесс LGWR не может выполнить запись в файл журнала повторов, чтобы зафиксировать эти транзакции, пока не закончит предыдущую операцию записи. После сохранения в файл журнала повторов записей из первой транзакции весь список записей повторов из ожидающих транзакций (еще не зафиксированных) может быть записан на диск одной операцией. Это потребует меньшего количества операций ввода/вывода, чем отдельная обработка каждой записи транзакции. Следовательно, Oracle Database сводит к минимуму количество операций дискового ввода/вывода и максимально повышает производительность процесса LGWR. Если высокая интенсивность поступления запросов фиксации не снижается, каждая запись (выполняемая процессом LGWR) из буфера журнала повторов может содержать по несколько записей фиксации транзакций.

Процесс создания контрольной точки (СКРТ)

- Записывает информацию о контрольной точке в следующие объекты:
 - управляющий файл;
 - заголовок каждого файла данных.



ORACLE

© Oracle, 2009. Все права защищены.

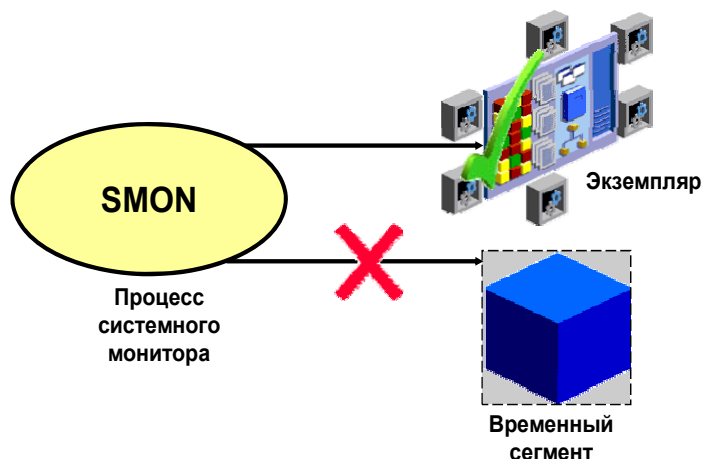
Процесс создания контрольной точки (СКРТ)

Контрольная точка – это структура данных, определяющая системный номер изменения (SCN) в потоке повторов базы данных. Контрольные точки записываются в управляющий файл и в заголовок каждого из файлов данных. Они являются важнейшим элементом процесса восстановления.

При фиксации контрольной точки Oracle Database нужно обновить заголовки всех файлов данных, чтобы записать сведения о контрольной точке. Эту операцию выполняет процесс СКРТ. Процесс СКРТ не записывает блоки на диск, эту операцию выполняет процесс DBWn. Запись номеров SCN в заголовки файлов гарантирует, что все изменения, внесенные в блоки базы данных до фиксации данного номера SCN, уже записаны на диск.

Процесс системного монитора (SMON)

- Выполняет восстановление при запуске экземпляра
- Очищает неиспользуемые временные сегменты



ORACLE

© Oracle, 2009. Все права защищены.

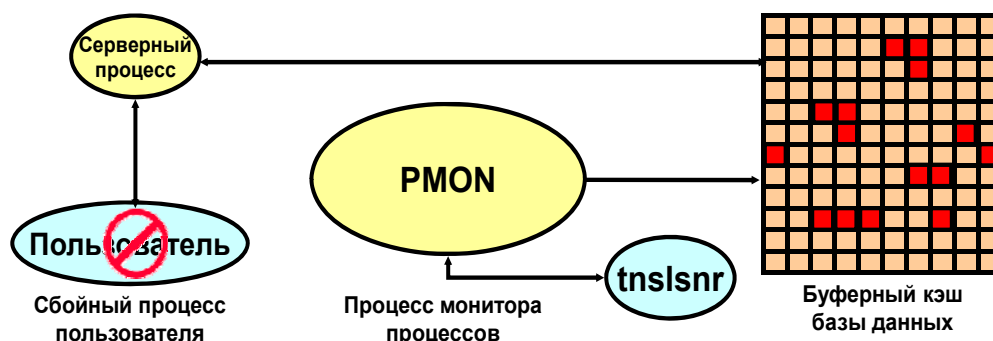
Процесс системного монитора (SMON)

Процесс системного монитора (SMON) в случае необходимости выполняет восстановление при запуске экземпляра. Процесс SMON также выполняет очистку временных сегментов, которые больше не используются. Если во время восстановления экземпляра какая-либо из завершенных транзакций была пропущена из-за ошибки чтения файла или ошибки автономного режима, процесс SMON восстановит их при переводе табличного пространства или файла обратно в оперативный режим.

Процесс SMON регулярно проверяет необходимость выполнения этого процесса. Процесс SMON также может быть вызван другими процессами при необходимости.

Процесс монитора процессов (PMON)

- Выполняет восстановление пользовательского процесса при сбое
 - Выполняет очистку буферного кэша базы данных
 - Освобождает ресурсы, использовавшиеся пользовательским процессом
- Выполняет мониторинг времени ожидания при отсутствии действий в сеансах
- Динамически регистрирует службы базы данных в прослушивателях



ORACLE

© Oracle, 2009. Все права защищены.

Процесс монитора процессов (PMON)

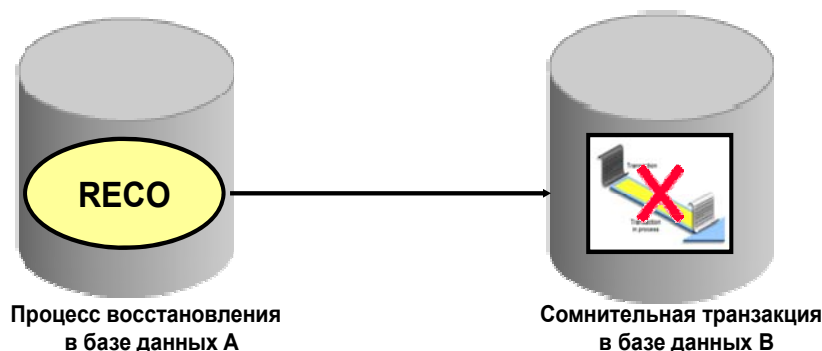
Процесс монитора процессов (PMON) выполняет восстановление пользовательского процесса в случае сбоя. Процесс PMON осуществляет очистку буферного кэша базы данных и освобождает ресурсы, использовавшиеся данным пользовательским процессом. Например, он сбрасывает состояние таблицы активных транзакций, снимает блокировки и удаляет идентификатор процесса из списка активных процессов.

Процесс PMON периодически проверяет состояние процессов диспетчера и сервера. Он перезапускает любой из этих процессов в случае его остановки (только если этот процесс не был завершен Oracle Database намеренно). Кроме того, процесс PMON регистрирует информацию об экземпляре и процессе диспетчера в сетевом прослушивателе.

Как и SMON, процесс PMON регулярно проверяет необходимость своего выполнения; при необходимости его может вызвать другой процесс.

Процесс восстановления

- Используется при распределенной конфигурации баз данных
- Автоматически подключается в другим базам данных, задействованным в сомнительных распределенных транзакциях
- автоматически разрешает все сомнительные транзакции
- Удаляет все строки, соответствующие сомнительным транзакциям



ORACLE

© Oracle, 2009. Все права защищены.

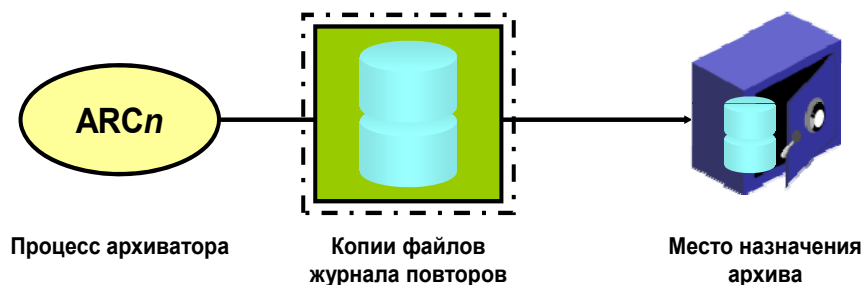
Процесс восстановления (RECO)

Процесс восстановления (RECO) представляет собой фоновый процесс, который используется при распределенной конфигурации базы данных и служит для автоматического разрешения сбоев, вызванных распределенными транзакциями. Процесс RECO экземпляра автоматически подключается к другим БД, задействованным в сомнительных распределенных транзакциях. Когда процесс RECO вновь устанавливает подключение между серверами БД, он автоматически разрешает все сомнительные транзакции, удаляя в каждой БД из таблицы ожидающих транзакций любые строки, связанные с разрешенными сомнительными транзакциями.

Если процессу RECO не удастся подключиться к удаленному серверу, RECO автоматически повторяет попытку подключения по истечении определенного интервала времени. При этом интервал времени между попытками подключения возрастает экспоненциально.

Процессы архивации (ARCn)

- Копируют файлы журнала повторов на указанное устройство хранения после переключения журнала
- Могут собирать данные для восстановления транзакций и передавать их резервным адресатам



ORACLE

© Oracle, 2009. Все права защищены.

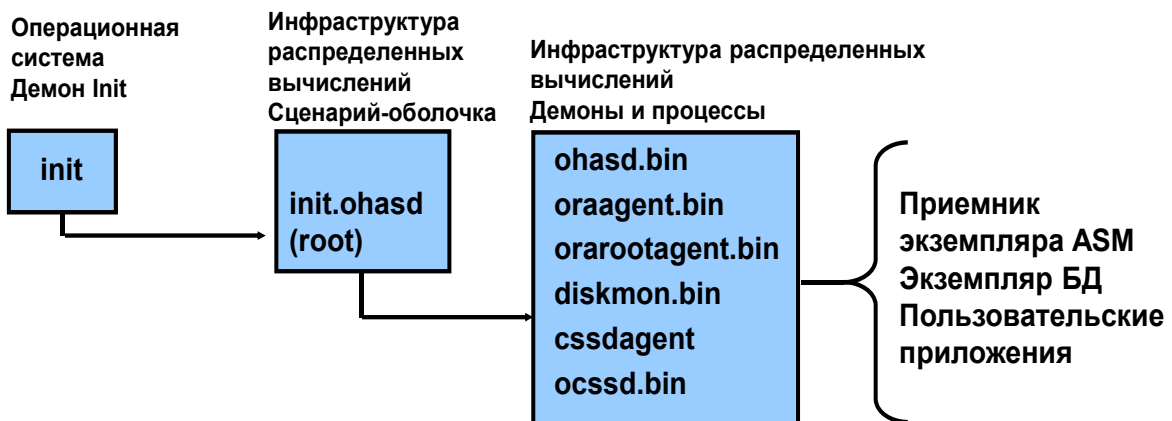
Процессы архивации (ARCn)

Процессы архивации (ARCn) копируют файлы журнала повторов на указанное устройство хранения после переключения журнала. Процессы ARCn функционируют только в том случае, если БД работает в режиме ARCHIVELOG, и при этом включено автоматическое архивирование.

Если ожидается, что архивирование вызовет интенсивную загрузку (например, при пакетной загрузке данных), можно увеличить максимальное количество процессов архиваторов. Возможно также наличие нескольких мест назначения архива. Рекомендуется установить как минимум один процесс архивации для каждого места назначения архива. По умолчанию существует четыре процесса архивации.

Последовательность запуска процессов

- Инфраструктура Oracle Grid Infrastructure запускается системным демоном init.



- При установке Oracle Grid Infrastructure изменяется файл /etc/inittab, что позволяет запускать инфраструктуру при запуске компьютера на соответствующем уровне.

```
# cat /etc/inittab
..
h1:35:respawn:/etc/init.d/init.ohasd run >/dev/null 2>&1 </dev/null
```

ORACLE

© Oracle, 2009. Все права защищены.

Последовательность запуска процессов

При установке инфраструктуры Oracle Grid Infrastructure в системном файле /etc/inittab размещаются записи для запуска сценария-оболочки. Сценарий-оболочка настраивает переменные среды, а затем запускает демоны и процессы Oracle Grid Infrastructure.

При остановке Oracle Grid Infrastructure с помощью специальной команды демоны также будут остановлены, но выполнение процесса сценария-оболочки продолжится.

Файл UNIX /etc/inittab имеет следующий формат:

идентификатор : уровни выполнения : действие : процесс с параметрами

Сценарий-оболочки запускается с действием respawn, поэтому при его прерывании он будет перезапущен.

После запуска некоторые демоны инфраструктуры Oracle Grid Infrastructure будут выполняться от имени пользователя root с приоритетом реального времени, а другие — от имени владельца инфраструктуры Grid Infrastructure с приоритетом режима пользователя. На платформе Windows вместо сценариев-оболочек для инициализации используются службы операционной системы, а демоны представляют собой исполняемые двоичные файлы.

Примечание. Прямое выполнение сценария оболочки не поддерживается.

Архитектура хранения БД



ORACLE

© Oracle, 2009. Все права защищены.

Архитектура хранения БД

Файлы, из которых состоит база данных Oracle, делятся на следующие категории:

- **Управляющие файлы:** содержат данные о самой базе данных (то есть информацию о физической структуре базы данных). Это критически важные файлы для базы данных. Без них невозможно открыть файлы данных и получить доступ к БД. Они также могут содержать метаданные резервных копий.
- **Файлы данных:** содержат данные пользователя или приложения базы данных, а также метаданные и словарь данных.
- **Оперативные файлы журнала повторов:** используются для восстановления экземпляра БД. Если происходит сбой сервера БД, при котором не теряются файлы данных, экземпляр позволит восстановить базу данных с помощью информации, содержащейся в этих файлах.

Для надежного функционирования БД также важны следующие файлы:

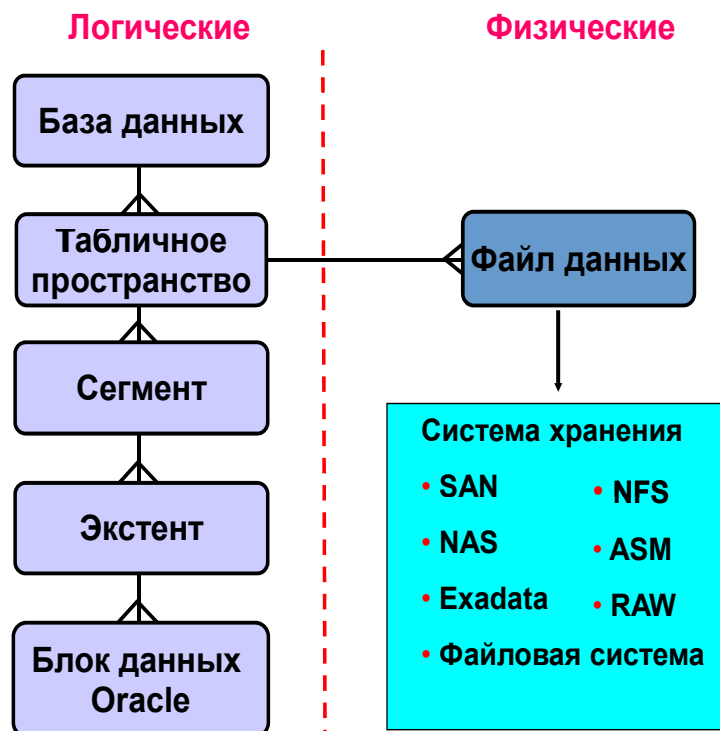
- **Файл параметров:** используется для определения конфигурации экземпляра для запуска.
- **Файл паролей:** позволяет пользователям с ролями sysdba, sysoper и sysasm удаленным образом подключаться к экземпляру и выполнять административные задачи.
- **Резервные копии файлов:** используются для восстановления БД. Обычно при сбое носителя или ошибке пользователя, вследствие которых был поврежден или удален исходный файл, для его восстановления используется резервная копия.
- **Архивные файлы журнала повторов:** содержат непрерывную историю изменений данных (повторных операций), которую создает экземпляр. С помощью этих файлов и резервной копии БД можно восстановить утраченные файлы данных. Таким образом, архивные журналы позволяют восстанавливать файлы данных.

Архитектура хранения БД (продолжение)

- **Файлы трассировки:** каждый сервер и фоновый процесс может создавать записи в связанном с ним файле трассировки. При обнаружении процессом внутренней ошибки он записывает дамп информации об ошибке в свой файл трассировки. Часть информации в файле трассировки предназначена для администратора БД, другая информация – для службы поддержки Oracle.
- **Файлы журнала предупреждений:** представляют собой особые записи трассировки. Журнал предупреждений БД – это хронологический журнал сообщений и ошибок. Oracle рекомендует периодически просматривать этот журнал.

Примечание. Файлы параметров, файлы паролей, файлы предупреждений и файлы трассировки рассматриваются в других уроках.

Логические и физические структуры БД



ORACLE

© Oracle, 2009. Все права защищены.

Логические и физические структуры БД

База данных состоит как из физических, так и из логических структур.

Базы данных, табличные пространства и файлы данных

На слайде показано, как связаны между собой базы данных, табличные пространства и файлы данных. Каждая база данных логически состоит из двух или более табличных пространств. Для каждого табличного пространства явным образом создается один или несколько файлов данных. В них физически хранятся данные всех сегментов табличного пространства. Если это пространство **TEMPORARY**, то файл данных заменяется временным файлом. Файл данных табличного пространства может физически храниться с помощью любой поддерживаемой технологии.

Табличные пространства

База данных делится на логические блоки хранения, которые называются *табличными пространствами*. Они объединяют связанные между собой логические структуры или файлы данных. Например, табличные пространства часто объединяют все объекты приложения, чтобы облегчить определенные операции администрирования.

Блоки данных

На низшем уровне логической структуры находятся *блоки данных*, в которых хранятся данные БД Oracle. Одному блоку данных соответствует определенное количество байтов физического пространства на диске. Размер блока данных задается для каждого табличного пространства при его создании. База данных использует и выделяет свободное пространство БД в блоках данных Oracle.

Логические и физические структуры БД (продолжение)

Экстенты

На следующем уровне логической структуры пространства БД находятся *экстенты*. Экстент представляет собой определенное количество непрерывных блоков данных Oracle, полученных при одном размещении. В этих блоках хранится информация определенного типа. Блоки данных Oracle в экстенте являются логически последовательными, но могут быть физически распределены по диску в связи с разнесением RAID и особенностями файловых систем.

Сегменты

Далее над экстентами в логической структуре хранения БД находятся *сегменты*. Сегмент – это набор экстентов, выделенных для конкретной логической структуры. Пример:

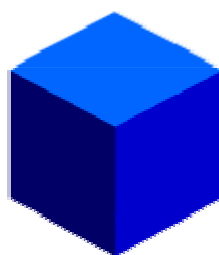
- **Сегменты данных:** любая некластеризованная неиндексная таблица использует сегмент данных. Исключение составляют внешние таблицы, глобальные временные таблицы и секционированные таблицы, использующие один или несколько сегментов. Все данные таблицы хранятся в экстентах данного сегмента данных. В секционированной таблице каждая секция использует свой сегмент данных. Каждый кластер использует сегмент данных. Данные всех таблиц в кластере хранятся в сегменте данных кластера.
- **Индексные сегменты:** любой индекс использует индексный сегмент, в котором хранятся все данные индекса. В секционированном индексе каждая секция использует свой индексный сегмент.
- **Сегменты отмены:** для каждого экземпляра БД создается одно табличное пространство отмены транзакций (UNDO). Это табличное пространство содержит множество сегментов отмены, в которых временно сохраняется информация об отмене операций. Информация, хранящаяся в сегментах отмены, используется для создания целостной для чтения информации БД, а также для отката незафиксированных пользовательских транзакций при восстановлении БД.
- **Временные сегменты:** временные сегменты создаются базой данных Oracle, когда для выполнения оператора SQL необходима временная рабочая область. После завершения выполнения оператора экстенты временного сегмента возвращаются в экземпляр и могут использоваться далее. Можно указать временное табличное пространство по умолчанию для каждого из пользователей или же для всей БД.

Примечание. Существуют другие типы сегментов, не указанные выше. Также существуют объекты схемы (например, представления, пакеты, триггеры и т.д.), которые не считаются сегментами несмотря на то, что являются объектами базы данных. Сегмент является владельцем выделенного ему дискового пространства. Другие объекты существуют в виде строк, хранящихся в сегменте метаданных системы.

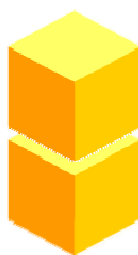
Сервер баз данных Oracle размещает пространство динамически. Если существующие экстенты в сегменте оказываются заполнены, в него добавляются новые. Так как экстенты размещаются по необходимости, экстенты сегмента необязательно располагаются на диске последовательно, а также могут быть получены из различных файлов данных одного и того же табличного пространства.

Сегменты, экстененты и блоки

- Сегменты существуют в табличном пространстве.
- Сегмент – это набор экстенентов.
- Экстенент – это набор блоков данных.
- Блоки данных связаны с дисковыми блоками.



Сегмент



Экстененты



Блоки
данных



Дисковые блоки
(хранение
в файловой
системе)

ORACLE

© Oracle, 2009. Все права защищены.

Сегменты, экстененты и блоки

Объекты базы данных – таблицы и индексы – хранятся в табличных пространствах в виде сегментов. Каждый сегмент содержит один или несколько экстенентов. Экстенент состоит из непрерывного участка блоков данных, соответственно, один экстенент может находиться только в одном файле данных. Блок данных – это наименьший элемент операции ввода-вывода в базе данных.

Когда БД запрашивает у операционной системы (ОС) набор блоков данных, ОС связывает его с текущей файловой системой или дисковыми блоками устройства хранения. Поэтому нет необходимости знать физический адрес данных, хранящихся в БД. Также это значит, что файл данных можно разнести или зеркально дублировать на несколько дисков.

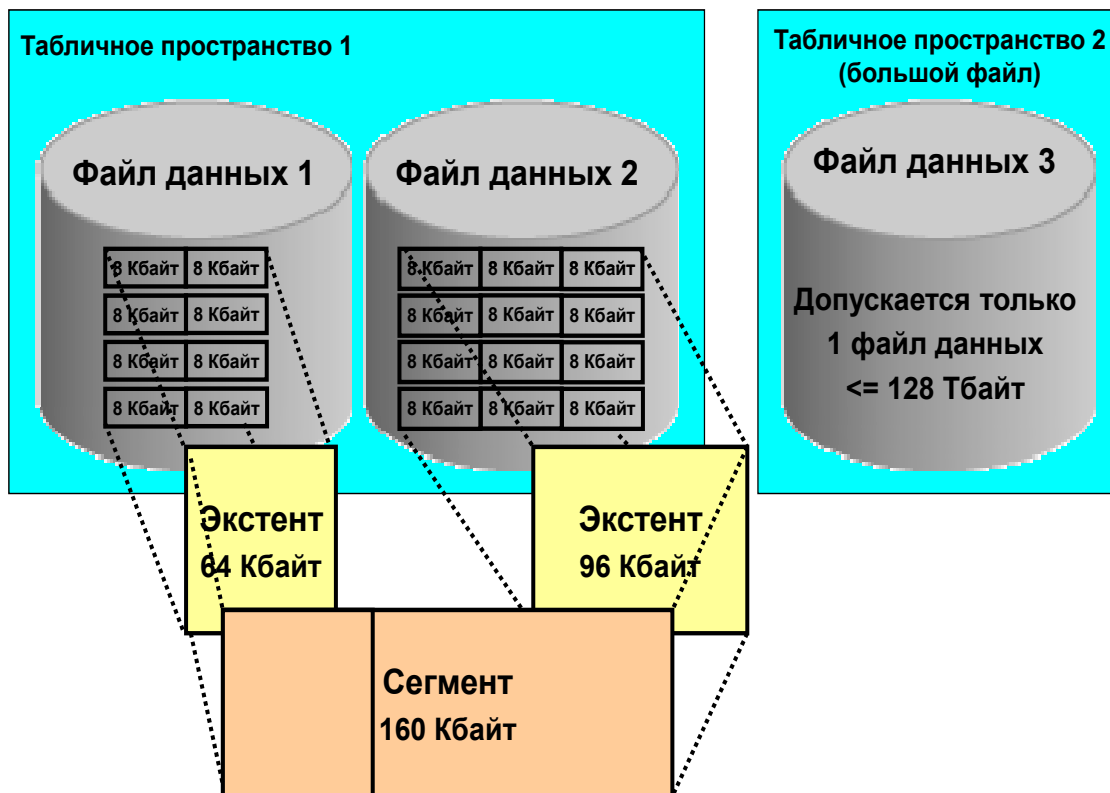
Размер блока данных можно задать при создании базы данных. Стандартный размер в 8 килобайт подойдет для большинства БД. Если ваша база данных поддерживает приложение хранилища данных с очень большими таблицами и индексами, может оказаться целесообразным выбрать больший размер блока.

Если ваша БД поддерживает приложение для обработки транзакций, использующее произвольное чтение и запись, разумно задать меньший размер блока данных.

Максимальный размер блока данных зависит от используемой ОС. Минимальный размер блока данных Oracle составляет 2 килобайта. Блоки такого размера используются крайне редко (практически не используются).

Можно также использовать табличное пространство с блоками данных нестандартного размера. Подробные сведения см. в руководстве *Oracle Database Administrator's Guide*.

Табличные пространства и файлы данных



ORACLE

© Oracle, 2009. Все права защищены.

Табличные пространства и файлы данных

База данных делится на *табличные пространства*, являющиеся логическими блоками хранения. Их можно использовать для объединения связанных между собой логических структур. Каждая база данных логически состоит из двух или более табличных пространств: SYSTEM и SYSAUX. Для каждого табличного пространства явным образом создается один или несколько файлов данных. В них физически хранятся данные всех логических структур табличного пространства.

На рисунке на этом слайде показано табличное пространство, состоящее из двух файлов данных. Сегмент размером 160 Кбайт охватывает два файла данных. Он состоит из двух экстенгов. Первый экстент размером 64 Кбайт находится в первом файле данных, а второй экстент размером 96 Кбайт находится во втором файле данных. Оба экстенга формируются из последовательных блоков Oracle размером по 8 Кбайт.

Примечание. Можно также создавать табличные пространства типа BIGFILE, использующие один большой файл. Его размер ограничивается только используемой архитектурой идентификатора строки. Максимальный размер такого файла равен произведению размера блока в табличном пространстве на 2^{36} , или 128 терабайт при размере блока 32 килобайта. Традиционные табличные пространства типа SMALLFILE (создаваемые по умолчанию) могут содержать несколько файлов данных меньшего размера. Подробнее о табличных пространствах типа BIGFILE см. в руководстве *Oracle Database Administrator's Guide*.

Табличные пространства SYSTEM и SYSAUX

- Табличные пространства SYSTEM и SYSAUX – обязательные табличные пространства, создаваемые одновременно с базой данных. Они должны находиться в оперативном режиме.
- Табличное пространство SYSTEM используется для основных функций (например, для хранения таблиц словаря данных).
- Вспомогательное табличное пространство SYSAUX используется для дополнительных компонентов базы данных (например, для репозитория Enterprise Manager).
- Табличные пространства SYSTEM и SYSAUX не рекомендуется использовать для хранения данных приложений.

ORACLE

© Oracle, 2009. Все права защищены.

Табличные пространства SYSTEM и SYSAUX

Все базы данных Oracle должны иметь табличное пространство SYSTEM и табличное пространство SYSAUX. Они автоматически создаются при создании базы данных.

По умолчанию создаются табличные пространства типа SMALLFILE. Возможно также создание табличных пространств BIGFILE, которые позволяют управлять очень большими файлами в базе данных Oracle.

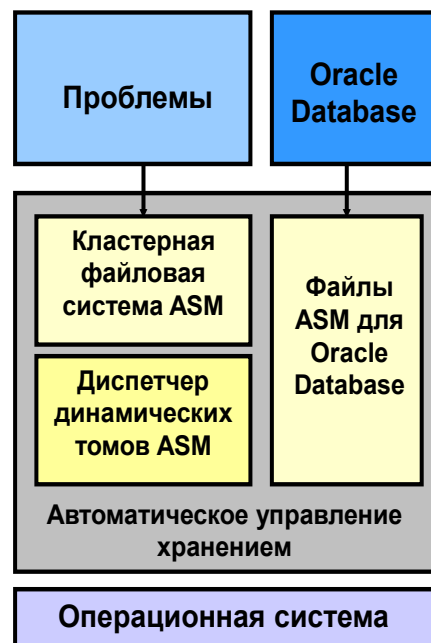
Табличное пространство может находиться в оперативном (доступном) и автономном (недоступном) режимах. Табличное пространство SYSTEM всегда находится в оперативном режиме, пока база данных открыта. В нем хранятся таблицы, обеспечивающие основные функции базы данных, например, таблицы словаря данных.

Табличное пространство SYSAUX является вспомогательным для пространства SYSTEM. В пространстве SYSAUX хранится множество компонентов базы данных, поэтому оно должно находиться в оперативном режиме, чтобы обеспечивать правильную работу всех компонентов базы данных. Табличные пространства SYSTEM и SYSAUX не рекомендуется использовать для хранения данных приложений. Для этих целей можно создать дополнительные пространства.

Примечание. Табличное пространство SYSAUX может быть переведено в автономный режим при восстановлении табличного пространства. Для табличного пространства SYSTEM это невозможно. Ни одно из этих табличных пространств нельзя сделать доступным только для чтения.

Автоматическое управление хранением

- Представляет собой переносимую и высокопроизводительную кластерную файловую систему управляет файлами базы данных Oracle
- Управляет файлами приложения с помощью кластерной файловой системы ASM (ACFS)
- Распределяет данные между дисками для балансирования нагрузки
- Зеркально дублирует данные на случай сбоев
- Решает проблемы, возникающие при управлении хранилищем



ORACLE

© Oracle, 2009. Все права защищены.

Автоматическое управление хранением

Система автоматического управления хранением (ASM) обеспечивает вертикальную интеграцию файловой системы и диспетчера томов для файлов базы данных Oracle. ASM может управлять отдельными компьютерами с симметричной многопроцессорной обработкой (SMP) или множеством узлов кластера с поддержкой технологии Oracle Real Application Clusters (RAC).

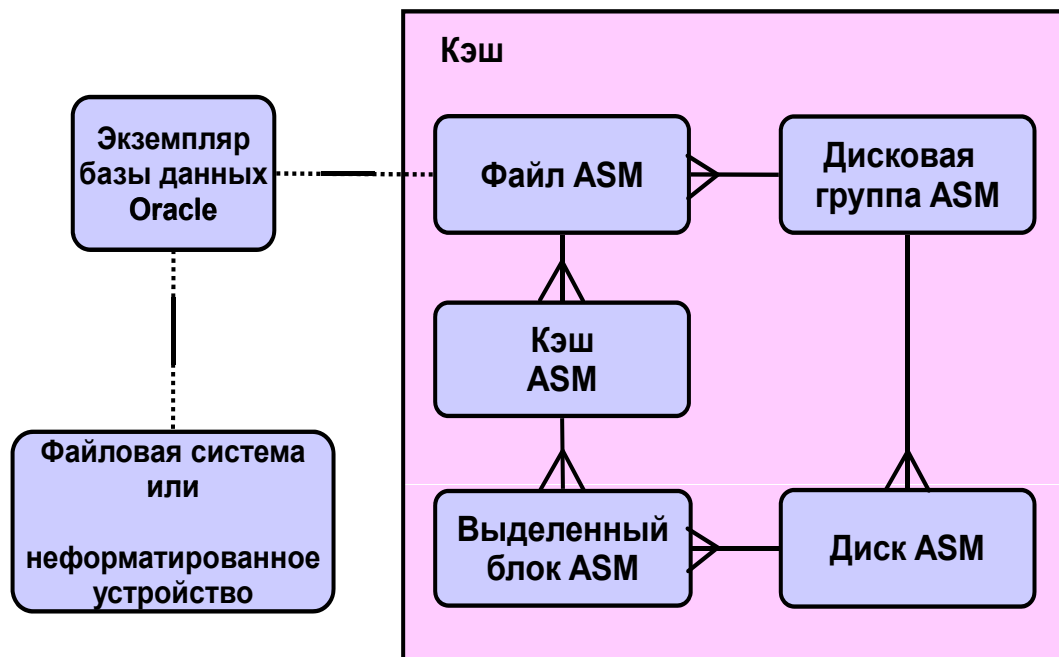
Кластерная файловая система Oracle ASM (ACFS) представляет собой многоплатформенную масштабируемую файловую систему и технологию управления хранением, которая расширяет функциональность ASM на файлы приложения вне базы данных Oracle (например, на исполняемые файлы, отчеты, файлы BFILE, видео, аудио, текст, изображения и другие файловые данные приложений).

ASM распределяет нагрузку от операций ввода/вывода (I/O) между всеми доступными ресурсами для оптимизации производительности и устраняет необходимость ручной настройки таких операций. ASM помогает администраторам управлять динамической средой базы данных, позволяя им увеличивать размер базы данных без необходимости останавливать ее для настройки параметров выделения пространства.

ASM может поддерживать избыточные копии данных для обеспечения защиты от сбоев, а также поддерживаемые производителем передовые технологии хранения. Управление данными осуществляется посредством выбора желаемых характеристик надежности и производительности для классов данных, при этом пользователю не нужно взаимодействовать с системой на уровне отдельных файлов.

Возможности ASM позволяют сэкономить время администратора благодаря автоматизации операций хранения, а также расширить возможности администратора по управлению большими базами данных с возросшей эффективностью.

Компоненты хранения ASM



ORACLE

© Oracle, 2009. Все права защищены.

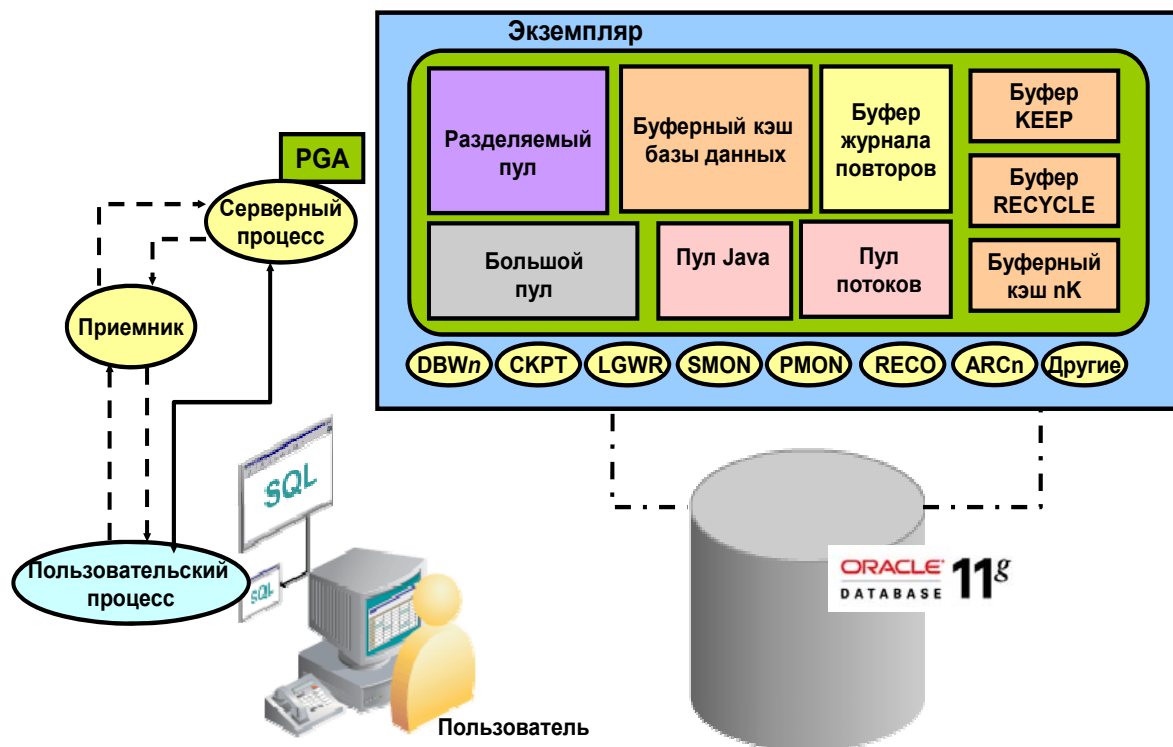
Компоненты хранения ASM

ASM не отключает существующую функциональность базы данных. Базы данных смогут функционировать как и прежде. Новые файлы можно создавать в виде файлов ASM, а существующие управляются прежним способом или переносятся в ASM.

На рисунке представлены связи между файлом данных базы Oracle и компонентами хранения ASM. Трехлучевая связь обозначает отношение "один к многим". Файл данных базы Oracle имеет отношение "один к одному" с файлом, хранящимся в файловой системе ОС, или с файлом ASM.

Группа дисков ASM Oracle представляет собой группу из одного или нескольких дисков ASM Oracle, которые являются одной логической единицей. Структуры данных в группе дисков автономны и используют некоторый объем пространства для метаданных. Диски ASM Oracle — это устройства хранения, предоставленные группе дисков ASM Oracle. Они могут быть физическими дисками, разделами, номерами логических блоков (LUN) из массива хранения, логическими томами (LV) или сетевыми файлами. Каждый диск ASM разделен на несколько выделенных блоков ASM, которые представляют собой минимальный объем пространства, выделяемый ASM. При создании группы дисков ASM размер блока ASM можно установить равным 1, 2, 4, 8, 16, 32 или 64 МБ в зависимости от уровня совместимости группы дисков. Один или несколько блоков ASM образуют экстенд ASM. Экстенд ASM Oracle представляет собой неформатированное пространство хранения, служащее для хранения содержимого файлов Oracle ASM. Файл Oracle ASM состоит из одного или нескольких экстендов файлов. Для поддержки очень больших файлов ASM используются различные размеры экстендов, равные размеру 1, 4 и 16 выделенных блоков.

Взаимодействие с базой данных Oracle: память, процессы и хранение



ORACLE

© Oracle, 2009. Все права защищены.

Взаимодействие с базой данных Oracle

В приведенном примере описываются операции базы данных Oracle на основном уровне. В примере иллюстрируется конфигурация базы данных Oracle, в которой пользовательский и связанный с ним серверный процессы выполняются на разных компьютерах, а подключение установлено с помощью сети.

1. Экземпляр, запущенный на узле, на котором установлена база данных Oracle, часто называется *хостом* или *сервером базы данных*.
2. Пользователь запускает приложение, порождающее пользовательский процесс. Приложение пытается установить соединение с сервером. Соединение может быть локальным, клиент-серверным или трехъярусным соединением из среднего яруса.
3. Сервер запускает приемник с соответствующим обработчиком Oracle Net Services. Приемник обнаруживает запрос от приложения на соединение и создает выделенный серверный процесс по требованию пользовательского процесса.
4. Пользователь выполняет SQL-оператор DML-типа и подтверждает транзакцию. Например, пользователь изменяет в таблице адрес заказчика и подтверждает изменения.
5. Серверный процесс получает оператор и проверяет разделяемый пул (компонент SGA) на наличие любой разделяемой области SQL, которая содержит идентичный SQL-оператор. Если разделяемая область SQL найдена, серверный процесс проверяет полномочия доступа пользователя к запрошенным данным, и оператор обрабатывается с помощью существующей разделяемой области SQL. Если же разделяемая область SQL не найдена, будет выделена новая разделяемая область SQL для синтаксического разбора и обработки оператора.

Взаимодействие с базой данных Oracle (продолжение)

6. Серверный процесс извлекает все необходимые значения данных из файла (таблицы) фактических данных или из значений, хранящихся в SGA.
7. Серверный процесс изменяет данные в SGA. Так как транзакция зафиксирована, процесс записи в журнал (LGWR) немедленно записывает транзакцию в файл журнала повторов. Процесс записи в базу данных (DBWn) окончательно записывает измененные блоки на диск, когда это является целесообразным.
8. Если транзакция была обработана успешно, процесс сервера посылает по сети соответствующее сообщение приложению. В противном случае будет передано сообщение об ошибке.
9. На протяжении всей процедуры выполняются другие фоновые процессы, отслеживая условия, которые требуют вмешательства. Кроме того, сервер базы данных управляет транзакциями других пользователей и предотвращает конкуренцию между транзакциями за одни и те же данные.

Экспресс-тест

Процесс монитора процессов (PMON):

1. выполняет восстановление при запуске экземпляра
2. выполняет восстановление пользовательского процесса при сбое
3. автоматически разрешает все сомнительные транзакции
4. записывает буфер журнала повторов в файл журнала повторов на диске

ORACLE

© Oracle, 2009. Все права защищены.

Ответ: 2

Экспресс-тест

Какие типы экземпляров обращаются к файлам ASM?

1. только экземпляры RDBMS
2. только экземпляры ASM
3. экземпляры RDBMS и ASM

ORACLE

© Oracle, 2009. Все права защищены.

Ответ: 3

Заключение

В ходе прохождения урока вы освоили следующие темы:

- список основных компонентов архитектуры базы данных Oracle;
- **структуры памяти;**
- фоновые процессы;
- разница между логическими и физическими структурами хранения;
- компоненты хранения ASM.

ORACLE

© Oracle, 2009. Все права защищены.

Упражнение 1: обзор

Это письменное упражнение с вопросами о следующих компонентах:

- архитектура базы данных;
- **память;**
- процессы;
- структуры файлов.

ORACLE

© Oracle, 2009. Все права защищены.

2

Установка программного обеспечения Oracle

ORACLE

© Oracle, 2009. Все права защищены.

Цели

Изучив материал этого занятия, вы освоите следующие темы:

- роль администратора базы данных (DBA), стандартные **задачи и средства**;
- планирование установки базы данных Oracle;
- установка инфраструктуры распределенных вычислений Oracle Grid Infrastructure для автономного сервера;
- установка программного обеспечения базы данных Oracle.

ORACLE

© Oracle, 2009. Все права защищены.

Задачи администратора БД Oracle

При разработке, реализации и обслуживании базы данных Oracle выполняются следующие задачи:

1. оценка аппаратных компонентов сервера БД;
2. установка программного обеспечения Oracle;
3. планирование стратегии базы данных и безопасности;
4. создание, перенос и открытие базы данных;
5. резервное копирование базы данных;
6. зачисление системных пользователей и планирование их доступа к сети Oracle;
7. реализация структуры базы данных;
8. восстановление базы данных после сбоя;
9. контроль производительности базы данных.



ORACLE

© Oracle, 2009. Все права защищены.

Задачи администратора БД Oracle

Обычно администратор баз данных (DBA) занимается установкой программного обеспечения Oracle и созданием базы данных.

Также DBA может отвечать за создание структур хранения базы данных, например, табличных пространств. Кроме того, он может создавать схемы или наборы объектов для хранения данных приложений.

Необходимо обеспечить доступ пользователей к базе данных. Для этого следует запустить базу данных, регулярно создавать ее резервные копии и контролировать ее производительность. Все эти задачи должны выполняться в рамках стратегии безопасности.

Проходя уроки данного курса, вы научитесь выполнять все описанные задачи. Более подробные сведения о задачах, показанных на слайде, содержатся в руководстве администратора *Oracle Database Administrator's Guide*.

В данном уроке мы подробно рассмотрим процесс установки. Задачу установки можно разделить на следующие подзадачи:

- рассмотрение установки с точки зрения общей технической архитектуры организации;
- пересмотр (и обновление) планов мощности;
- выбор ПО для управления базой данных (нужная версия и возможности);
- проверка соответствия требований к системе для всех выбранных элементов.

Средства администрирования базы данных Oracle

- Программа установки Oracle Universal Installer
- Помощник по конфигурированию Database Configuration Assistant
- **Мастер обновления Database Upgrade Assistant**
- Oracle Net Manager
- Oracle Net Configuration Assistant
- Oracle Enterprise Manager
- Служебная программа управления сервером Server Control Utility
- SQL*Plus
- Диспетчер восстановления
- Насос данных
- SQL*Loader

ORACLE

© Oracle, 2009. Все права защищены.

Средства администрирования базы данных Oracle

Для установки и обновления используются следующие средства:

- **Oracle Universal Installer (OUI):** служит для установки ПО Oracle и дополнительных компонентов. Может автоматически запускать помощник Database Configuration Assistant для создания базы данных.
- **Database Configuration Assistant (DBCA):** служит для создания базы данных при помощи готовых шаблонов Oracle. Это позволяет скопировать начальную предварительно настроенную базу данных (также можно создать собственную базу данных и шаблоны).
- **Database Upgrade Assistant (DBUA):** помогает выполнить обновление существующей базы данных до новой версии Oracle.
- **Oracle Net Manager (netmgr):** служит для настройки сетевых подключений к базам данных и приложениям Oracle.
- **Oracle Net Configuration Assistant (NetCA):** графический мастер для настройки и управления конфигурациями сети Oracle.

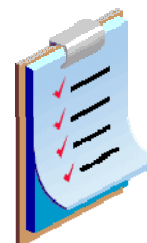
Средства администрирования базы данных Oracle (продолжение)

Для администрирования экземпляра Oracle используются следующие средства:

- **Oracle Enterprise Manager (ЕМ):** включает в себя графическую консоль, агенты, общие службы и средства, представляющие собой интегрированную всестороннюю платформу для управления продуктами Oracle. После установки ПО Oracle, создания или обновления базы данных и настройки сети вы можете использовать ЕМ как единый интерфейс для управления БД. Он предоставляет пользовательский Web-интерфейс для выполнения команд SQL, а также взаимодействует с другими компонентами Oracle для администрирования баз данных (такими как Recovery Manager и Scheduler).
- Основные средства ЕМ, которые используются для администрирования базы данных Oracle:
 - **консоль Enterprise Manager Database:** служит для администрирования одной БД;
 - **Enterprise Manager Grid Control:** служит для одновременного администрирования нескольких БД;
- **Server Control Utility (srvctl):** стандартный интерфейс командной строки, который может использоваться для запуска и остановки базы данных и экземпляров, управления экземплярами ASM, а также перемещения или удаления экземпляров и служб. Для добавления служб и управления сведениями о конфигурации также можно использовать SRVCTL.
- **SQL*Plus:** стандартный интерфейс командной строки для управления БД;
- **Диспетчер восстановления (RMAN):** средство Oracle, в полной мере удовлетворяющее потребностям в резервном копировании и восстановлении всей БД или же определенных ее файлов;
- **Насос данных:** служит для высокоскоростной передачи данных между базами данных (например, при экспортировании таблицы из одной БД и импортировании ее в другую БД);
- **SQL*Loader:** служит для загрузки данных из внешнего файла в базу данных Oracle. Это одна из утилит Oracle, с помощью которой можно загружать данные в таблицы БД;
- **Средства командной строки:**
 - для администрирования Enterprise Manager:
emctl start | status | stop dbconsole
 - для администрирования приемника:
lsnrctl start | status | stop

Планирование установки

- Какое ПО Oracle устанавливается?
- Соответствует ли аппаратное обеспечение минимальным требованиям?
- **Существует ли рекомендованный порядок установки при работе с несколькими продуктами?**
- Существуют ли предварительные действия, которые должны выполняться не администратором, а другими людьми?



ORACLE

© Oracle, 2009. Все права защищены.

Планирование установки

Перед установкой ПО Oracle необходимо задать себе следующие вопросы, позволяющие спланировать процесс установки.

- **Какое ПО Oracle устанавливается?**
В данном курсе предполагается, что устанавливается база данных Oracle. Корпорация Oracle рекомендует использовать метод автоматического управления хранением Automatic Storage Management (ASM). Процесс установки включает в себя установку инфраструктуры распределенных вычислений Oracle Grid Infrastructure с носителя кластерного ПО. При этом будут установлены компоненты, необходимые для ASM, а также Oracle Restart.
- **Соответствует ли аппаратное обеспечение минимальным требованиям?**
Определите все аппаратное обеспечение, которое будет задействовано в процессе установки и убедитесь в том, что минимальные рекомендованные требования соблюдены.
- **Существует ли рекомендованный порядок установки при работе с несколькими продуктами?**
Рекомендуется по возможности устанавливать инфраструктуру Oracle Grid Infrastructure перед ПО баз данных Oracle. Выполнение установки в этом порядке означает, что новая база данных будет настроена на использование групп дисков ASM и будет автоматически зарегистрирована в Oracle Restart. Если Oracle Grid Infrastructure устанавливается после базы данных Oracle, то для ее регистрации в Oracle Restart потребуются настройки вручную. Чтобы использовать в данной базе группы дисков ASM для хранения, необходимо выполнить миграцию.

Планирование установки (продолжение)

- **Существуют ли предварительные действия, которые должны выполняться не администратором, а другими людьми?**

В соответствии с рекомендованной стратегией разделения должностных обязанностей администратор скорее всего не отвечает за настройку аппаратного обеспечения и устройств хранения, которые используются при установке. Перед установкой Oracle Grid Infrastructure администратор хранения должен выполнить определенные действия по настройке необходимых разделов диска. Дополнительную информацию см. в руководстве Oracle Database Installation Guide.

Установка инфраструктуры Oracle Grid Infrastructure и базы данных Oracle: системные требования

- Требования к памяти:
 - 1 Гбайт для экземпляра базы данных с Oracle Enterprise Manager Database Control;
 - 1,5 Гбайт для экземпляра ASM и Oracle Restart.
- Требования к дисковому пространству:
 - 3 Гбайт для подкачки (при 2 Гбайт ОЗУ);
 - 1 Гбайт дискового пространства для каталога /tmp;
 - 3,8 Гбайт для ПО базы данных Oracle;
 - 4,5 Гбайт для ПО инфраструктуры Grid Infrastructure;
 - 1,7 Гбайт для предварительно настроенной БД (необязательно);
 - 2,4 Гбайт для области мгновенного восстановления (необязательно);
- Операционная система (см. документацию)



ORACLE

© Oracle, 2009. Все права защищены.

Установка инфраструктуры Oracle Grid Infrastructure и базы данных Oracle: системные требования

- Для обычной установки подходит компьютер, оснащенный 1 Гбайт ОЗУ, на котором используется файл подкачки размером не менее 1,5 Гбайт. Стандартная установка Oracle Grid Infrastructure для автономного сервера также требует минимум 1,5 Гбайт ОЗУ.
- Размер необходимого пространства для подкачки зависит от количества ОЗУ (например, для 2 Гбайт ОЗУ требуется 3 Гбайт пространства для подкачки). Подробные сведения см. в руководстве по установке в конкретной системе.
- В зависимости от интенсивности операций на компьютере, на который устанавливается ПО базы данных Oracle, обычная установка может длиться до 20 минут.
- Некоторые сведения об установке:
 - Oracle Database 11g поставляется с двумя шаблонами начальной БД.
 - Одинаковые файлы удаляются.
 - С дополнительных компакт-дисков можно установить множество других продуктов и демонстраций.

На слайде представлены минимальные аппаратные требования для всех платформ. В каждом конкретном случае при установке могут возникнуть дополнительные требования (это в частности касается дискового пространства).

Примечание. Установка типа Enterprise Edition, в которую входит стандартная начальная БД, называется "стандартной установкой".

Подготовка операционной системы

Создайте необходимых пользователей операционной системы и группы:

- Группы:
 - oinstall
 - dba
 - Необязательные группы (при разделении обязанностей):
 - oper
 - asmdba
 - asmoper
 - asmadmin
- Пользователи:
 - владелец ПО, обычно `oracle`;
 - можно создать несколько пользователей для установки нескольких продуктов.

ORACLE

© Oracle, 2009. Все права защищены.

Подготовка операционной системы

Системный администратор должен выполнить определенные действия с аппаратным обеспечением, которое используется для установки Oracle. Подробное рассмотрение всех настроек и команд операционной системы выходит за рамки данного курса. Каждая операционная система также имеет специальные требования для ПО Oracle. В этом документе будут описаны только основные требования. Более подробные сведения см. в документации по установке Oracle в конкретной системе.

Одним из необходимых действий является создание необходимых системных групп и пользователей. Требуется наличие двух групп операционной системы: `oinstall` и `dba`. При разделении обязанностей по нескольким учетным записям пользователей необходимо создать следующие группы: `oper`, `asmdba`, `asmoper` и `asmadmin`. Как минимум один системный пользователь должен выступать в роли владельца установки Oracle. В большинстве случаев для этого настраивается пользователь `oracle`. Для полного разделения обязанностей можно настроить владельцев для каждого из продуктов Oracle.

Настройка переменных среды

Переменные среды Oracle:

- **ORACLE_BASE**: основание структуры каталогов Oracle. Рекомендуется настроить параметр перед установкой.
- **ORACLE_HOME**: среда, в которой выполняются продукты Oracle. Настройка перед установкой не требуется, если настроена переменная **ORACLE_BASE**.
- **ORACLE_SID**: не требуется перед установкой, но полезна после нее для облегчения взаимодействия с определенным экземпляром.
- **NLS_LANG**: необязательная переменная среды, которая управляет языком, местностями и клиентским набором символов



ORACLE

© Oracle, 2009. Все права защищены.

Настройка переменных среды

Среда Oracle задействует множество переменных среды. Переменные, представленные на слайде, наиболее важны для успешной установки и использования базы данных Oracle. Их настройка не является обязательной, но их правильная настройка позволяет избежать проблем в будущем.

- **ORACLE_BASE**: указывает основание структуры каталогов Oracle для оптимальной гибкой инфраструктуры (OFA), рекомендованной службой поддержки Oracle. Использование необязательно, однако оно может упростить последующие установки и обновления. Это путь к каталогу, показанный в примере:
`/u01/app/oracle`
- **ORACLE_HOME**: среда, в которой выполняются продукты Oracle. Настройка перед установкой не требуется, если настроена переменная **ORACLE_BASE**. Средство установки OUI может использовать параметр **ORACLE_BASE** для определения рекомендованного значения **ORACLE_HOME**. Настройка этой переменной облегчает обслуживание и администрирование ПО Oracle. Это путь к каталогу, показанный в примере:
`/u01/app/oracle/product/11.2.0/dbhome_1`
- **ORACLE_SID**: системный идентификатор для экземпляра Oracle, например `orcl` для базы данных или `+ASM` для экземпляра ASM. Не требуется перед установкой, но полезен после нее для облегчения взаимодействия с определенным экземпляром.
- **NLS_LANG**: необязательная переменная среды, которая управляет языком, местностями и клиентским набором символов, как в следующем примере:
`AMERICAN_DENMARK.WE8MSWIN1252`

Подробнее о допустимых языках, местностях, наборах символов и языковой поддержке описано в руководстве Oracle Database Globalization Support Guide.

Проверка требований к системе

- Достаточное пространство для временных файлов
- 64-битные и
- 32-битные системы
- Надлежащая операционная система (ОС)
- Уровень исправлений ОС
- Системные пакеты
- Параметры системы и ядра
- Разрешения сервера X
- Достаточный размер файла подкачки
- Статус ORACLE_HOME

```
[oracle@edrsr12p1-+ASM Disk1]$ ./runInstaller
Starting Oracle Universal Installer...

Checking Temp space: must be greater than 80 MB.   Actual 15067 MB   Passed
Checking swap space: must be greater than 150 MB.   Actual 4000 MB   Passed
Checking monitor: must be configured to display at least 256 colors.   Actual 65536   Passed
Preparing to launch Oracle Universal Installer from /tmp/OraInstall2009-05-15_12-04-10AM. Please wait ...
```

ORACLE

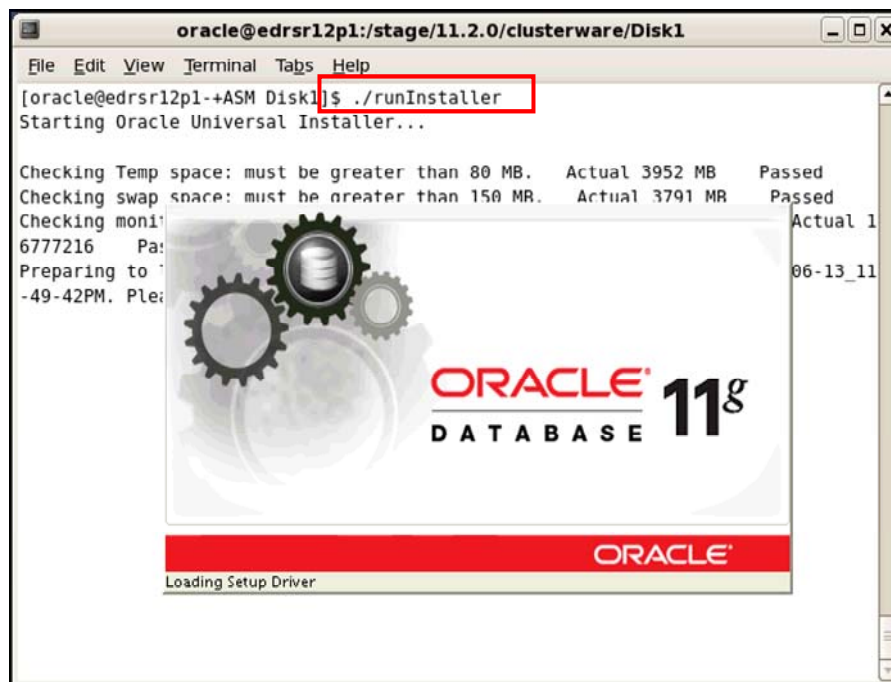
© Oracle, 2009. Все права защищены.

Проверка требований к системе

Средство установки Oracle Universal Installer автоматически выполняет проверку следующих требований:

- Наличие минимального необходимого размера временного пространства для установки и конфигурации. Эти требования проверяются во время установки.
- Установка 64-битной версии не может быть выполнена в домашний каталог Oracle с уже установленным 32-битным ПО (и наоборот).
- ПО Oracle Grid Infrastructure 11g и Oracle Database 11g сертифицировано для различных версий платформы Linux, а также для других платформ.
- Установлены все необходимые исправления ОС.
- Правильно настроены все необходимые системные параметры и параметры ядра.
- Переменная среды DISPLAY настроена, а пользователям предоставлены достаточные разрешения для отображения заданной переменной DISPLAY.
- Размер файла подкачки в системе достаточен для установки.
- Домашний каталог Oracle для новой установки пуст или принадлежит к группе поддерживаемых версий, поверх которых возможна установка Oracle Database 11g. При установке также проверяется, зарегистрированы ли эти версии в Oracle Inventory.

Oracle Universal Installer (OUI)



ORACLE

© Oracle, 2009. Все права защищены.

Oracle Universal Installer (OUI)

Oracle Universal Installer (OUI) – это Java-приложение, выполняющее установку на основе компонентов и позволяющее создавать пакетные и Web-установки разного уровня, равно как и объединять комплексную логику в едином пакете. Механизм установки легко переносится на любую платформу с поддержкой Java, что устраняет любые связанные с платформой проблемы при установке.

OUI обладает следующими возможностями по управлению и распространению ПО:

- автоматическое разрешение зависимостей и управление комплексной логикой;
- установка из Web;
- компонентная и пакетная установка;
- неявное удаление компонентов;
- поддержка нескольких домашних каталогов Oracle;
- поддержка NLS или глобализации;
- поддержка распределенной установки;
- автоматическая "бесшумная" установка с использованием файлов ответов.

Пример: сценарий установки

Сценарий установки, рассматриваемый в этом уроке, разделен на две части:

- **Часть 1.** Установка инфраструктуры распределенных вычислений Oracle Grid Infrastructure для автономного сервера
- **Часть 2.** Установка программного обеспечения базы данных Oracle

ORACLE

© Oracle, 2009. Все права защищены.

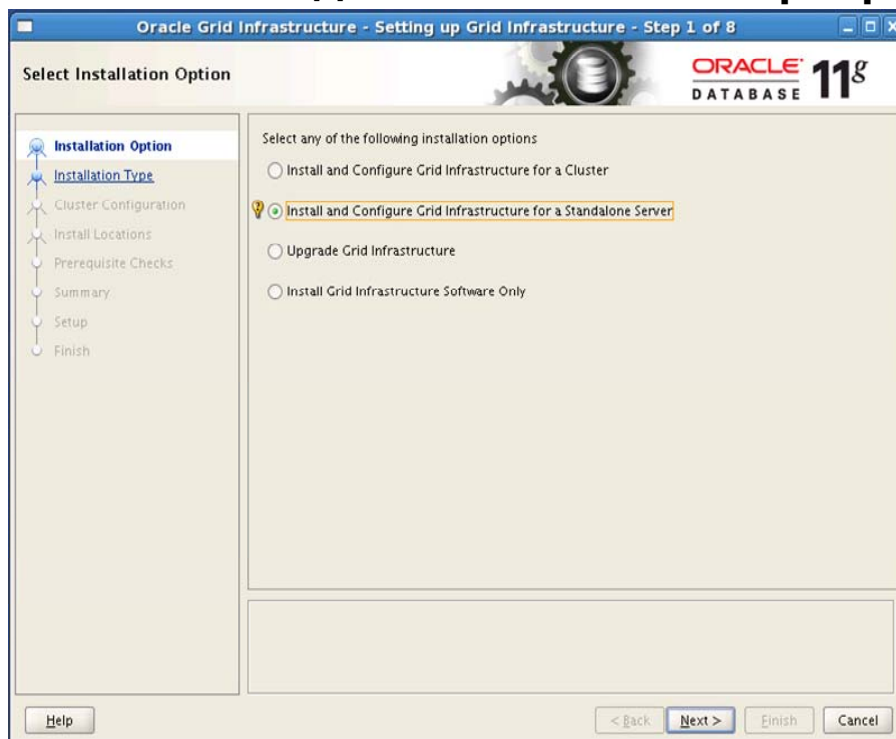
Пример: сценарий установки

Сценарий установки, рассматриваемый в этом уроке, разделен на две части:

- **Часть 1.** Установка инфраструктуры распределенных вычислений Oracle Grid Infrastructure для автономного сервера.
- **Часть 2.** Установка программного обеспечения базы данных Oracle.

В ходе установки инфраструктуры Oracle Grid Infrastructure описываются действия по настройке групп дисков ASM и Oracle Restart. Инфраструктура Oracle Grid Infrastructure устанавливается первой, чтобы база данных, созданная после установки ПО Oracle Database, могла использовать группы дисков ASM и была автоматически зарегистрирована в Oracle Restart.

Часть 1. Установка инфраструктуры распределенных вычислений Oracle Grid Infrastructure для автономного сервера



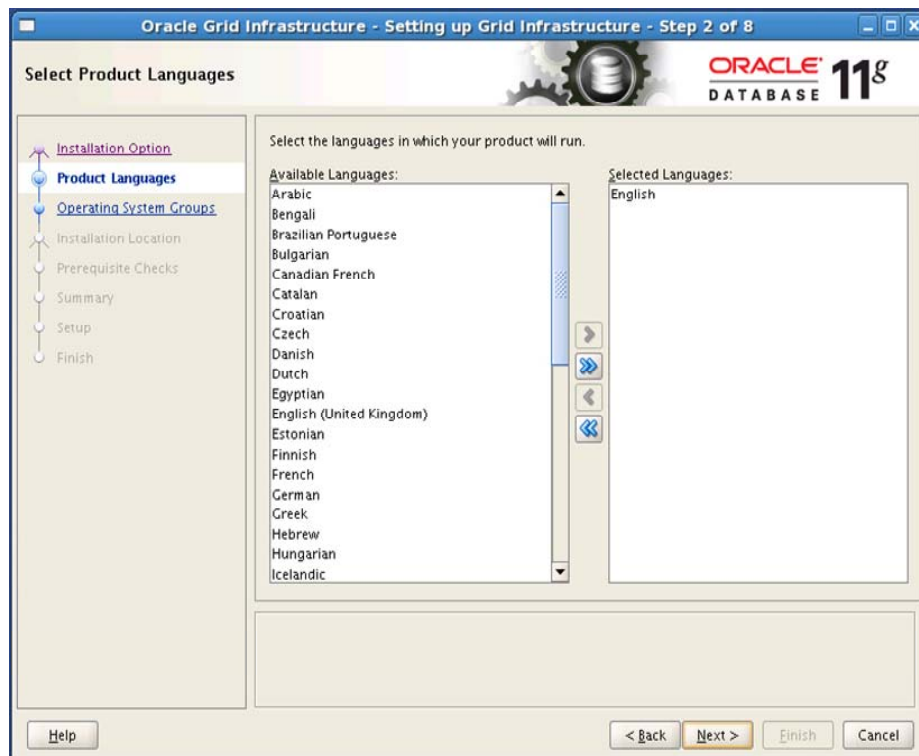
ORACLE

© Oracle, 2009. Все права защищены.

Часть 1. Установка инфраструктуры распределенных вычислений Oracle Grid Infrastructure для автономного сервера

Чтобы установить ПО Oracle Grid Infrastructure с помощью средства Oracle Universal Installer (OUI), войдите в систему компьютера как пользователь из группы администраторов с полномочиями установки ПО Oracle, создания баз данных и управления ими. Вставьте в привод для чтения компакт-диск с дистрибутивом или перейдите к каталогу с дистрибутивом Oracle Clusterware. Введите в этой папке `./runInstaller` для запуска средства OUI. Появится окно Installation Option ("Параметры установки") средства OUI. Выберите вариант Install and Configure Grid Infrastructure for a Standalone Server ("Установить и настроить Oracle Grid Infrastructure для автономного сервера") и нажмите кнопку Next ("Далее").

Выбор языков продукта



ORACLE

© Oracle, 2009. Все права защищены.

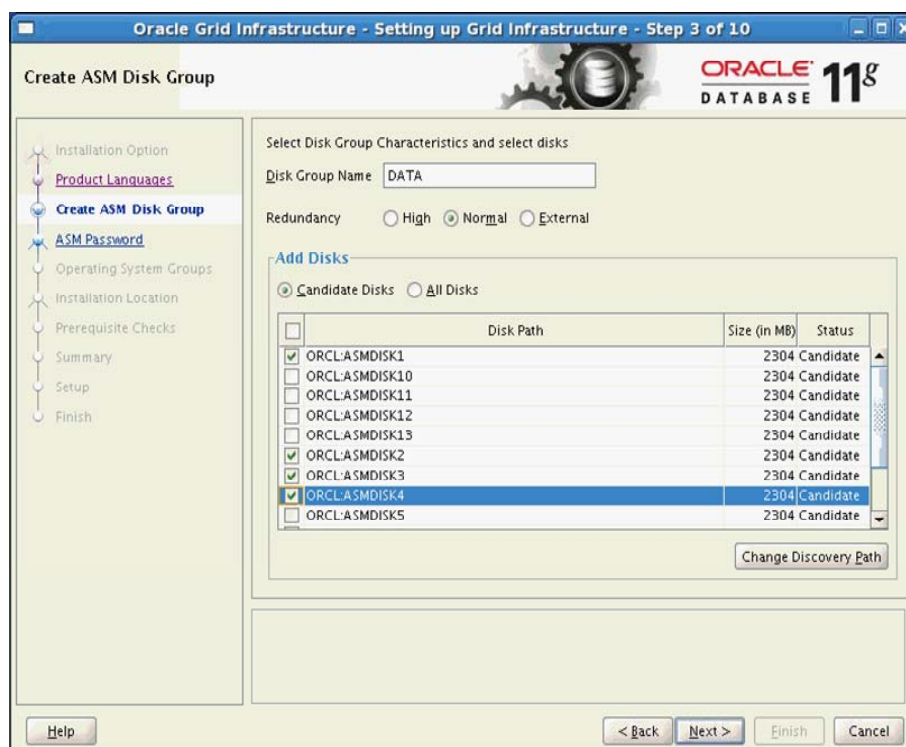
Выбор языков продукта

Отобразится страница Select Product Languages ("Выбор языков продукта"). Чтобы добавить языки в устанавливаемую программу, щелкните язык, а затем переместите его в список Selected Languages ("Выбранные языки") с помощью правой кнопки мыши. Чтобы выбрать несколько языков, удерживайте клавишу Ctrl и щелкните требуемые языки. Нажмите кнопку Next ("Далее"), чтобы продолжить установку.

В настоящее время доступны следующие языки: арабский, бенгальский, португальский (Бразилия), болгарский, французский (Канада), каталонский, хорватский, чешский, датский, голландский, арабский (Египет), английский (Великобритания), эстонский, финский, французский, немецкий, греческий, иврит, венгерский, исландский, индонезийский, итальянский, японский, корейский, испанский (Латинская Америка), латышский, литовский, малайский, испанский (Мексика), норвежский, польский, португальский, румынский, русский, китайский (упрощенный), словацкий, словенский, испанский, шведский, тайский, китайский (традиционный), турецкий, украинский и вьетнамский.

Примечание. Этот список может изменяться.

Создание группы дисков ASM



ORACLE

© Oracle, 2009. Все права защищены.

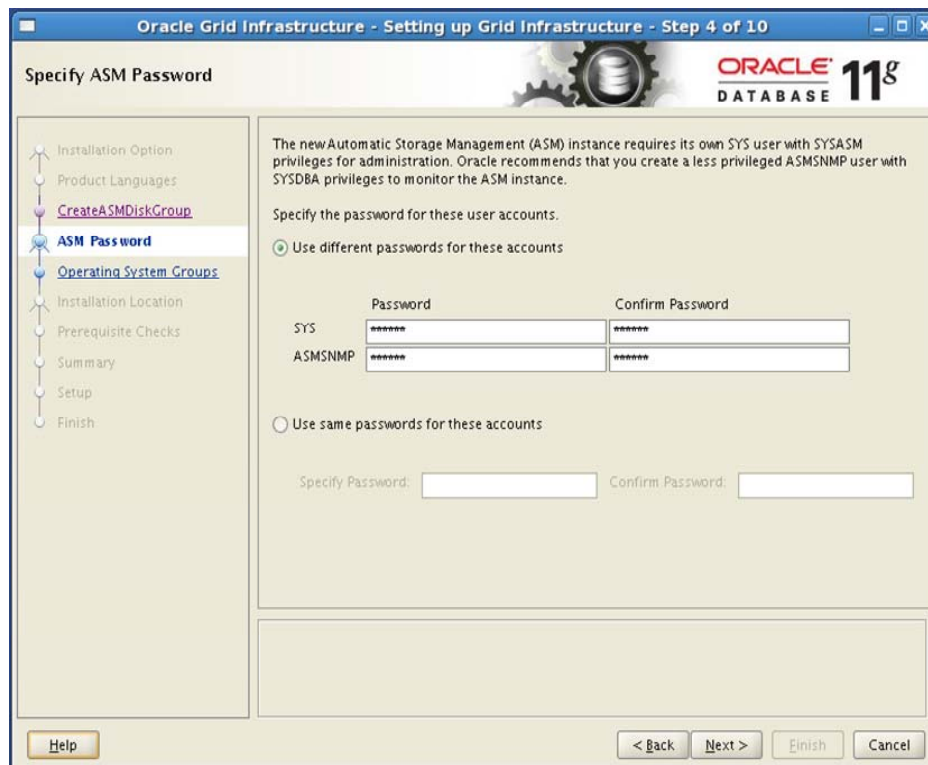
Создание группы дисков ASM

Открывается страница Create ASM Disk Group ("Создать группу дисков ASM"). Oracle Grid Infrastructure поддерживает ASM и Oracle Restart. Средство OUI продолжает работу только после создания группы дисков ASM. Введите имя первой группы дисков ASM в поле Disk Group Name ("Имя группы дисков"). Если раздел Add Disks ("Добавить диски") не заполнен возможными дисками, нажмите кнопку Change Discovery Path ("Изменить путь обнаружения") и введите путь обнаружения для дисков ASM.

Для большинства сред Oracle рекомендует создавать две группы дисков. Средство OUI в настоящее время позволяет создавать одну группу дисков ASM. Рекомендованную вторую группу дисков ASM можно создать после установки с помощью служебной программы ASM Configuration Assistant (asmca) или SQL*Plus. Нажмите кнопку Next ("Далее"), чтобы продолжить установку.

Примечание. Устройства появляются в списке примеров в связи с конфигурацией ASMLib. ASMLib предназначено только для платформ Linux. На других платформах может потребоваться нажатие кнопки Change Discovery Path ("Изменить путь обнаружения").

Установка паролей ASM



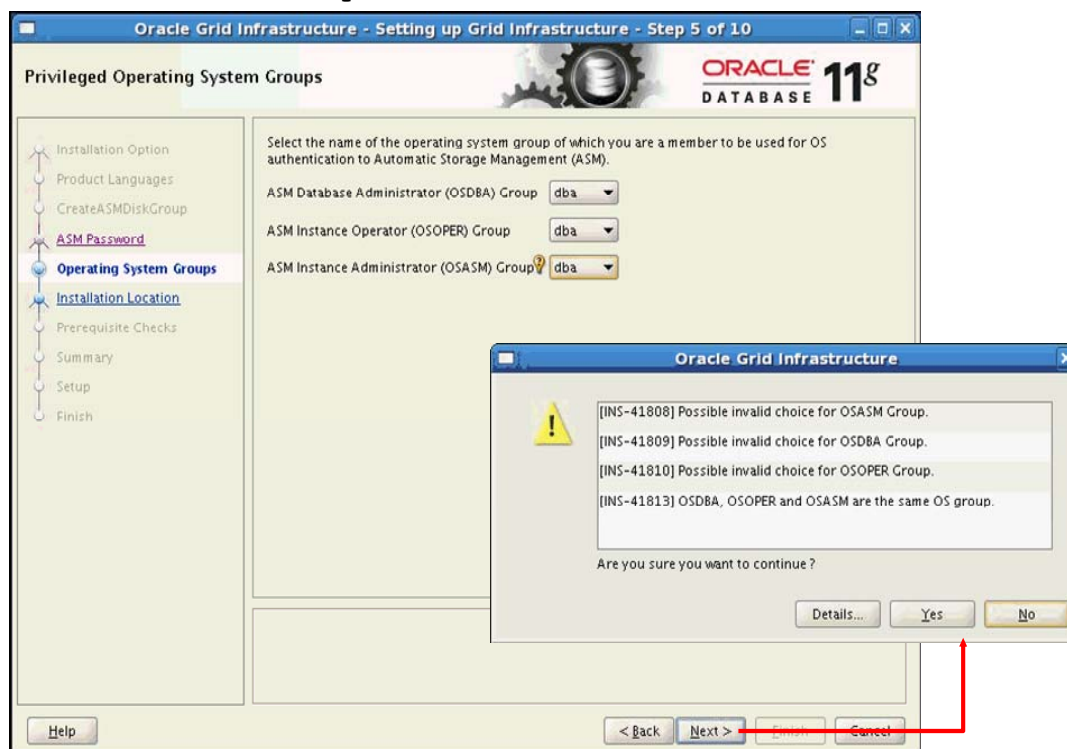
ORACLE

© Oracle, 2009. Все права защищены.

Установка паролей ASM

Открывается страница Specify ASM Password ("Указать пароль ASM"). Пароли должны указываться для двух учетных записей: SYS и ASMSNMP. Существует возможность использования различных паролей для учетных записей или одного пароля. Экземпляр ASM не содержит словаря данных как база данных Oracle, поэтому единственными методами аутентификации являются аутентификация операционной системы и файла паролей. Учетная запись SYS будет добавлена в файл паролей (orapw+ASM в системе Linux). Ей будут предоставлены привилегии SYSDBA, SYSOPER и SYSASM. Учетная запись ASMSNMP будет добавлена в файл паролей, но ей будут предоставлены только привилегии SYSDBA. После ввода соответствующих паролей нажмите кнопку Next ("Далее"), чтобы продолжить установку.

Определение привилегированных групп операционной системы



ORACLE

© Oracle, 2009. Все права защищены.

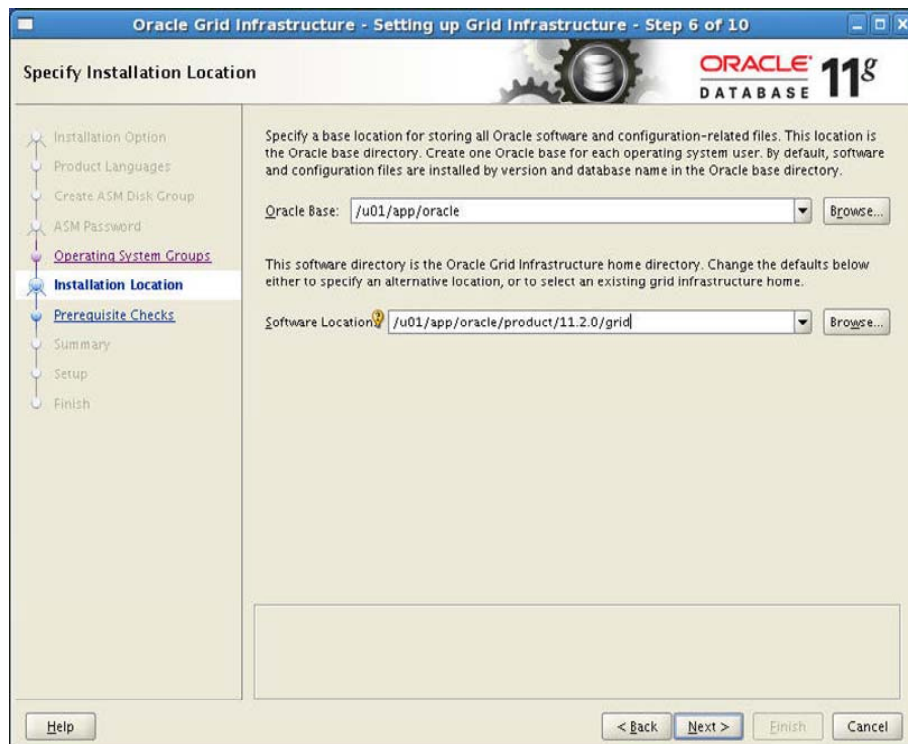
Определение привилегированных групп операционной системы

Отобразится страница Privileged Operating System Groups ("Привилегированные группы операционной системы"). Средство OUI предоставляет рекомендуемые значения по умолчанию для следующих групп, если текущий пользователь является системным участником этих групп:

- группа администраторов БД ASM (OSDBA) – asmdba;
- группа операторов экземпляров ASM (OSOPER) – asmoper;
- группа администраторов экземпляров ASM (OSASM) – asmadmin.

Так как эта установка выполняется на автономном сервере, то для всех обычно используется одна и та же операционная система (например, dba, как показано на слайде). Нажмите кнопку Next ("Далее"), чтобы продолжить установку. Нажмите кнопку Yes ("Да") в окне с предупреждением об использовании одной группы операционной системы для OSDBA, OSOPER и OSASM.

Указание расположения установки



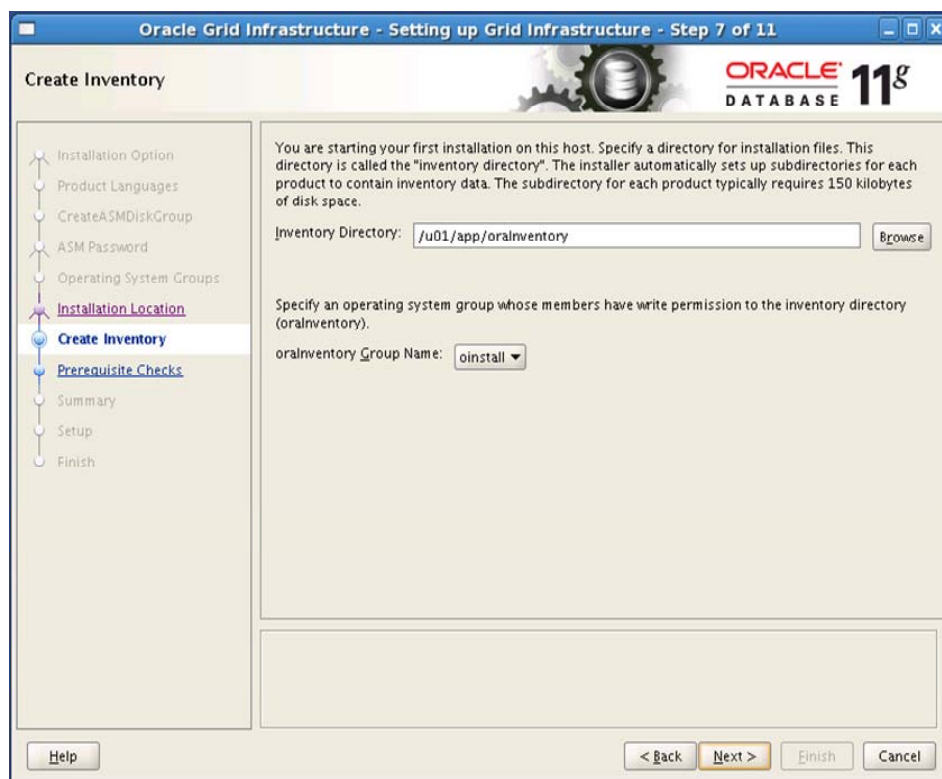
ORACLE

© Oracle, 2009. Все права защищены.

Указание расположения установки

Открывается страница Specify Installation Location ("Указать расположение установки"). В поле Oracle Base ("База Oracle") введите значение `ORACLE_BASE` для владельца ПО. Значение по умолчанию равно `/u01/app/oracle`. В поле Software Location ("Расположение установки") введите значение `ORACLE_HOME` для ПО Grid Infrastructure. Значение по умолчанию равно `/u01/app/oracle/product/11.2.0/grid`. Нажмите кнопку Next ("Далее"), чтобы продолжить установку.

Создание каталога Inventory



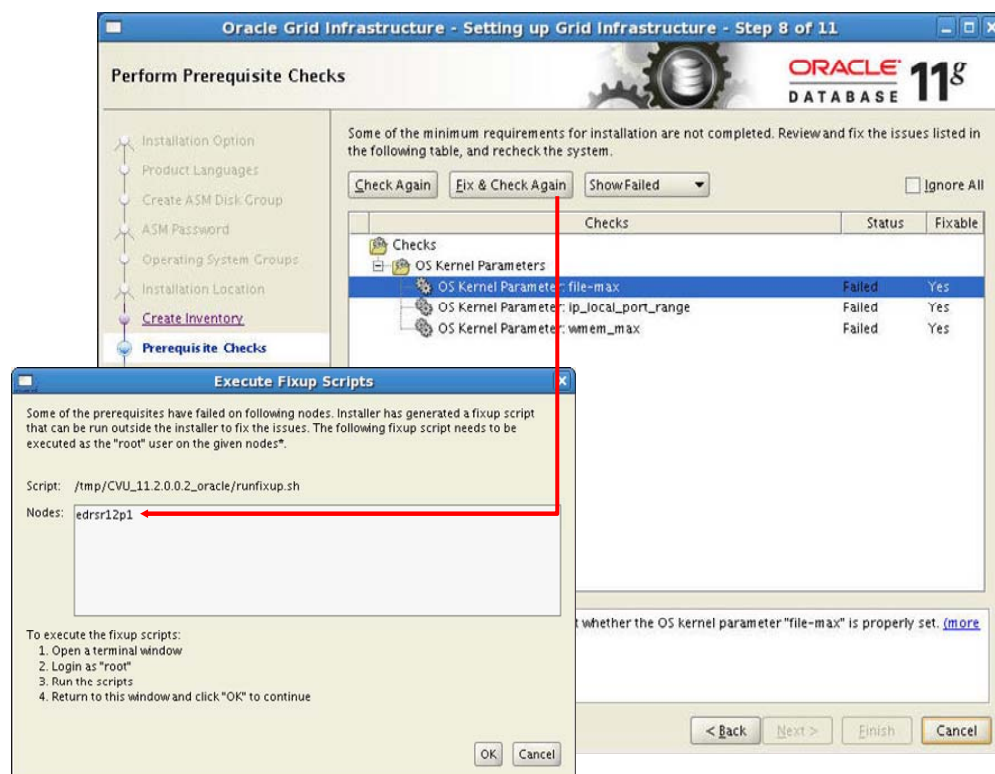
ORACLE

© Oracle, 2009. Все права защищены.

Создание каталога Inventory

Если каталог Oracle Central Inventory на хост-компьютере еще не существует, то отображается страница Create Inventory ("Создать каталог Inventory"). В поле Inventory Directory ("Каталог Inventory") введите требуемое расположение каталога oraInventory. Рекомендуется каталог /u01/app/oraInventory. Каталог oraInventory должен находиться на один уровень выше каталога ORACLE_BASE. Выберите имя группы oraInventory из списка. Рекомендуется имя oinstall. Нажмите кнопку Next ("Далее"), чтобы продолжить установку.

Выполнение предварительных проверок



ORACLE

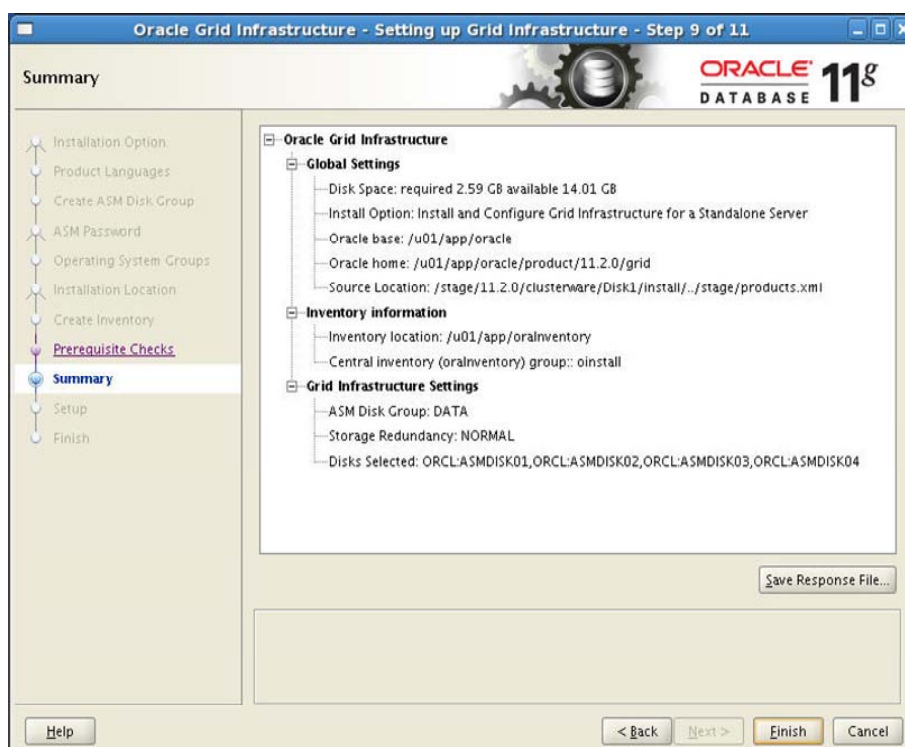
© Oracle, 2009. Все права защищены.

Выполнение предварительных проверок

Отобразится страница Perform Prerequisite Checks ("Выполнить предварительные проверки"). Выполнение действий на этой странице необходимо только в том случае, если проверки завершаются с ошибками. При возникновении ошибок их необходимо исправить перед продолжением установки. В приведенном примере предварительная проверка не пройдена для некоторых параметров ядра. Выводятся указания по запуску сценария, исправляющего эти параметры. После этого необходимо повторно запустить проверку. Если ошибку невозможно устранить автоматически, ее следует устранить вручную или выбрать команду Ignore All ("Игнорировать все") и продолжить установку (если непройденные проверки не являются критически важными).

Примечание. Сценарий runfixup.sh, создаваемый средством OUI, не позволяет исправлять некоторые ошибки. Если ошибка не устранена, то перед повторным запуском установки необходимо устранить ошибку вручную.

Проверка сводных данных установки



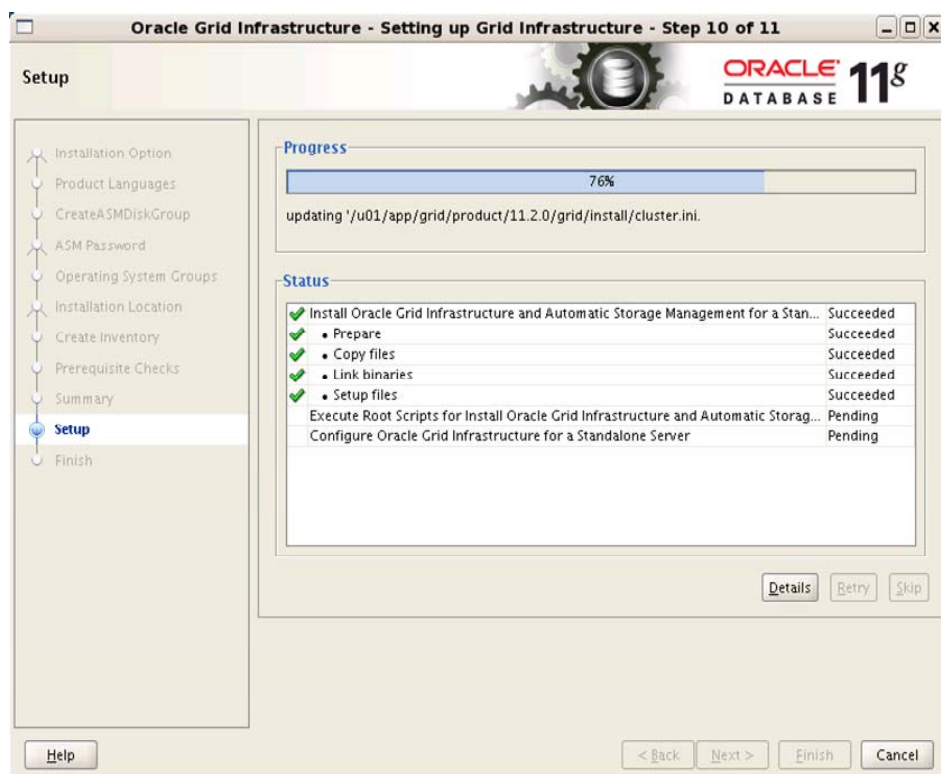
ORACLE

© Oracle, 2009. Все права защищены.

Проверка сводных данных установки

Отобразится страница Summary ("Сводка"). Просмотрите представленные сведения о варианте установки, заданные на предыдущих страницах. Все вопросы и ответы установки можно сохранить в файл ответов. Файл ответов может использоваться в будущем для проведения автоматической установки, схожей с выполняемой в настоящее время. Файл ответов является текстовым файлом. Его можно изменять для другой установки с помощью текстового редактора. Нажмите кнопку Finish ("Готово"), чтобы продолжить установку.

Отслеживание хода установки



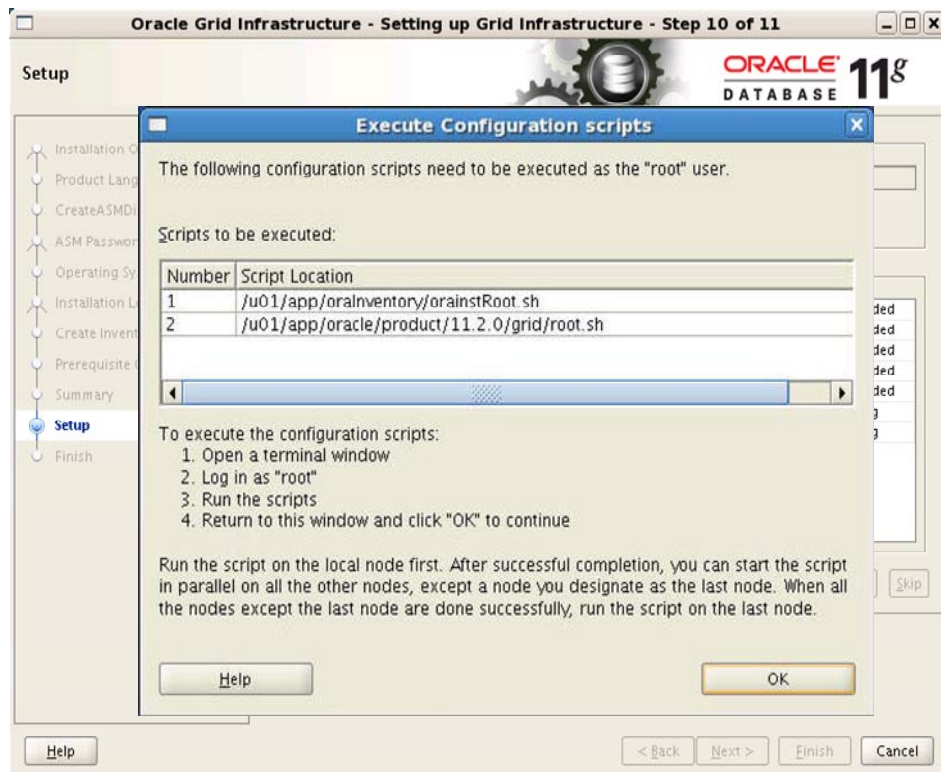
ORACLE

© Oracle, 2009. Все права защищены.

Отслеживание хода установки

Отобразится страница Setup ("Настройка"), на которой представлен ход установки. На этой странице указываются действия по подготовке к установке, копированию файлов, связыванию библиотек и созданию файлов установки. Выполнение действий на этой странице необходимо только в том случае, если установка завершается с ошибкой. При возникновении ошибок их необходимо исправить перед продолжением установки.

Выполнение сценариев конфигурации root



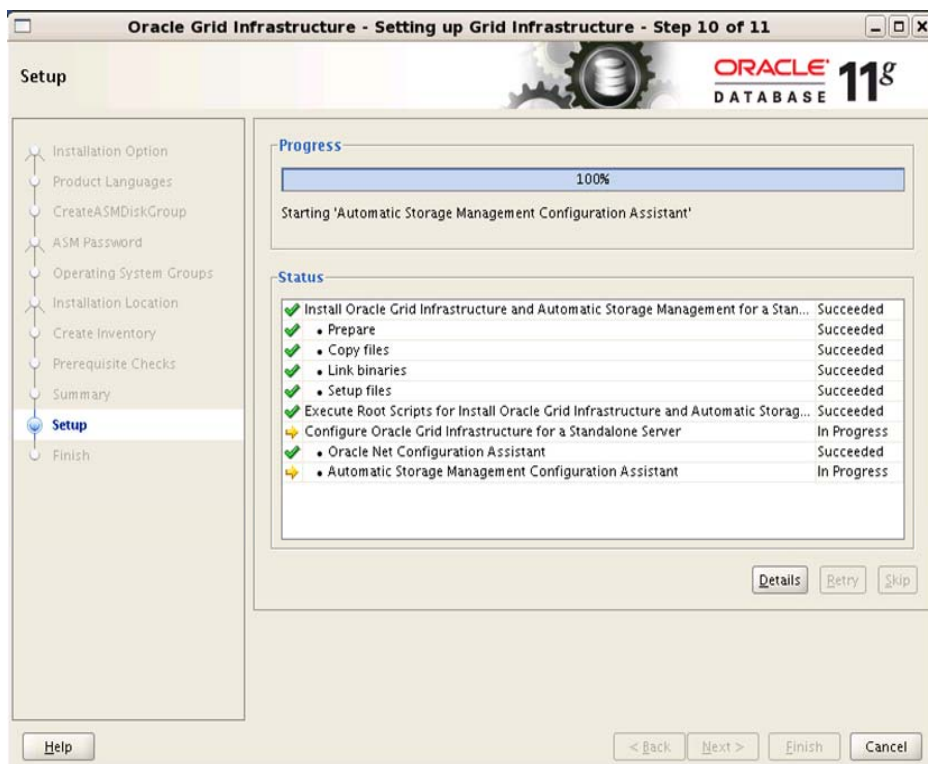
ORACLE

© Oracle, 2009. Все права защищены.

Выполнение сценариев конфигурации root

Отобразится диалоговое окно Execute Configuration Scripts ("Выполнить сценарии конфигурации") с указанием сценариев конфигурации, которые должны быть выполнены от имени пользователя root. Сценарий orainstRoot.sh изменяет разрешения каталога Oracle Central Inventory путем предоставления группе разрешений на чтение и запись, а также удаления разрешений на чтение, запись и исполнение для всех остальных пользователей. Сценарий root.sh копирует файлы в каталог /usr/local/bin, создает файл /etc/oratab, создает ключи OCR для пользователя grid, запускает демон ohasd и изменяет /etc/inittab для автоматического запуска демона ohasd при запуске компьютера. После выполнения сценариев root нажмите кнопку ОК, чтобы вернуться к странице хода выполнения Setup ("Настройка") и продолжить установку.

Запуск помощников Configuration Assistant



ORACLE

© Oracle, 2009. Все права защищены.

Запуск помощников Configuration Assistant

Средство Oracle Net Configuration Assistant (netca) запускается при автоматической установке для создания файлов конфигурации сети. После этого запускается помощник Automatic Storage Management Configuration Assistant (asmca), создающий группу дисков ASM и регистрирующий компоненты ASM в Oracle Restart.

Завершение установки



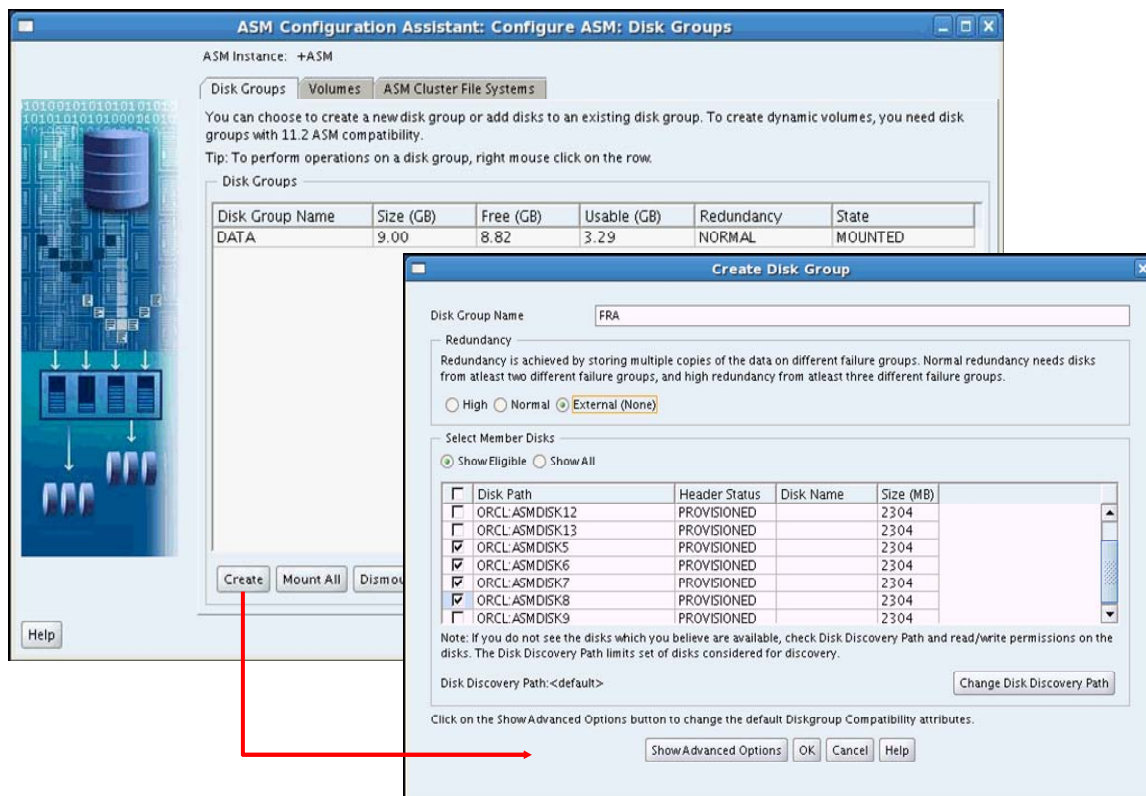
ORACLE

© Oracle, 2009. Все права защищены.

Завершение установки

После завершения установки отображается страница Finish ("Готово"). Нажмите кнопку Close ("Закрыть"), чтобы выйти из служебной программы OUI.

Настройка группы дисков FRA



ORACLE

© Oracle, 2009. Все права защищены.

Настройка группы дисков FRA

Так как в ходе установки Oracle Grid Infrastructure можно настроить только одну группу дисков, создание дополнительных групп дисков выполняется вручную. В рассматриваемом сценарии требуется группа дисков FRA, которая используется для области мгновенного восстановления БД. Помощник ASM Configuration Assistant (asmca) имеет интуитивно понятный графический интерфейс, позволяющий быстро создавать или удалять группы дисков ASM.

Экспресс-тест

Универсальное средство установки выполняет все необходимые настройки для установки ПО Oracle.

1. Да
2. Нет

The Oracle logo is displayed in white capital letters on a red rectangular background.

© Oracle, 2009. Все права защищены.

Ответ: 2

Комментарии к тесту

Перед установкой необходимо создать в UNIX и Linux пользователей и группы операционной системы.

Экспресс-тест

При установке Grid infrastructure можно выполнить следующие действия:

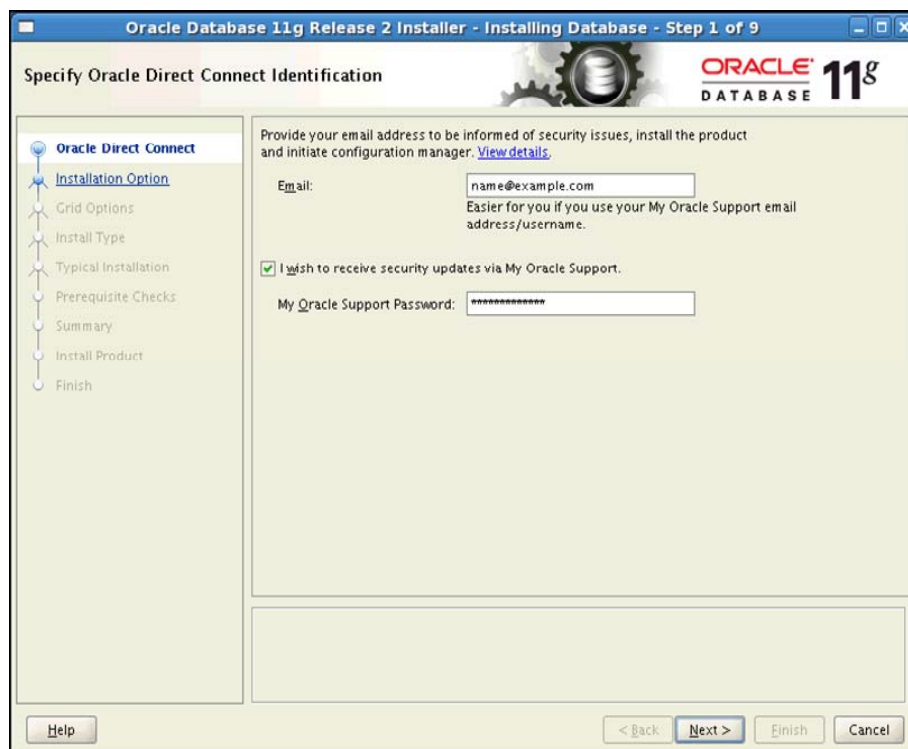
1. указать точное расположение файлов данных экземпляра **ASM**;
2. создать только одну группу DISKGROUP;
3. указать размер SGA для экземпляра ASM;
4. создать несколько групп DISKGROUP.

ORACLE

© Oracle, 2009. Все права защищены.

Ответ: 2

Часть 2. Установка программного обеспечения базы данных Oracle



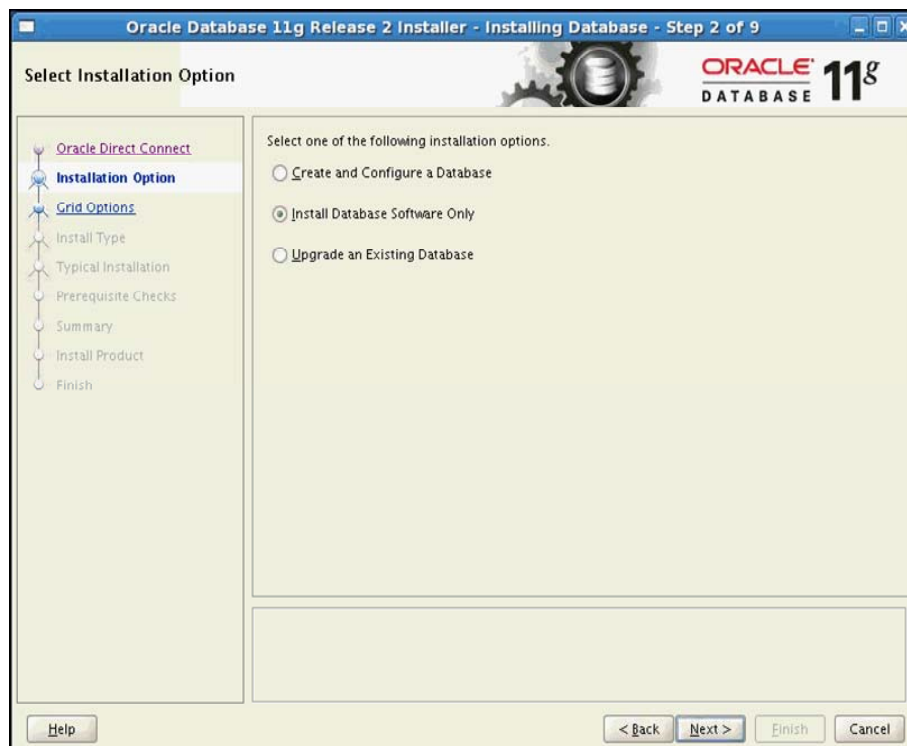
ORACLE

© Oracle, 2009. Все права защищены.

Часть 2. Установка программного обеспечения базы данных Oracle

Рассмотрим этапы установки ПО базы данных Oracle. Как и в предыдущем случае, выполните вход в ОС вашего компьютера с правами участника группы администраторов, у которого есть разрешение на установку ПО Oracle, а также создание и управление базой данных. Вставьте в привод для чтения компакт-диск с дистрибутивом или перейдите к каталогу с дистрибутивом БД Oracle. Введите в этой папке `./runInstaller` для запуска средства OUI. При необходимости введите адрес электронной почты, на который следует отправлять уведомления об угрозах безопасности. Чтобы получать обновления безопасности через службу My Oracle Support, укажите свой пароль My Oracle Support. Если адрес не введен, отобразится запрос для подтверждения того, что информирование о критических ошибках конфигурации не требуется. Нажмите кнопку Yes ("Да"), чтобы продолжить установку.

Выбор типа установки



ORACLE

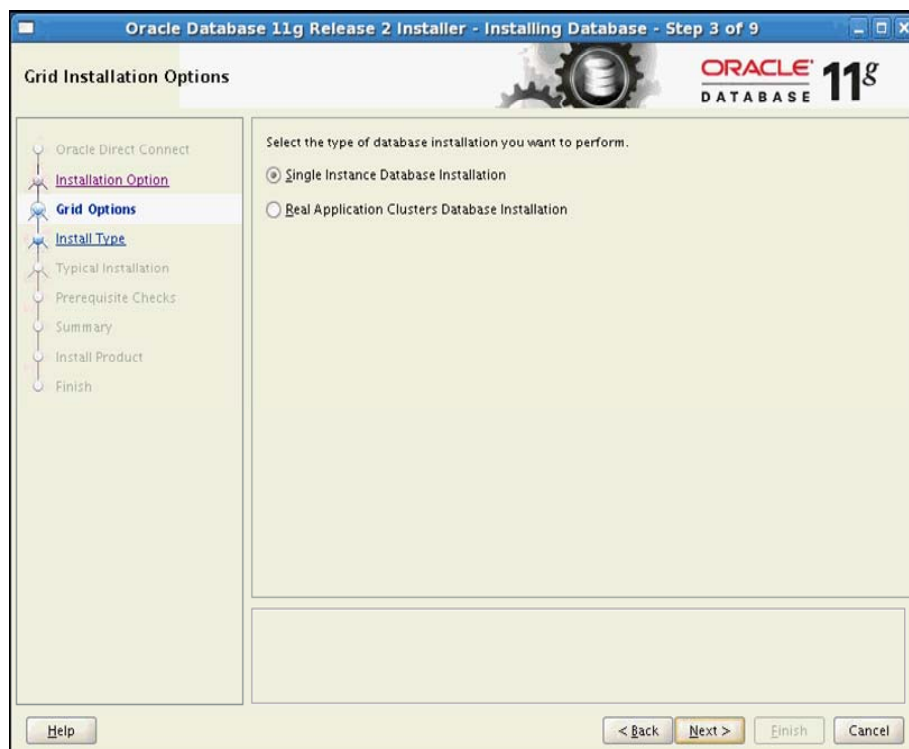
© Oracle, 2009. Все права защищены.

Выбор типа установки

Отобразится окно Select Installation Option ("Выбрать вариант установки"). Выберите необходимый тип установки и нажмите кнопку Next ("Далее"):

- **Create and Configure a Database ("Создать и настроить базу данных"):**
после установки продукта создается база данных.
- **Install Database Software Only ("Установить только ПО базы данных"):**
устанавливаются только двоичные файлы базы данных Oracle.
- **Upgrade an Existing Database ("Обновить существующую базу данных"):**
этот вариант используется для обновления базы данных более ранней версии.

Выбор вариантов установки Infrastructure Grid



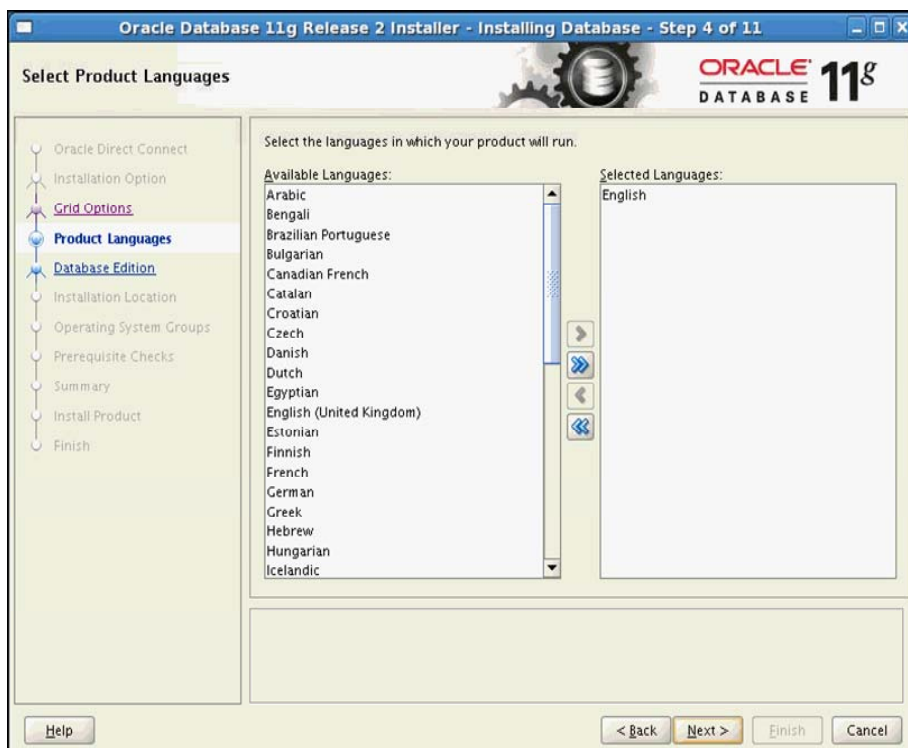
ORACLE

© Oracle, 2009. Все права защищены.

Выбор вариантов установки Infrastructure Grid

Отобразится страница Grid Installation Option ("Вариант установки Infrastructure Grid"). Выберите необходимость установки одного экземпляра базы данных или базы данных Oracle RAC в кластере. Для продолжения нажмите кнопку Next ("Далее").

Выбор параметров языка



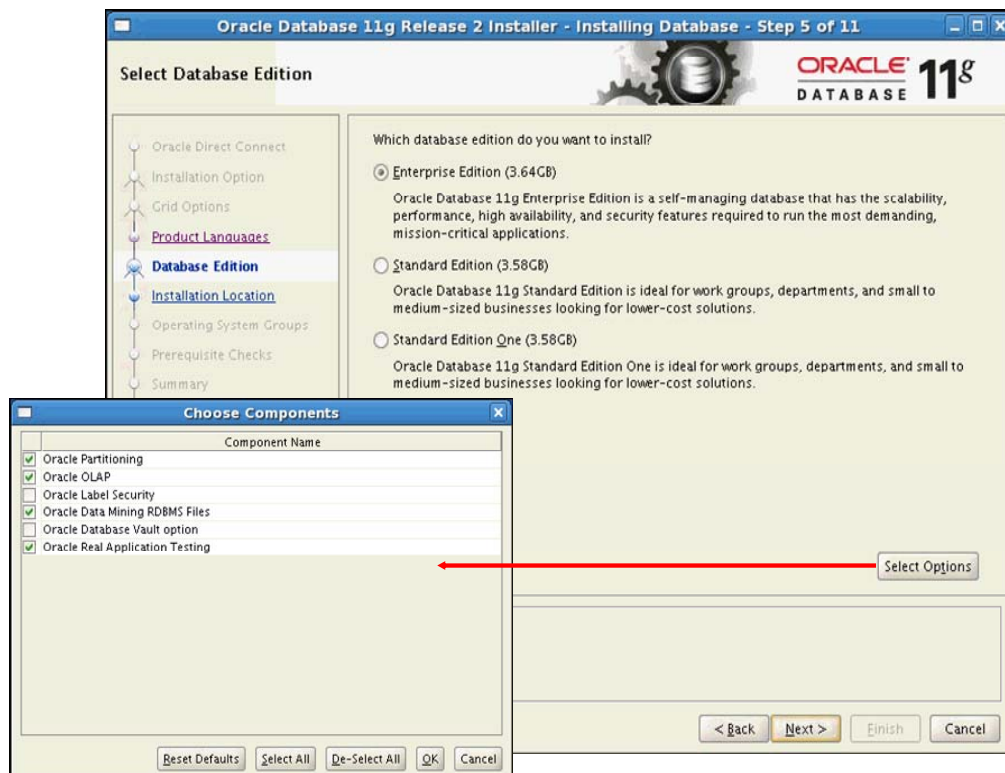
ORACLE

© Oracle, 2009. Все права защищены.

Выбор параметров языка

Отобразится страница Select Product Languages ("Выбор языков продукта"). На ней можно выбрать языки, которые будут использоваться в продукте. Английский язык выбран по умолчанию. Его невозможно удалить из списка Selected Languages ("Выбранные языки"). Для продолжения нажмите кнопку Next ("Далее").

Выбор выпуска базы данных



ORACLE

© Oracle, 2009. Все права защищены.

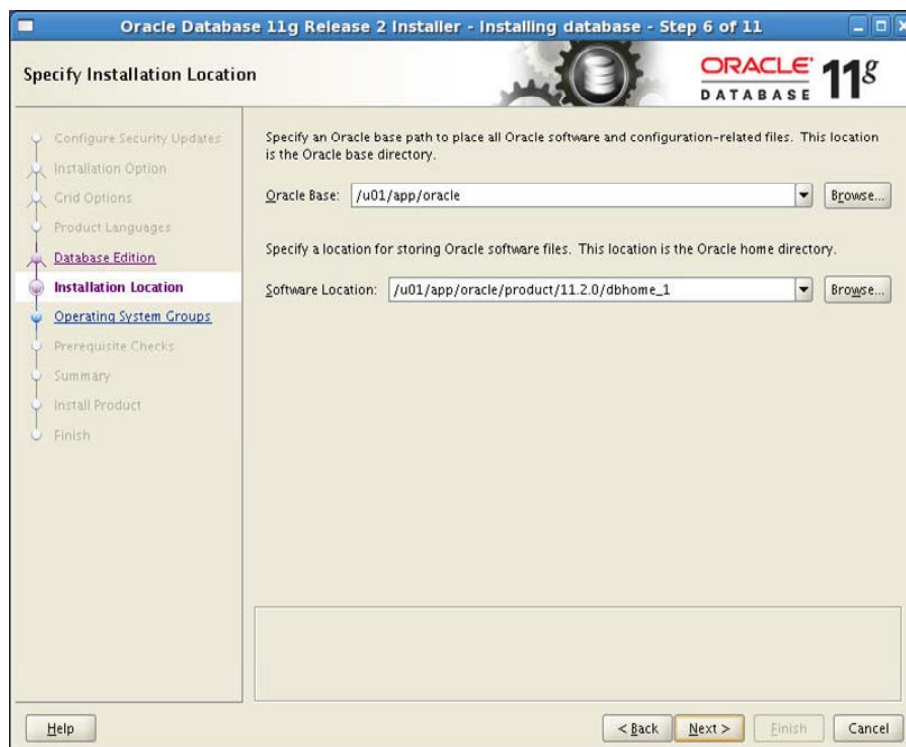
Выбор выпуска базы данных

Отобразится страница Select Database Edition ("Выбрать выпуск базы данных"). Выберите один из следующих выпусков:

- **Enterprise Edition:** этот выпуск содержит базу данных с функцией автономного управления, которая обеспечивает большую производительность, масштабируемость, безопасность и надежность на кластеризованных или отдельных серверах. В этот выпуск входит полный набор функций, позволяющий управлять самыми требовательными приложениями для обработки транзакций, сбора информации и управления содержимым.
- **Standard Edition:** этот выпуск содержит полнофункциональную базу данных для серверов, имеющих до четырех сокетов. Выпуск также включает в себя кластеры Oracle Real Application Cluster для повышения доступности, обеспечивает производительность и безопасность корпоративного класса, прост в управлении и может быстро масштабироваться при необходимости. Этот выпуск совместим с выпуском Enterprise.
- **Standard Edition One:** этот выпуск содержит полнофункциональную базу данных для серверов, имеющих до двух сокетов. Он обеспечивает производительность и безопасность корпоративного класса, прост в управлении и может быстро масштабироваться при необходимости. Этот выпуск также совместим с другими выпусками.

Нажмите кнопку Select Options ("Выбрать параметры"), чтобы определить устанавливаемые компоненты. Нажмите кнопку Next ("Далее") после выбора всех необходимых параметров на странице Select Database Edition ("Выбрать выпуск базы данных").

Указание расположения установки



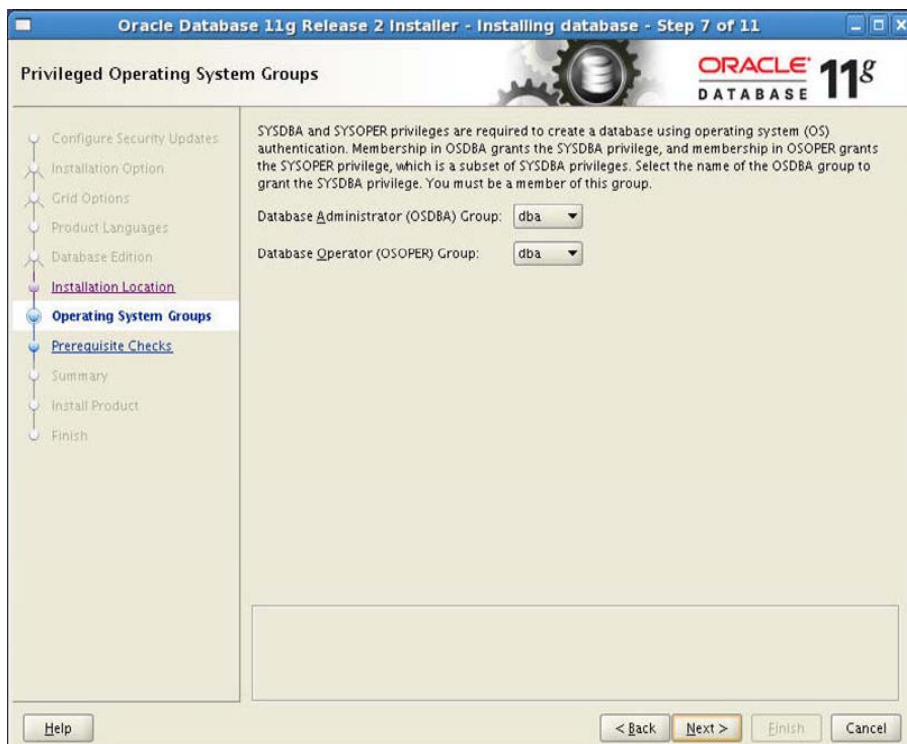
ORACLE

© Oracle, 2009. Все права защищены.

Указание расположения установки

Открывается страница Installation Location ("Расположение установки"). По умолчанию отображается основной путь Oracle. При необходимости этот путь можно изменить. В разделе Software Location ("Расположение установки") можно принять значения по умолчанию или ввести имя домашнего каталога Oracle и путь для установки компонентов Oracle. Путь каталога не может содержать пробелов. Нажмите кнопку Next ("Далее"), чтобы продолжить процесс установки.

Выбор групп операционной системы



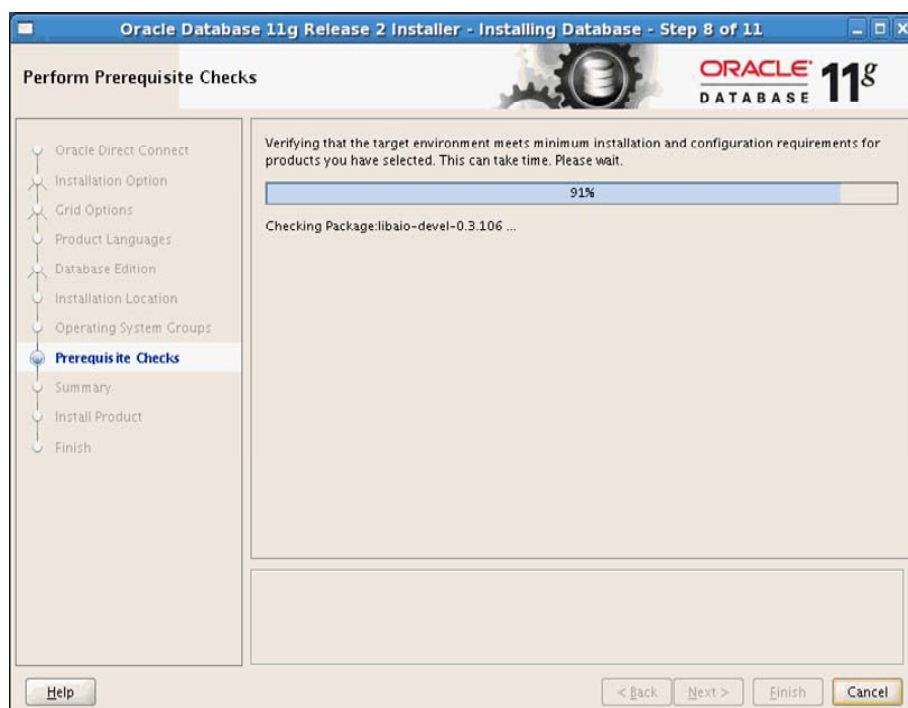
ORACLE

© Oracle, 2009. Все права защищены.

Выбор групп операционной системы

Отобразится страница Privileged Operating System Groups ("Привилегированные группы операционной системы"). Выберите соответствующую группу операционной системы для привилегий OSDBA и OSOPER. По умолчанию группа dba выбрана для привилегии OSDBA, а группа oper — для OSOPER. В данном примере для обеих привилегий используется группа dba, так как выполняется автономная установка, не подразумевающая разделения обязанностей. Для продолжения нажмите кнопку Next ("Далее").

Выполнение предварительных проверок



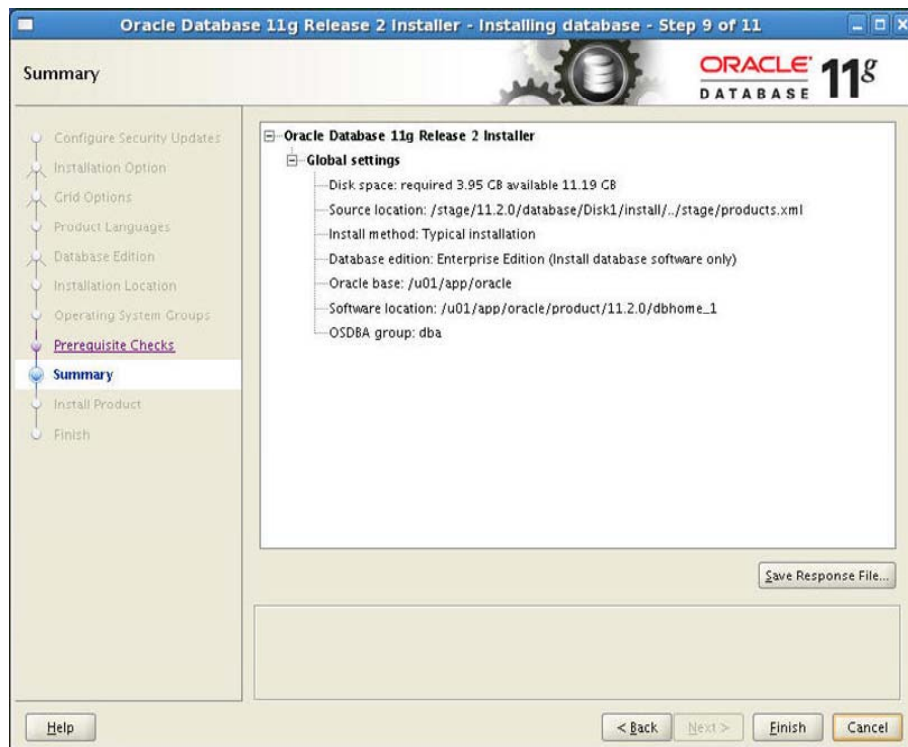
ORACLE

© Oracle, 2009. Все права защищены.

Выполнение предварительных проверок

Отобразится страница Perform Prerequisite Checks ("Выполнение проверки"). Средство OUI выполняет различные проверки. После выполнения 100% всех проверок средство OUI возвращает информацию о наличии ошибок проверки. Выполнение действий на этой странице необходимо только в том случае, если проверки завершаются с ошибками. При наличии ошибок будет открыта страница с кнопкой Fix & Check Again ("Исправить и проверить еще раз"), которая позволяет средству OUI создать сценарий для устранения ошибок. Если ошибку невозможно устранить автоматически, ее следует устранить вручную или выбрать команду Ignore All ("Игнорировать все") и продолжить установку (если непройденные проверки не являются критически важными).

Страница сводки установки



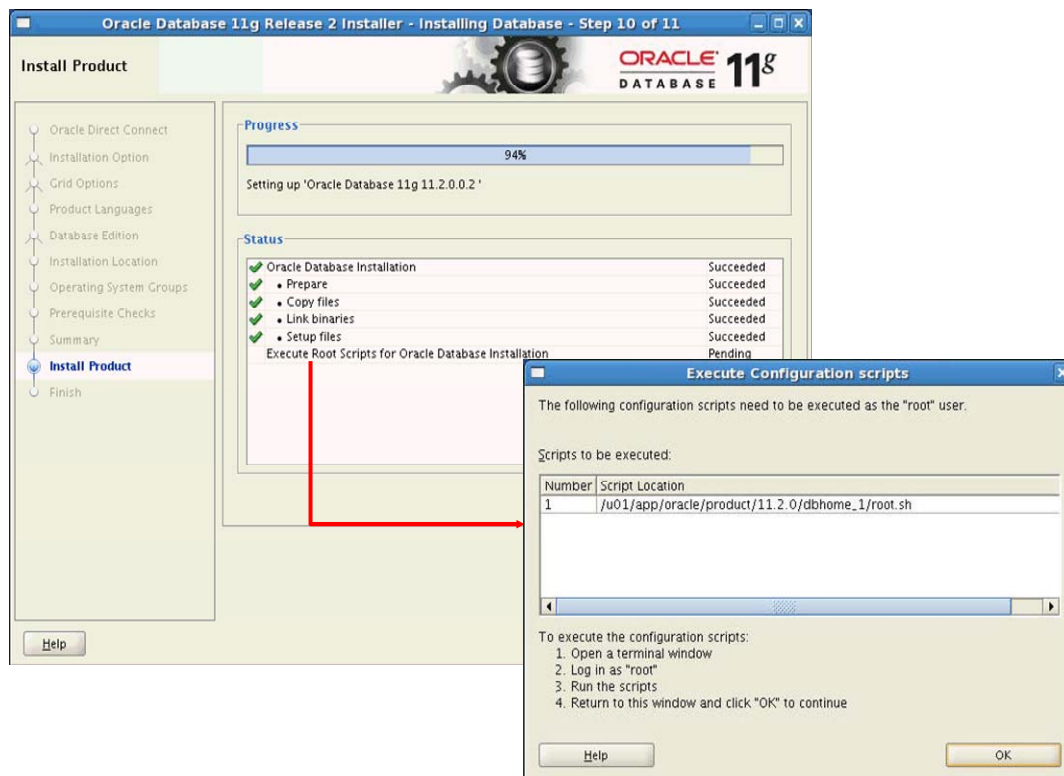
ORACLE

© Oracle, 2009. Все права защищены.

Страница сводки установки

Отобразится страница Summary ("Сводка"). Просмотрите представленные сведения о варианте установки, заданные на предыдущих страницах. Все вопросы и ответы установки можно сохранить в файл ответов. Файл ответов может использоваться в будущем для проведения автоматической установки, схожей с выполняемой в настоящее время. Файл ответов является текстовым файлом. Его можно изменять для другой установки с помощью текстового редактора. Нажмите кнопку Finish ("Готово"), чтобы продолжить установку.

Страница установки продукта



ORACLE

© Oracle, 2009. Все права защищены.

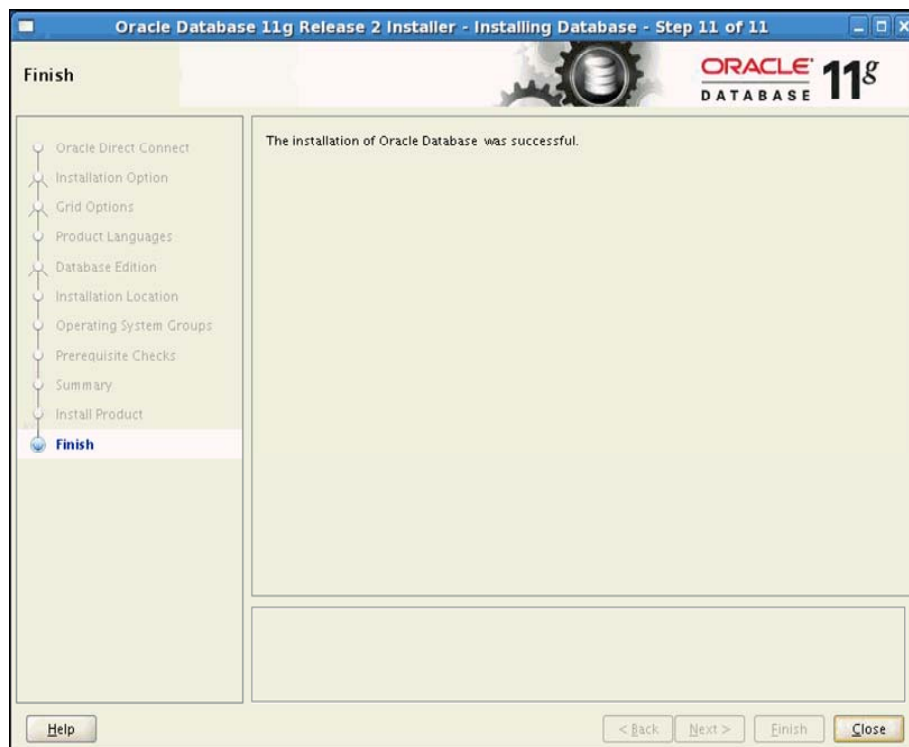
Страница установки продукта

Процесс установки в этот момент приостанавливается, что требует выполнение дополнительного конфигурационного сценария от имени пользователя `root`. В отдельном окне терминала введите следующую команду:

```
$ su
# password: oracle <корневой пароль, в окне не отображается>
# /u01/app/oracle/product/11.2.0/dbhome_1/root.sh
```

При установке в ОС Linux или UNIX примите настройки по умолчанию для локального каталога `bin`. После завершения работы сценариев выйдите из учетной записи `root` и закройте окно. После этого нажмите кнопку **OK** в диалоговом окне **Execute Configuration scripts** ("Выполнить сценарии конфигурации"), чтобы завершить установку.

Установка готова



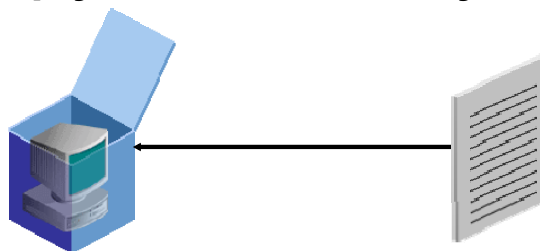
ORACLE

© Oracle, 2009. Все права защищены.

Установка готова

После завершения всех действий по установке отображается страница Finish ("Готово"). Нажмите кнопку Close ("Заккрыть"), чтобы выйти из служебной программы OUI.

Параметр установки: бесшумный режим



Чтобы установить и настроить продукты Oracle с помощью OUI в бесшумном режиме, выполните следующие действия:

1. Подготовьте файл ответов:

- с помощью одного из шаблонов, поставляемых вместе с ПО Oracle;
- путем записи файла ответов в ходе процесса установки нажав в средстве OUI кнопку Save Response File ("Сохранить файл ответов") на странице сводки.

2. Запустите средство OUI в бесшумном или принудительном режиме.

```
./runInstaller -silent -responsefile <имя_файла>
```

При необходимости запустите NetCA и DBCA в бесшумном режиме.

ORACLE

© Oracle, 2009. Все права защищены.

Параметр установки: бесшумный режим

Чтобы установить и настроить продукты Oracle с помощью OUI в бесшумном или принудительном режиме:

1. Подготовьте файл ответов. Для этого можно использовать один из шаблонов, которые предоставляются для каждого типа продукта и установки, например `enterprise.rsp`, `standard.rsp` и `netca.rsp`. Если эти шаблоны не подходят для конкретного сценария, можно записать файл ответов с помощью OUI в интерактивном режиме, отредактировать его и затем использовать для установки в бесшумном или принудительном режиме. Для этого нажмите кнопку Save Response File ("Сохранить файл ответов") на странице сводки в ходе интерактивной установки с помощью OUI.
2. Запустите OUI в бесшумном или принудительном режиме. В конце установки запустите `$ORACLE_BASE/oraInventory/orainstRoot.sh` и `$ORACLE_HOME/root.sh`. Если выполнена только установка программного обеспечения, запустите помощников Oracle Net Configuration Assistant (NetCA) и Database Configuration Assistant (DBCA) в бесшумном или неинтерактивном режиме (при необходимости).

Подробнее об установке см. в руководстве *Oracle Database Installation Guide* для соответствующей ОС.

Экспресс-тест

Файл ответа – это...

1. двоичный файл, который можно изменять в редакторе двоичных данных;
2. **двоичный файл, который создается программой установки;**
3. текстовый файл, который не редактируется, но может создаваться программой установки;
4. текстовый файл, который может редактироваться в текстовом редакторе.

ORACLE

© Oracle, 2009. Все права защищены.

Ответ: 4

Экспресс-тест

При установке ПО базы данных можно указать группы для...

1. группы osoper;
2. группы osasm;
3. группы osdba;
4. группы osadmin.

ORACLE

© Oracle, 2009. Все права защищены.

Ответы: 1, 3

Заключение

В ходе прохождения урока вы освоили следующие темы:

- роль администратора базы данных (DBA), стандартные задачи и средства;
- **планирование установки базы данных Oracle;**
- установка инфраструктуры распределенных вычислений Oracle Grid Infrastructure для автономного сервера;
- установка программного обеспечения базы данных **Oracle.**

ORACLE

© Oracle, 2009. Все права защищены.

Обзор упражнения 2: подготовка среды базы данных

В данном упражнении рассматривается установка ПО Oracle с помощью Oracle Universal Installer.

Примечание. Выполнение данного упражнения критически важно для выполнения всех последующих упражнений.

ORACLE

© Oracle, 2009. Все права защищены.

3

Создание базы данных Oracle с помощью DBCA

ORACLE

© Oracle, 2009. Все права защищены.

Цели

Изучив материал этого занятия, вы освоите следующие темы:

- создание баз данных с использованием помощника Database Configuration Assistant (DBCA);
- **генерация сценариев создания базы данных с помощью DBCA;**
- управление шаблонами баз данных с помощью DBCA;
- выполнение дополнительных задач с помощью DBCA.

ORACLE

© Oracle, 2009. Все права защищены.

Проектирование базы данных

Как администратору БД, вам следует спроектировать:

- логическую структуру хранения базы данных и ее физическую реализацию:
 - **Сколько доступно жестких дисков?**
Хранилище какого типа используется?
 - Сколько необходимо файлов данных?
(план с перспективой роста)
 - Сколько табличных пространств будет использоваться?
 - **Какого типа информация будет храниться в БД?**
 - Существуют ли какие-либо особые требования к хранению, обусловленные типом или размером?
- общую структуру базы данных;
- стратегию резервного копирования базы данных.



ORACLE

© Oracle, 2009. Все права защищены.

Проектирование базы данных

При проектировании важно учитывать, как логическая структура хранения базы данных будет влиять на производительность системы, а также различные операции по управлению базой данных. К примеру, прежде чем создать в базе данных табличные пространства, необходимо знать, из скольких файлов данных они будут состоять, какого типа информацию будет хранить каждое из них, а также на каких дисковых накопителях будут расположены файлы данных. Большое значение имеет информация о доступности сетевых устройств хранения данных (NAS) и пропускной способности частной сети хранения данных. Если будут использоваться локальные сети хранения данных (SAN), важно знать конфигурацию логических томов и размер разнесения.

При проектировании общего логического хранилища в структуре базы данных следует принять во внимание факторы, возникающие после создания и запуска базы данных. В базе данных могут присутствовать объекты со специальными требованиями к хранению, обусловленными типом или размером.

Этот этап проектирования чрезвычайно важен в случае распределенной среды баз данных. Физическое расположение данных с частым обращением значительно влияет на производительность приложений.

На этапе проектирования также необходимо разработать стратегию резервного копирования базы данных. Для повышения эффективности этой стратегии можно изменить структуру логического хранилища или структуру базы данных. Стратегии резервного копирования будут рассмотрены в одном из следующих уроков.

Базы данных: примеры

- Общее назначение или обработка транзакций:
 - система интерактивной обработки транзакций (OLTP), например, розничная биллинговая система для компании-разработчика ПО или медицинской компании
- Пользовательские:
 - базы данных универсального назначения (например, совмещение функций OLTP и функций хранилища данных)
- Хранилища данных:
 - **маркетинговые и исследовательские данные**
 - данные государственного или федерального налогообложения
 - данные о профессиональных лицензиях (врачи, медсестры и т.д.)

ORACLE

© Oracle, 2009. Все права защищены.

Базы данных: примеры

Каждый тип базы данных имеет собственные требования к экземплярам и хранению. В комплект ПО базы данных Oracle входят шаблоны для создания баз данных различного типа. В примерах приведены базы данных следующего типа:

- **Общего назначения:** эти базы предназначены для общих целей или для обработки транзакций (например, для работы с транзакциями и их хранения в течение среднего срока.
- **Пользовательские:** для пользовательских баз данных, которые не входят в категорию общего назначения и не соответствуют шаблонам хранилища данных.
- **Хранилища данных:** служат для долговременного хранения данных и получения информации с помощью операций чтения.

На текущей и предыдущей страницах отображены вопросы, с которыми сталкивается DBA. Данный курс целиком и полностью посвящен освещению этих вопросов.

Выбор соответствующего набора символов

- База данных Oracle поддерживает различные классы схем кодировки символов:
 - однобайтовые наборы символов
 - 7-битные
 - 8-битные
 - многобайтовые наборы символов, в т.ч. Юникод
- Набор символов выбирается во время создания базы данных. Выберите набор символов, который оптимально соответствует текущим и будущим требованиям бизнеса, так как последующее изменение набора символов может быть очень трудоемким.
- В общем случае рекомендуется набор символов Юникод, так как он является наиболее гибким.

ORACLE

© Oracle, 2009. Все права защищены.

Выбор соответствующего набора символов

При обработке символов компьютер использует числовые коды вместо графического представления символа. *Закодированный набор символов* сопоставляет числовые коды с символами, которые компьютер или терминал может отображать и принимать. Различные наборы символов поддерживают различные алфавиты. Так как наборы символов обычно основаны на определенной системе письма, то они могут поддерживать несколько языков. При этом основанные на одной системе письма наборы ограничены определенными группами языков, имеющих схожую графику. Универсальные наборы символов охватывают большинство современных систем письма и являются более универсальным решением для поддержки нескольких языков. Сведения о стандарте Юникод см. на web-узле <http://www.unicode.org>.

База данных Oracle поддерживает три класса схем кодировки: однобайтовые, многобайтовые переменной длины и универсальные. Выберите набор символов, который оптимально соответствует текущим и будущим требованиям бизнеса, так как последующее изменение набора символов может быть очень трудоемким. Для оптимальной производительности выберите набор символов, не требующий преобразования набора символов и использующий наиболее эффективную кодировку для требуемых языков. Однобайтовые наборы символов обеспечивают лучшую производительность по сравнению с многобайтовыми, а также являются наиболее эффективными по требованиям к свободному пространству. При этом однобайтовые наборы символов ограничивают число поддерживаемых языков. Чтобы выбрать корректный набор символов для базы данных, оцените текущие и будущие требования бизнеса, а также технические требования (например, стандарты XML и Java требуют использования Юникода). В общем случае корпорация Oracle рекомендует использовать Юникод для всех новых баз данных, так как он является наиболее гибким набором символов и не требует последующих преобразований.

Выбор соответствующего набора символов (продолжение)

Однобайтовые наборы символов

В однобайтовом наборе символов один символ занимает один байт. Однобайтовые 7-битные схемы кодировки могут определять до 128 (2^7) символов, а однобайтовые 8-битные схемы — до 256 (2^8) символов.

Примеры однобайтовых схем

7-битный набор символов:

- American Standard Code for Information Interchange (ASCII), 7-битный американский набор (US7ASCII)

8-битный набор символов:

- International Organization for Standards (ISO) 8859-1, западноевропейский набор (WE8ISO8859P1)
- DEC 8-битный, западноевропейский набор (WE8DEC)
- Кодовая страница Extended Binary Coded Decimal Interchange Code (EBCDIC) 1144, 8-битная, итальянский язык (I8EBCDIC1144)

Многобайтовые наборы символов

Многобайтовый набор символов с переменной длиной отличается тем, что символ занимает один или несколько байтов. Многобайтовые наборы символов обычно используются для поддержки азиатских языков. Некоторые многобайтовые схемы кодировки используют значение наиболее значимого бита для указания того, является ли байт самостоятельным или входит в последовательность байтов, представляющих собой один символ. При этом другие схемы кодировки символов отличают однобайтовые символы от многобайтовых. Отправляемый устройством код перехода на верхний регистр указывает на то, что все байты до отправки кода перехода на нижний регистр являются двухбайтовыми символами. Зависимые от регистра схемы кодировки в основном используются на платформах IBM.

Юникод — это универсальный набор символов, который позволяет хранить информацию любого языка в одном наборе символов. Юникод предоставляет уникальное значение кода для каждого символа вне зависимости от платформы, программы или языка.

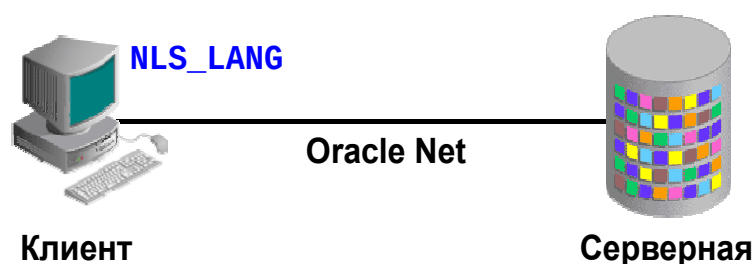
Стандарт Юникод принят многими поставщиками аппаратного и программного обеспечения. Юникод также поддерживается многими операционными системами и браузерами. Наличие Юникода требуется стандартами XML, Java, JavaScript, LDAP и WML. Он также синхронизирован со стандартом ISO/IEC 10646.

Примеры многобайтовых схем с переменной длиной

- Shift-JIS 16-битный набор, японский язык (JA16SJIS)
- Кодовая страница MS Windows Code Page 950 с дополнительным набором символов HKSCS-2001 для Гонконга (ZHT16HKSCS)
- **Универсальный набор символов Юникод 4.0 UTF-8 (AL32UTF8)** - тип кодировки с переменной длиной, расширенная версия ASCII.
- Юникод (AL16UTF16 – 16-битная кодировка Юникода, которая используется Microsoft Windows 2000 и Windows XP.

Как используются наборы символов?

- Oracle Net сравнивает параметр NLS_LANG клиента с набором символов на сервере.
- При необходимости выполняется автоматическое преобразование.



ORACLE

© Oracle, 2009. Все права защищены.

Как используются наборы символов?

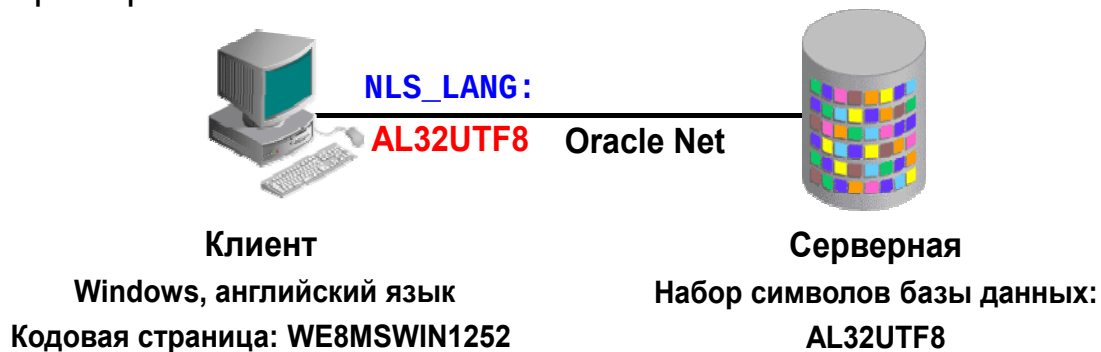
Параметр NLS_LANG определяет схему кодировки терминала клиента. В различных клиентах могут использоваться различные схемы кодировки. Данные, которыми обмениваются клиент и сервер, автоматически преобразуются из одной схемы кодировки в другую. Схема кодировки базы данных должна быть надмножеством или эквивалентом для всех схем кодировки клиентов. Преобразование выполняется прозрачно для приложения клиента.

Если наборы символов базы данных и клиента совпадают, то предполагается, что все получаемые и отправляемые данные принадлежат одному и тому же набору символов, поэтому проверки и преобразования не выполняются.

Преобразование наборов символов может потребоваться в среде "клиент-сервер", если клиентское приложение выполняется на платформе, отличной от платформы сервера, а платформы используют различные схемы кодировки символов. Данные, которыми обмениваются клиент и сервер, должны преобразовываться из одной схемы кодировки в другую. Преобразование символов выполняется автоматически и прозрачным образом с помощью Oracle Net.

Возможные проблемы

Пример



Преобразование не выполняется, так как оно не требуется согласно введенным параметрам.

Проблема: в базу данных вводятся недействительные данные.

ORACLE

© Oracle, 2009. Все права защищены.

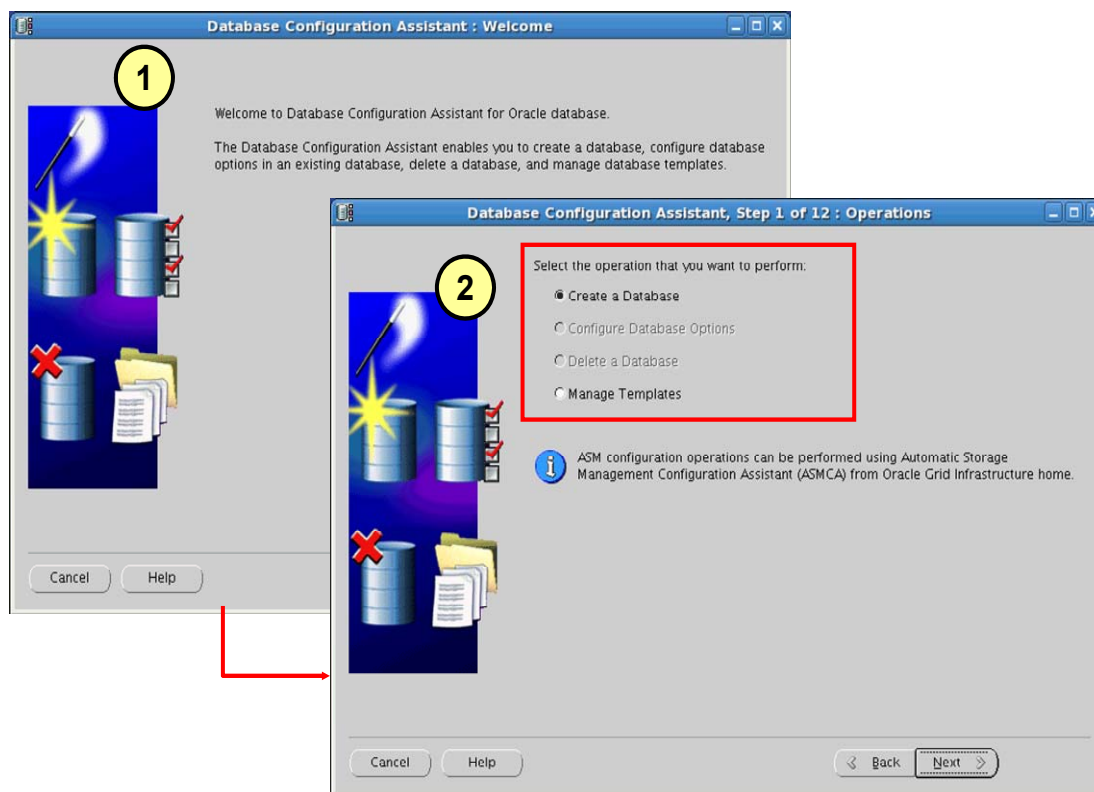
Возможные проблемы

Недействительные данные обычно вводятся в базу при некорректной установке параметра NLS_LANG в клиенте. Значение NLS_LANG должно соответствовать кодировке входящих данных.

- Если параметр NLS_LANG установлен корректно, то база данных автоматически преобразует входящие данные из клиентской операционной системы.
- Если параметр NLS_LANG установлен некорректно, то поступающие в базу данные преобразуются неверно.

Например, предположим, что в базе используется набор символов AL32UTF8, клиент представляет собой английскую операционную систему Windows (кодовая страница: WE8MSWIN1252), а параметр клиента NLS_LANG равен AL32UTF8. Данные, поступающие в базу, кодируются в кодировке WE8MSWIN1252, но не преобразуются в данные AL32UTF8, так как параметр клиента NLS_LANG совпадает с набором символов базы данных. Таким образом, база данных Oracle предполагает, что преобразование не требуется, а в базу попадают недействительные данные.

Database Configuration Assistant (DBCA)



ORACLE

© Oracle, 2009. Все права защищены.

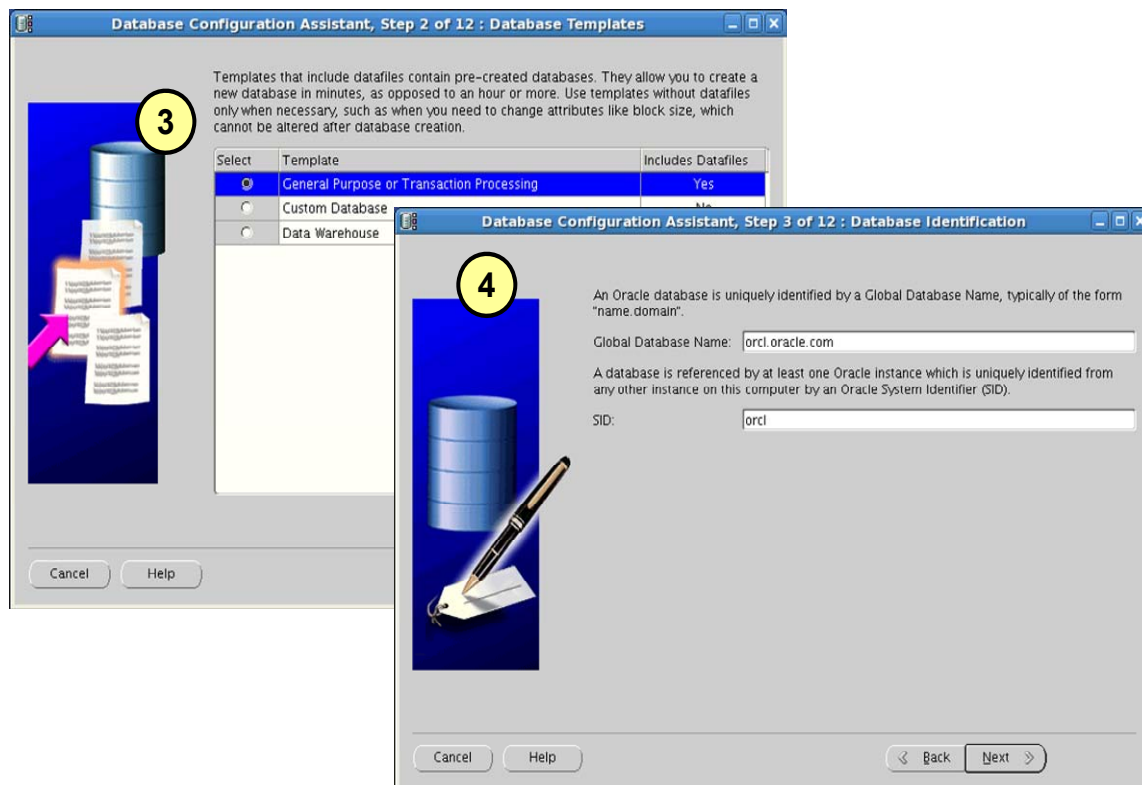
Database Configuration Assistant (DBCA)

Помощник Database Configuration Assistant (DBCA) предоставляет несколько функций, облегчающих выполнение различных действий. Помощник Database Configuration Assistant (DBCA) предназначен для создания и изменения конфигурации, а также для удаления базы данных. Также с его помощью можно создать базу данных из списка заранее определенных шаблонов или воспользоваться готовой базой данных как образцом для создания новой базы данных или шаблона.

DBCA предоставляет несколько возможностей по созданию базы данных, соответствующей определенным требованиям. DBCA содержит несколько страниц для ввода сведений конфигурации. На большинстве страниц DBCA предоставляются значения по умолчанию, которые можно принять и использовать. Чтобы создать базу данных с помощью DBCA, необходимо выполнить следующие действия:

1. Выполните вход в ОС вашего компьютера с правами участника группы DBA, у которого есть разрешение на установку ПО Oracle. При необходимости установите переменные среды и введите `dbca` для запуска DBCA. Отобразится основная страница приветствия. Нажмите кнопку Next ("Далее") для продолжения работы.
2. Выберите необходимое действие. В этом случае выберите команду Create a Database ("Создать базу данных") и нажмите кнопку Next ("Далее"), чтобы начать это действие.

Создание базы данных с помощью DBCA



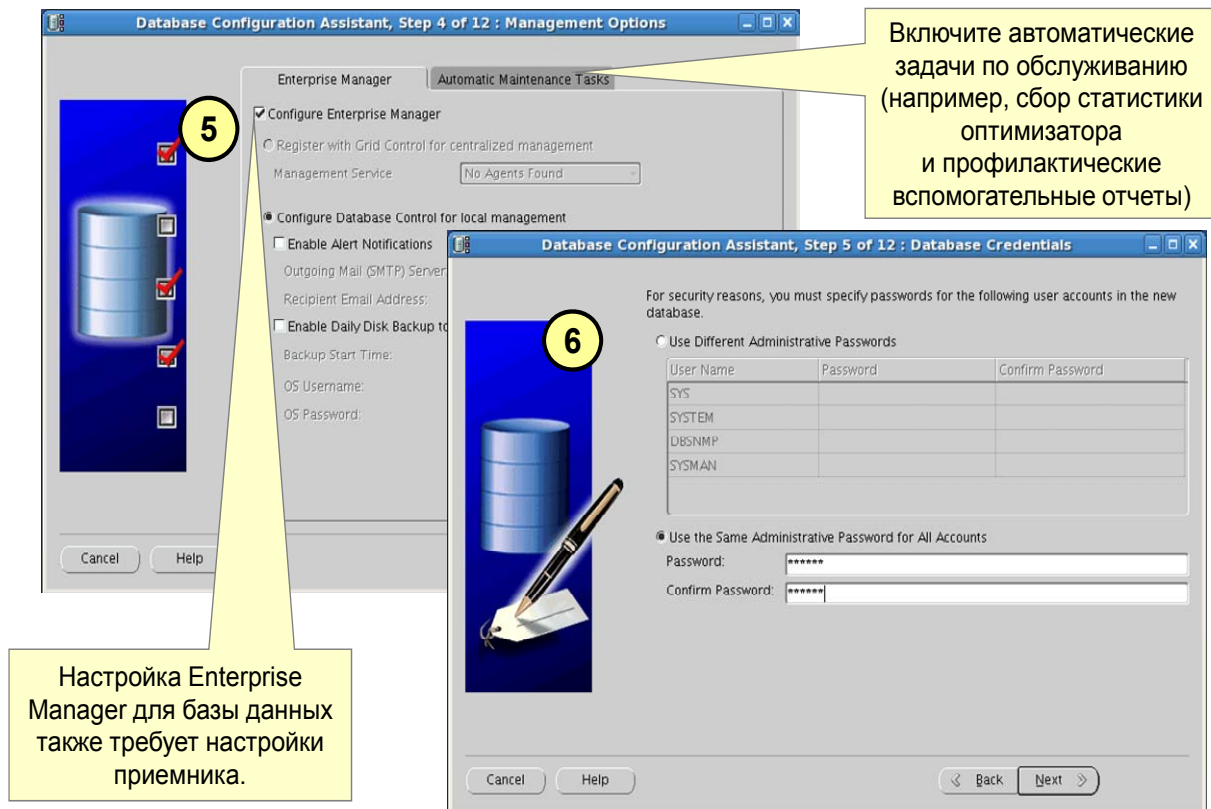
ORACLE

© Oracle, 2009. Все права защищены.

Создание базы данных с помощью DBCA

3. **Шаблоны баз данных:** Выберите тип шаблона базы данных, который будет использован. Доступны шаблоны баз данных Data Warehouse ("Хранилище данных"), General Purpose or Transaction Processing ("Общего назначения или обработка транзакций"), Custom Database ("Пользовательская база данных"). Эти шаблоны содержат предварительно настроенную базу данных с файлами данных. К этим файлам данных относятся управляющие файлы, файлы журнала повторов и файлы данных различных табличных пространств базы данных. Нажмите кнопку Show Details ("Показать подробную информацию"), чтобы просмотреть конфигурацию каждого типа базы данных.
В более сложных средах может потребоваться выбор параметра Custom Database ("Пользовательская база данных").
4. **Идентификация базы данных:** введите глобальное имя базы данных в формате имя_базы_данных.имя_домена, а также системный идентификатор (SID). SID по умолчанию имеет значение имени базы данных и уникальным образом определяет экземпляр, связанный с этой базой данных.

Создание базы данных с помощью DBCA



ORACLE

© Oracle, 2009. Все права защищены.

Создание базы данных с помощью DBCA (продолжение)

- 5. Параметры управления:** на этой странице можно настроить управление базой данных с помощью Oracle Enterprise Manager. Выберите значение по умолчанию: Configure the Database with Enterprise Manager ("Настроить базу данных с помощью Enterprise Manager"). Эта страница также позволяет настраивать параметры уведомлений и области ежедневного резервного копирования диска. Параметр Register with Grid Control ("Зарегистрировать в Grid Control") неактивна, если DBCA не удастся обнаружить доступный для коммуникации агент Enterprise Manager Grid Control.
Примечание. Для настройки Enterprise Manager в базе данных также требуется приемник (в данном примере приемник был настроен в ходе установки Oracle Grid Infrastructure).
- 6. Реквизиты базы данных:** на этой странице можно указать пароли для формуляров администратора, например SYS и SYSTEM. В учебном примере в качестве пароля для формуляров администратора используйте oracle_4U.

Создание базы данных с помощью DBCA

Выберите файловую систему или ASM (при наличии ASM)

При необходимости создайте несколько копий журналов повтора и управляющих файлов.

Select	Disk Group Name	Size (MB)	Free (MB)	Redundancy	State
☒	DATA	9216	3365	Normal	Mounted
☐	FRA	9216	9158	External	Mounted

Note: Free (MB) reflects the usable free space available taking redundancy into account.

OK Cancel

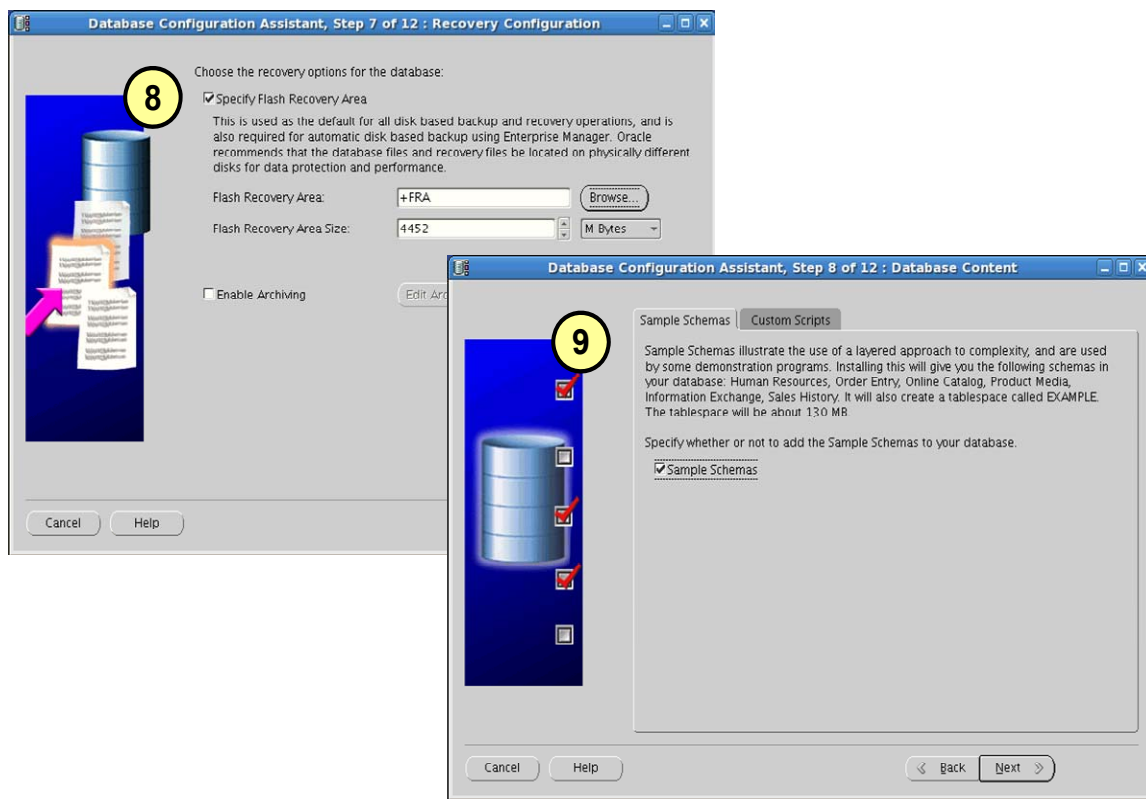
ORACLE

© Oracle, 2009. Все права защищены.

Создание базы данных с помощью DBCA (продолжение)

7. **Местоположение файлов БД:** укажите тип механизма хранения для базы данных (например, ASM или файловая система). Выберите расположение хранения в соответствии с требованиями. В данном примере механизмом хранения является ASM, что позволяет использовать управляемые файлы Oracle (OMF). Кнопка Browse ("Обзор") позволяет просматривать доступные группы дисков и выбирать наиболее подходящую группу (в примере на слайде выбрана группа +DATA). После выбора группы дисков потребуется указать пароль ASMSNMP, предназначенный для ASM. Управляемые файлы Oracle (OMF) могут также использоваться с системным хранилищем, что устраняет необходимость прямого управления файлами операционной системы в базе данных Oracle. При необходимости можно создать дополнительные копии журналов повторов и управляющих файлов, нажав кнопку Multiplex Redo Logs and Control Files ("Мультиплексирование журнала повторов и управляющих файлов"). Другая кнопка позволяет настроить переменные местоположения файлов: ORACLE_BASE, ORACLE_HOME, DB_NAME, DB_UNIQUE_NAME, SID.

Создание базы данных с помощью DBCA



ORACLE

© Oracle, 2009. Все права защищены.

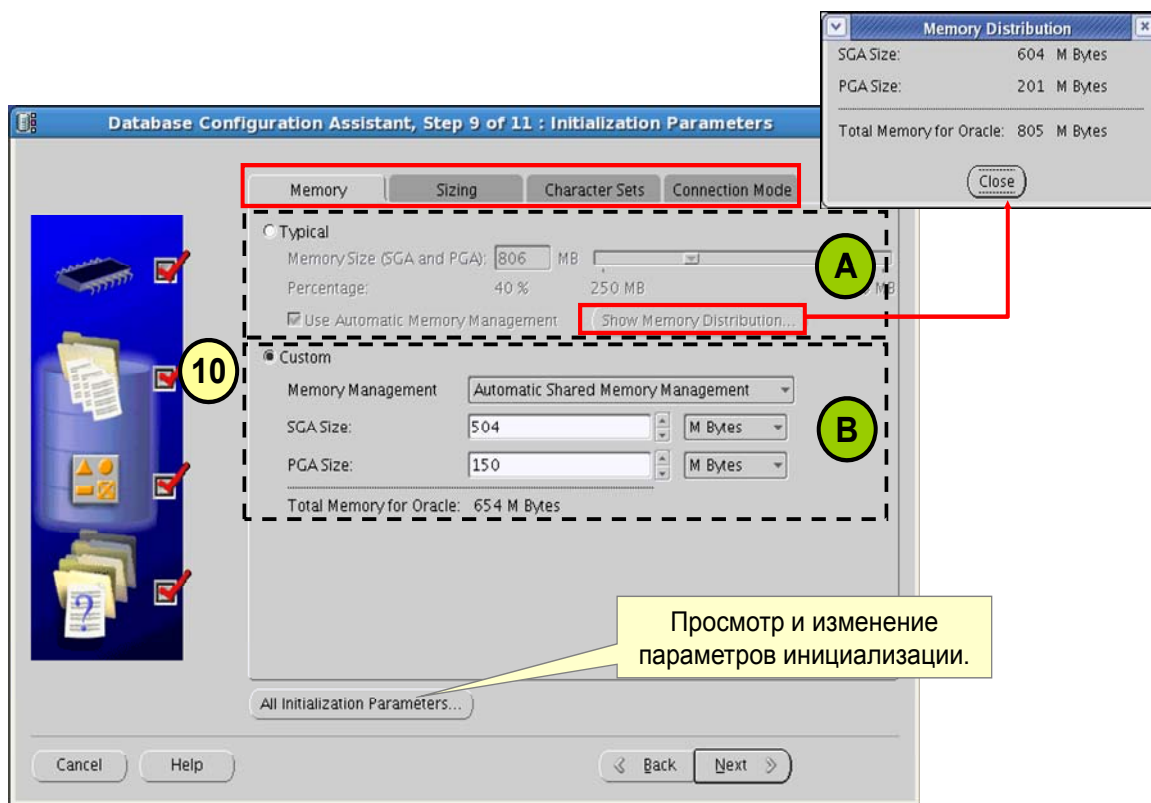
Создание базы данных с помощью DBCA (продолжение)

8. **Конфигурация восстановления:** выберите параметры восстановления базы данных. Чтобы настроить область мгновенного восстановления, установите параметр Specify Flash Recovery Area ("Задать область мгновенного восстановления"), а затем укажите размер этой области. В представленном примере расположение области мгновенного восстановления настроено в группе дисков ASM +FRA, а размер равен 4452 Мбайт. Рекомендуется установить размер области мгновенного восстановления, превышающий размер базы данных. Архивация и связанные с архивацией параметры также можно настраивать на этом экране.

Примечание. Название области мгновенного восстановления (flash recovery area) изменено на fast recovery area, но в других продуктах Oracle могут присутствовать ссылки на старое название.

9. **Содержимое базы данных:** эти страницы предоставляют возможность выбора компонентов (например, примеров схем), а также содержат вкладку для указания любых пользовательских сценариев, которые должны запускаться после создания базы данных.

Создание базы данных с помощью DBCA



ORACLE

© Oracle, 2009. Все права защищены.

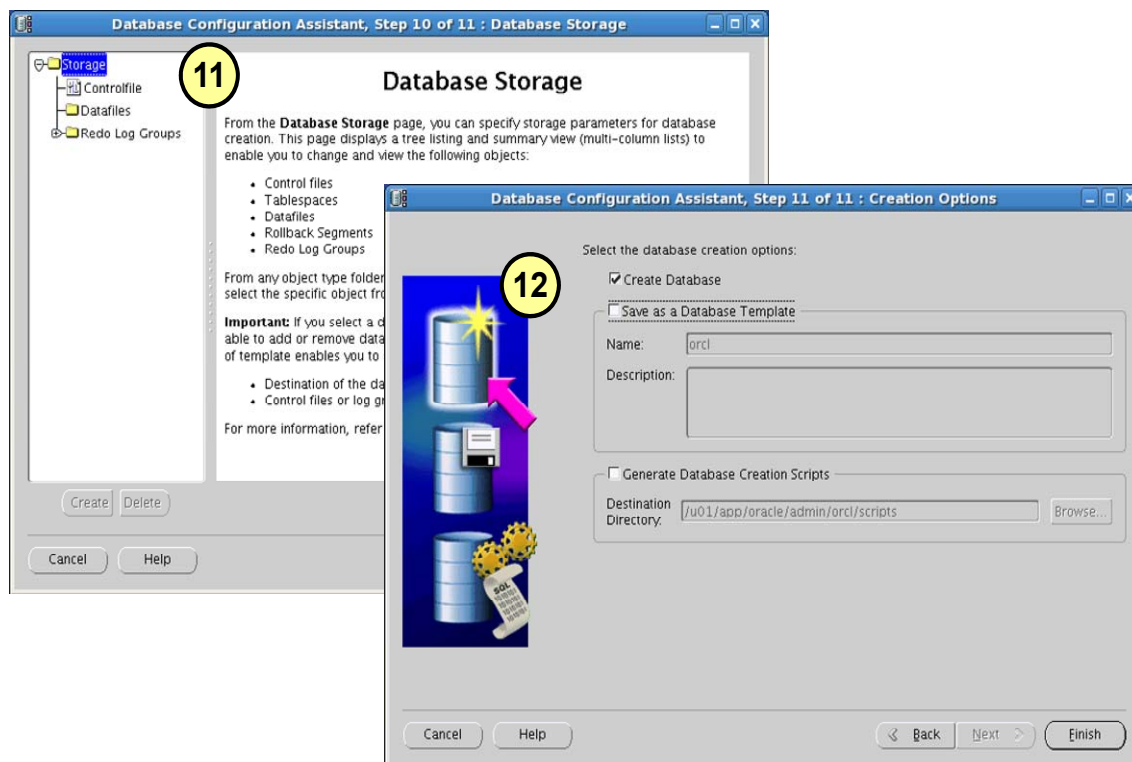
Создание базы данных с помощью DBCA (продолжение)

10. **Параметры инициализации:** с помощью закладок на данной странице можно изменить настройки базы данных по умолчанию:

- **Память:** на этой странице можно задать параметры инициализации, управляющие использованием памяти. Для выбора доступно (A) Typical ("Стандартное") или (B) Custom ("Нестандартное") выделение памяти.
- **Размеры:** укажите размер блока в байтах или примите стандартный размер.
- **Наборы символов:** на этой странице можно указать кодовые таблицы для базы данных.
- **Практические рекомендации Совет:** Корпорация Oracle рекомендует при возможности использовать в базе данных кодовую таблицу Юникод, так как она обеспечивает оптимальную гибкость для поддержки Web-технологий, а также поддерживает множество существующих языков.
- **Режим соединения:** выберите режим выделенного или разделяемого сервера. Подробные сведения содержатся в уроке "Настройка сетевой среды Oracle".
- Просмотрите и измените параметры инициализации, нажав кнопку All Initialization Parameters ("Все параметры инициализации").

Примечание. Некоторые параметры инициализации задаются на весь срок существования базы данных, например DB_BLOCK_SIZE.

Создание базы данных с помощью DBCA



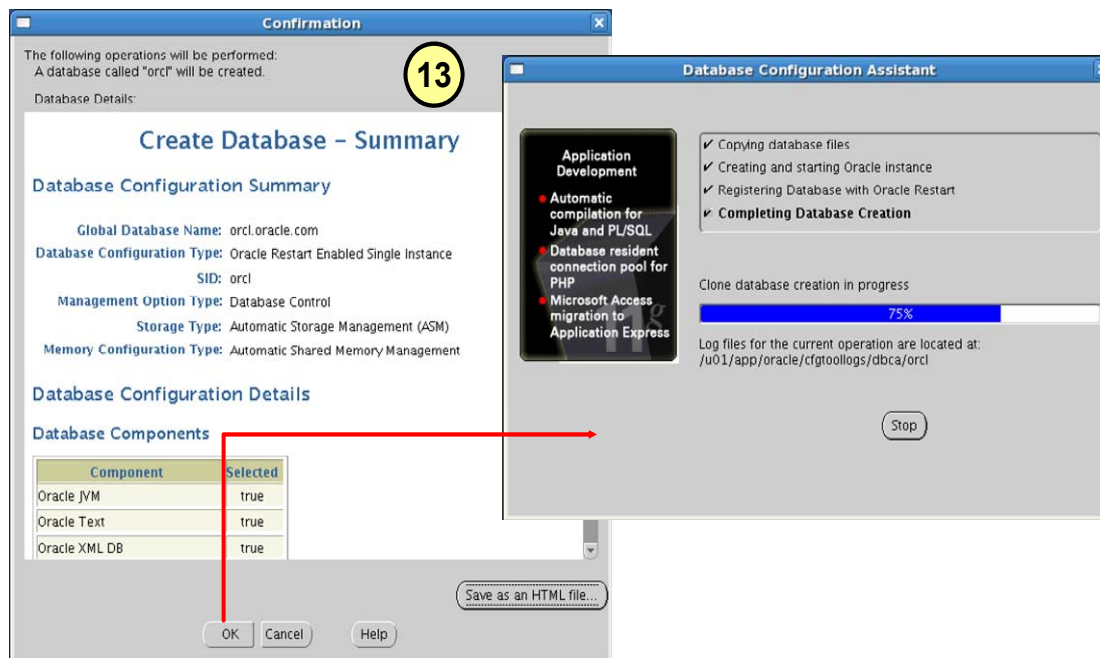
ORACLE

© Oracle, 2009. Все права защищены.

Создание базы данных с помощью DBCA (продолжение)

11. **Хранение базы данных:** просмотрите и при необходимости измените текущие параметры хранения базы данных. Если выбран один из предварительно настроенных шаблонов для базы данных, добавление или удаление управляющих файлов и файлов данных будет невозможным.
12. **Параметры создания:** создать базу данных, сохранить определение базы данных как шаблон и создать сценарии. Если выбраны все параметры, DBCA сначала сохранит шаблон базы данных, затем создаст сценарии в указанном каталоге, после чего создаст базу данных.

Сводка создания базы данных



ORACLE

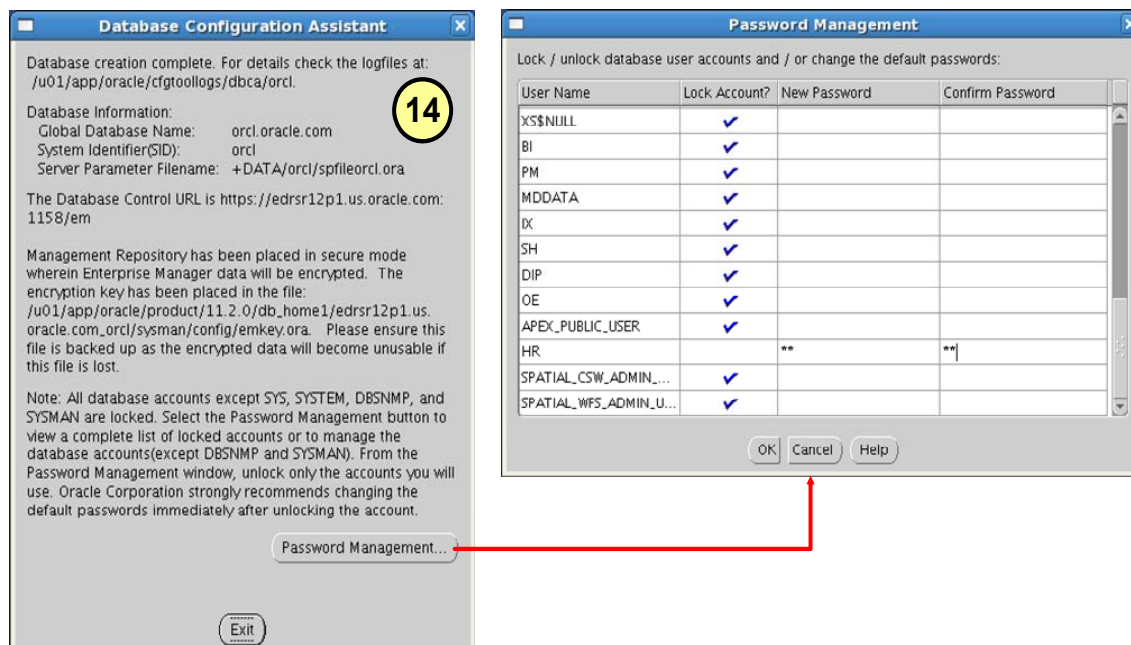
© Oracle, 2009. Все права защищены.

Сводка создания базы данных

13. После нажатия кнопки Finish ("Готово") отображается страница подтверждения, которая позволяет проверить конфигурацию базы данных перед созданием. Нажмите OK, чтобы закрыть страницу подтверждения и начать создание базы данных.

Примечание. В целях облегчения поиска можно сохранить определение базы данных в виде HTML-файла.

Управление паролями



ORACLE

© Oracle, 2009. Все права защищены.

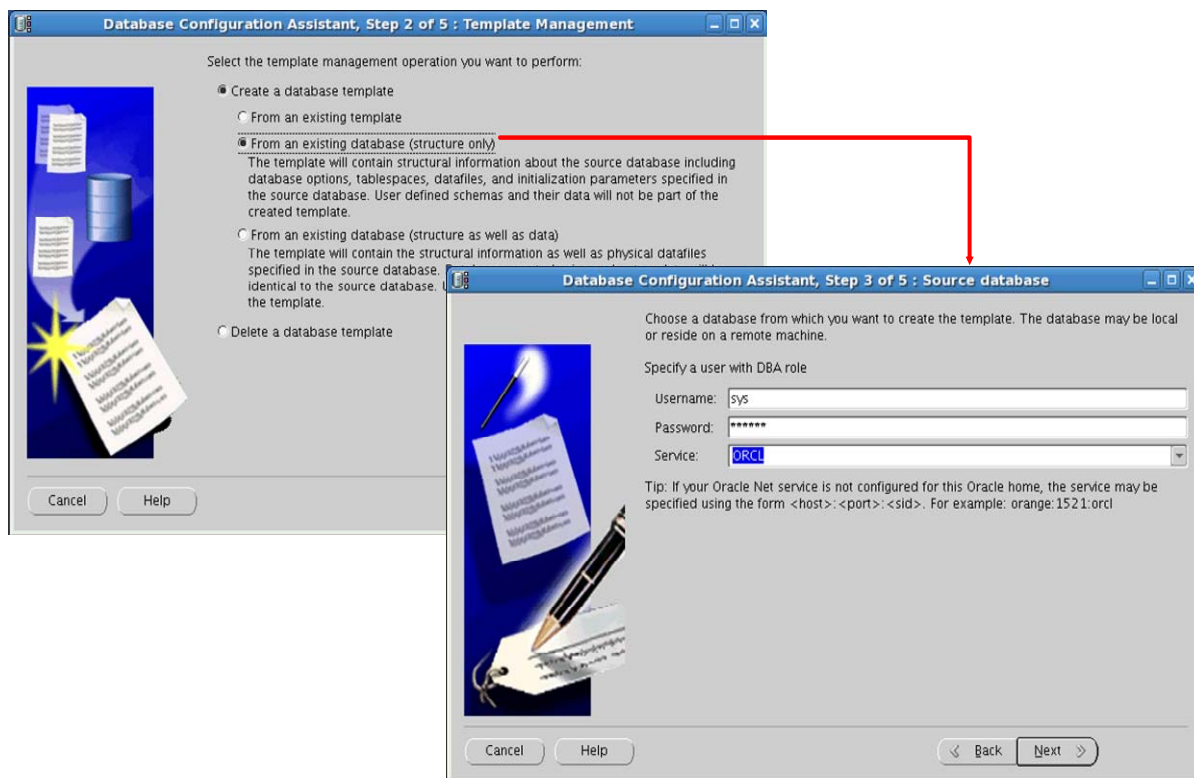
Управление паролями

14. После завершения работы DBCA обратите внимание на следующую справочную информацию:

- расположение файлов журнала установки;
- глобальное имя базы данных;
- системный идентификатор (SID);
- расположение и имя файла параметров сервера;
- URL-адрес Enterprise Manager.

Выберите пункт Password Management ("Управление паролями"), чтобы разблокировать учетные записи базы данных, которые планируется использовать. При разблокировании учетной записи необходимо ввести пароль. Неразблокированные учетные записи можно разблокировать впоследствии.

Создание шаблона структуры базы данных



ORACLE

© Oracle, 2009. Все права защищены.

Создание шаблона структуры базы данных

Шаблон – это заранее заданное определение базы данных, с помощью которого можно создать новую базу данных. Если шаблон не был создан при создании базы данных, его можно создать в любой момент с помощью команды DBCA Manage Templates ("Управление шаблонами").

Существует три способа создания шаблона:

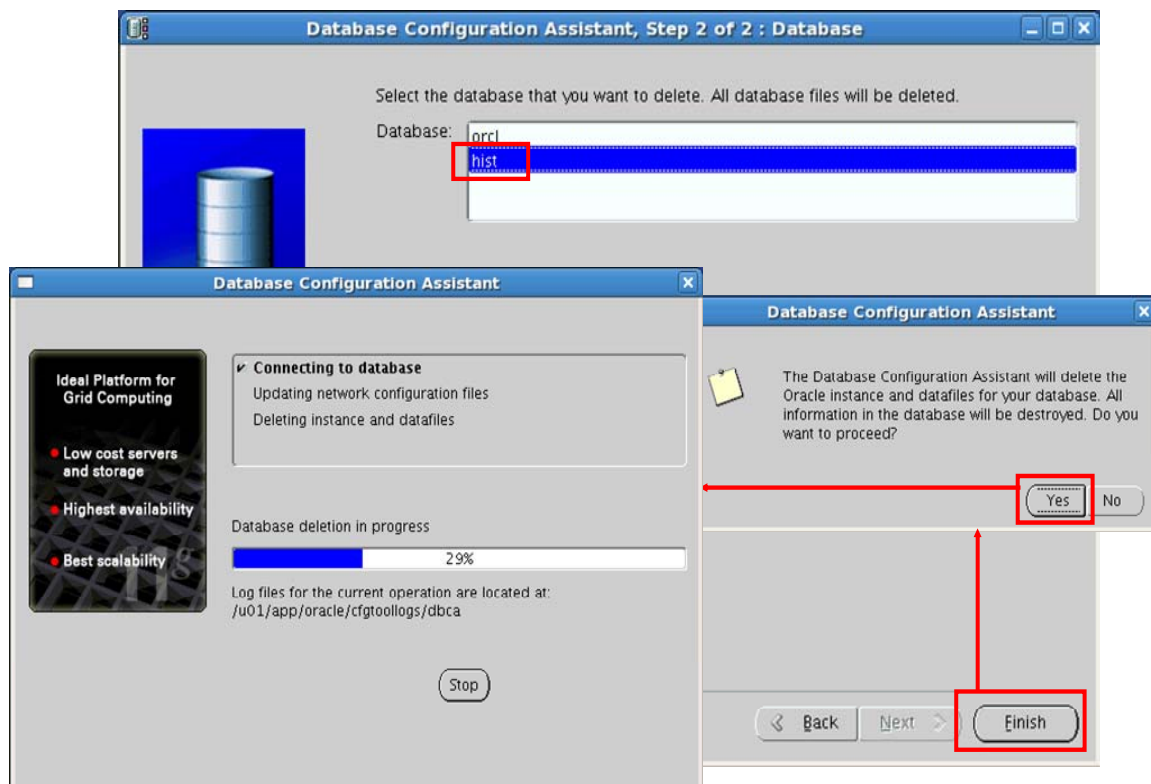
- с помощью существующего шаблона;
- с помощью существующей базы данных (только структура);
- с помощью существующей базы данных (структура и данные).

DBCA поможет выполнить все шаги создания шаблона структуры базы данных.

Если шаблон больше не требуется, используйте параметр Delete a database template ("Удалить шаблон базы данных") на странице DBCA Template Management ("Управление шаблонами").

Примечание. Созданные шаблоны отобразятся в списке Database Templates ("Шаблоны баз данных") при создании новой базы данных с помощью DBCA.

Удаление базы данных с помощью DBCA



ORACLE

© Oracle, 2009. Все права защищены.

Удаление базы данных с помощью DBCA

Для запуска DBCA введите в окне терминала `dbca` и нажмите кнопку Next ("Далее") на странице приветствия. Для удаления базы данных выполните следующие действия:

1. На странице Operations ("Операции") выберите пункт Delete a Database ("Удалить базу данных"). Затем нажмите кнопку Next ("Далее").
2. Выберите базу данных для удаления (в данном примере это БД `hist`), а затем нажмите кнопку Finish ("Готово").
3. Нажмите кнопку Yes ("Да"), чтобы подтвердить удаление.
4. После удаления будет выведен запрос на выполнение другого действия. Выберите необходимый вариант.

Примечание. Удаляемая база данных должна быть активной и работать, так как DBCA подключается к базе для определения сведений о местоположении файлов.

Удаление базы данных с помощью DBCA (продолжение)

При удалении базы данных также удаляются ее файлы данных, файлы журнала повторов, управляющие файлы и файлы параметров инициализации. Базу данных также можно удалить вручную с помощью оператора SQL `DROP DATABASE`. Оператор `DROP DATABASE` удаляет все управляющие файлы и все другие файлы базы данных, перечисленные в управляющем файле. Для успешного выполнения оператора `DROP DATABASE` необходимо соблюсти следующие условия:

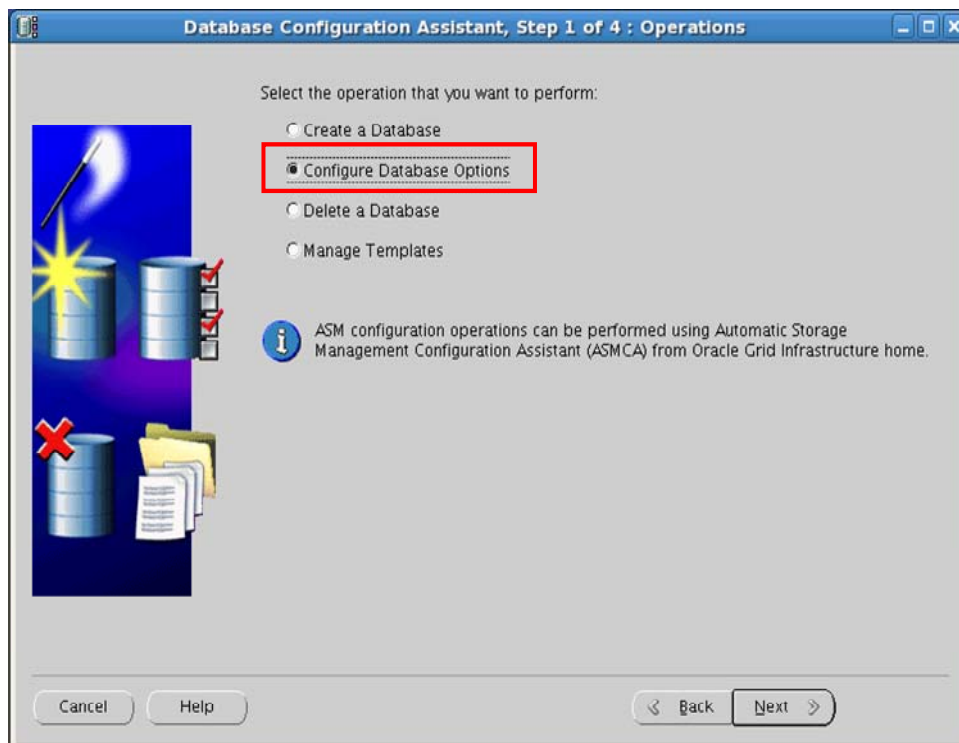
- база данных должна быть смонтирована и закрыта;
- база данных должна быть смонтирована с монопольным доступом (а не в разделяемом режиме);
- база данных должна быть запущена в режиме `RESTRICT`.

Примеры операторов:

```
STARTUP RESTRICT FORCE MOUNT;  
DROP DATABASE;
```

Оператор `DROP DATABASE` не затрагивает архивные файлы журнала, а также копии или резервные копии базы данных. Их лучше всего удалять с помощью диспетчера восстановления Recovery Manager (RMAN). Если база данных размещена на **неформатированных дисках, то специальные файлы самих дисков не удаляются.**

Выполнение дополнительных задач с помощью DBCA



ORACLE

© Oracle, 2009. Все права защищены.

Выполнение дополнительных задач с помощью DBCA

DBCA позволяет настраивать параметры баз данных (например, Oracle Label Security или Oracle Database Vault). Не все указанные параметры по умолчанию устанавливаются в ходе установки ПО баз данных, поэтому их необходимо установить перед попыткой настройки базы данных, использующей эти параметры.

Примечание

- Дополнительные сведения о Oracle Label Security см. в руководстве *Oracle Label Security Administrator's Guide*.
- Дополнительные сведения о Oracle Database Vault см. в руководстве *Oracle Database Vault Administrator's Guide*.

Экспресс-тест

Параметр DB_BLOCK_SIZE устанавливается на весь срок службы базы данных и не подлежит изменению.

1. Да
2. Нет

ORACLE

© Oracle, 2009. Все права защищены.

Ответ: 1

Экспресс-тест

Для удаления базы данных с помощью команды DROP DATABASE база данных должна быть в следующем состоянии:

1. открыта в режиме RESTRICT;
2. смонтирована исключительно в режиме RESTRICT;
3. закрыта наиболее быстрым способом.

ORACLE

© Oracle, 2009. Все права защищены.

Ответ: 2

Заключение

В ходе прохождения урока вы освоили следующие темы:

- создание баз данных с использованием помощника Database Configuration Assistant (DBCA);
- **генерация сценариев создания базы данных** с помощью DBCA;
- управление шаблонами баз данных с помощью DBCA;
- выполнение дополнительных задач с помощью DBCA.

ORACLE

© Oracle, 2009. Все права защищены.

Обзор упражнения 3: использование DBCA

В данном упражнении рассматриваются следующие темы:

- создание базы данных ORCL с помощью DBCA;
- разблокирование схемы HR;

Примечание. Создание базы данных и разблокирование схемы HR очень важны для выполнения последующих упражнений.

- создание шаблона структуры базы данных ORCL с помощью DBCA;
- создание сценариев для создания базы данных с помощью DBCA.

ORACLE

© Oracle, 2009. Все права защищены.

4

Управление экземпляром базы данных

ORACLE

© Oracle, 2009. Все права защищены.

Цели

Изучив материал этого занятия, вы освоите следующие темы:

- запуск и остановка базы данных Oracle и ее **компонентов**;
- использование Oracle Enterprise Manager;
- обращение к базе данных с помощью SQL*Plus;
- изменение параметров инициализации базы данных;
- описание этапов запуска базы данных;
- описание параметров остановки базы данных;
- просмотр журнала предупреждений;
- обращение к динамическим представлениям производительности.

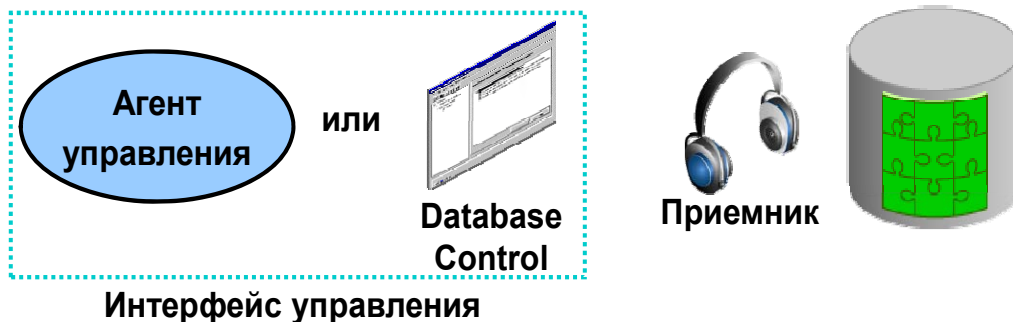
ORACLE

© Oracle, 2009. Все права защищены.

Структура управления

Компоненты структуры управления Oracle Database 11g версии 2:

- экземпляр базы данных;
- приемник;
- интерфейс управления:
 - Database Control
 - агент управления (при использовании компонента Grid Control).



ORACLE

© Oracle, 2009. Все права защищены.

Структура управления

Структура управления базой данных Oracle содержит три основных компонента.

- Экземпляр базы данных, который подлежит управлению.
- Приемник, который обеспечивает подключение к базе данных.
- Интерфейс управления. Им может выступать агент управления, расположенный на узле, где работает сервер базы данных (который подключается к Oracle Enterprise Manager Grid Control), или же отдельный модуль Oracle Enterprise Manager Database Control. Также его называют *консолью базы данных* (Database Console).

Для использования служб этих компонентов каждый из них должен быть запущен, а при остановке сервера базы данных Oracle следует полностью завершить работу компонентов.

Запуск и остановка средства управления базой данных Database Control

```
$ . oraenv
ORACLE_SID = [orcl] ? orcl
The Oracle base for ORACLE_HOME=/u01/app/oracle/product/11.2.0/db_home1
is /u01/app/oracle
$ emctl start dbconsole
Oracle Enterprise Manager 11g Database Control Release 11.2.0.1.0
Copyright (c) 1996, 2009 Oracle Corporation. All rights reserved.
http://host01.example.com:1158/em/console/aboutApplication
Starting Oracle Enterprise Manager 11g Database Control . . . . .started.
-----
Logs are generated in directory /u01/app/oracle/product/11.2.0/db_home1/
host01.example.com_orcl/sysman/ log
```

```
$ emctl start dbconsole
Oracle Enterprise Manager 11g Database Control Release 11.2.0.1.0
Copyright (c) 1996, 2009 Oracle Corporation. All rights reserved.
https://host01.example.com:1158/em/console/aboutApplication
Stopping Oracle Enterprise Manager 11g Database Control ...
... Stopped.
```

ORACLE

© Oracle, 2009. Все права защищены.

Запуск и остановка средства управления базой данных Database Control

База данных Oracle содержит средство управления базой данных *Database Control*, которое представляет собой консоль для управления базами данных, не подключенными к оболочке Grid Control. Для каждой базы данных, которая управляется средством Database Control, создается отдельная установка Database Control. Из любой установки Database Control можно управлять только одной базой данных. Перед использованием Database Control следует убедиться, что процесс `dbconsole` запущен.

Команда для запуска процесса `dbconsole`:

```
emctl start dbconsole
```

Команда для завершения процесса `dbconsole`:

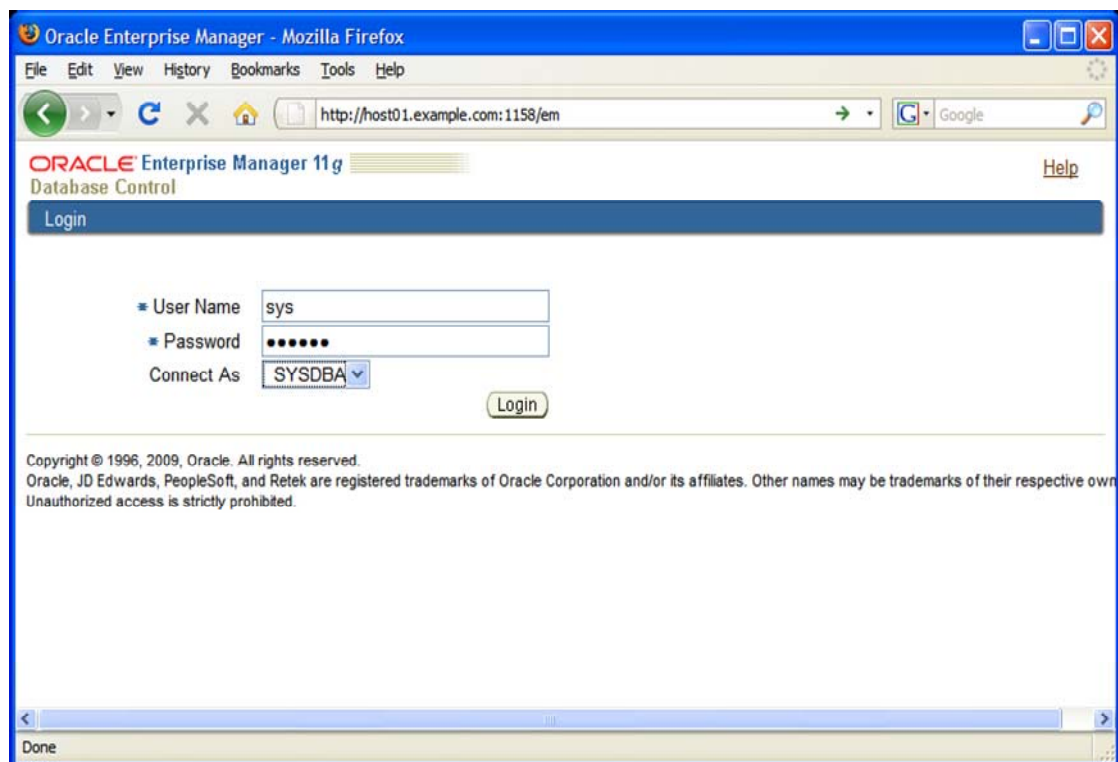
```
emctl stop dbconsole
```

Команда для просмотра состояния процесса `dbconsole`:

```
emctl start dbconsole
```

Примечание. Если каталог `$ORACLE_HOME/bin` не входит в путь операционной системы, к нему следует перейти самостоятельно. Если ПО Grid Infrastructure установлено, то существует два расположения `$ORACLE_HOME`, которые содержат служебную программу `emctl`. Служебная программа `emctl` всегда должна вызываться с помощью `$ORACLE_HOME` базы данных Oracle, а не `$ORACLE_HOME` Grid Infrastructure. Database Control использует процесс агента на стороне сервера. Процесс агента запускается и завершается автоматически при запуске или завершении процесса `dbconsole`.

Oracle Enterprise Manager



ORACLE

© Oracle, 2009. Все права защищены.

Oracle Enterprise Manager

При установке ПО баз данных Oracle Database средство Oracle Universal Installer (OUI) также устанавливает Oracle Enterprise Manager (Enterprise Manager). Его web-ориентированное средство управления базой данных Database Control является основным инструментом для управления базой данных Oracle. Enterprise Manager предоставляет графический интерфейс для решения практически всех задач, которые могут возникнуть перед администратором базы данных (DBA). Просмотр журнала предупреждений и графиков производительности, создание и изменение объектов, выполнение операций резервного копирования и восстановления – вот примеры операций, которые можно выполнять с помощью Enterprise Manager. В большинстве случаев в Enterprise Manager можно открывать ссылки для получения более подробных сведений о содержимом страницы.

Примечание. В Oracle Database 11g, версия 2 доступ к Enterprise Manager осуществляется по протоколу HTTPS (а не HTTP), что позволяет обеспечить защищенность соединения. Для обращения к Enterprise Manager dbconsole необходимо ввести URL в следующем формате:

`https://имя_компьютера:порт/em`

Для первой базы данных, созданной на компьютере, номер порта по умолчанию для доступа к Enterprise Manager Database Control равен 1158. Номер может быть другим, особенно в тех случаях, если один узел имеет несколько баз данных. Чтобы определить номер порта, обратитесь к файлу `portlist.ini`. Порты некоторых приложений базы данных Oracle Database указаны в файле `portlist.ini`, который находится в каталоге `$ORACLE_HOME/install`.

Oracle Enterprise Manager (продолжение)

При вводе URL-адреса Enterprise Manager отображаемое содержимое зависит от статуса базы данных:

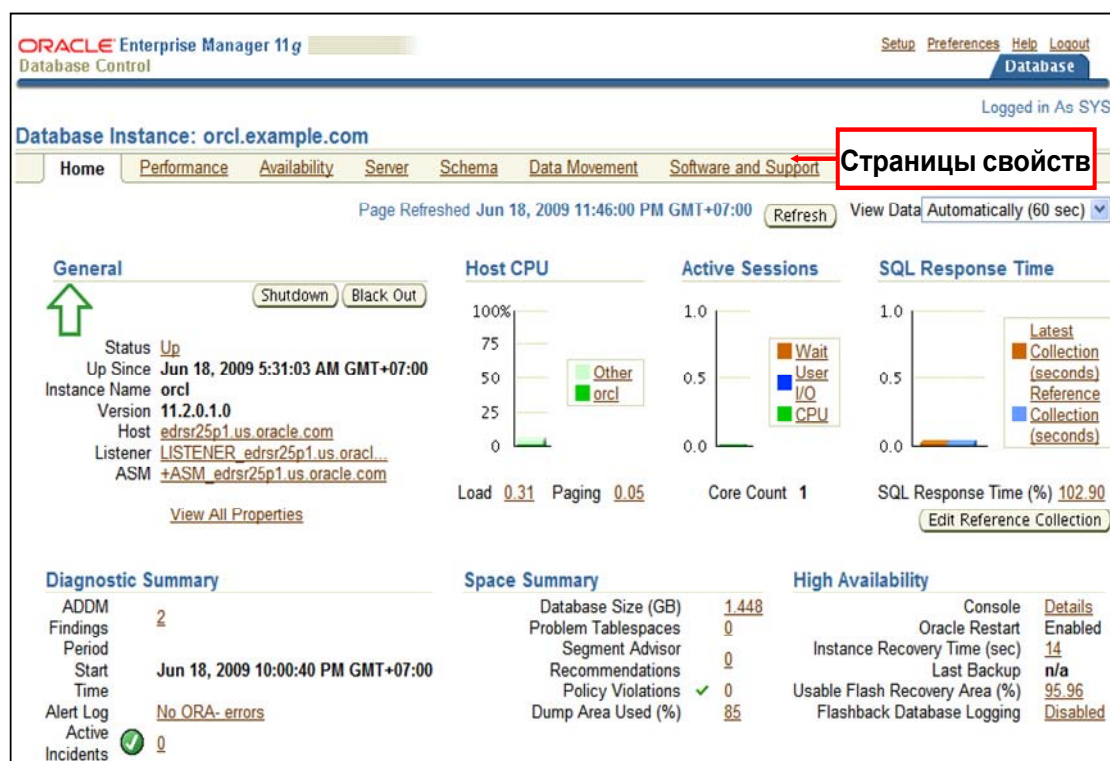
- Если база данных *запущена*, Enterprise Manager отобразит страницу входа в Database Control.

Подключитесь к базе данных с использованием имени пользователя, имеющего доступ к Database Control. Изначально такими пользователями являются SYS, SYSMAN и SYSTEM. Введите пароль, указанный для учетной записи во время установки базы данных. Для параметра Connect As ("Подключиться как") выберите имя пользователя Normal или SYSDBA, чтобы подключиться к базе данных с особыми полномочиями администратора.

- Если база данных *остановлена*, Enterprise Manager отобразит страницу Startup/Shutdown and Perform Recovery ("Запуск/остановка и выполнение восстановления"). В таком случае следует нажать кнопку Startup/Shutdown ("Запуск/остановка"). Вам будет предложено ввести узел, имена пользователей и пароли для входа в целевую базу данных.

Примечание. Если при запуске Enterprise Manager возникли проблемы, убедитесь в том, что прослушиватель запущен.

Домашняя страница базы данных



ORACLE

© Oracle, 2009. Все права защищены.

Домашняя страница базы данных

На домашней странице базы данных присутствуют показатели текущего состояния базы данных, отражающие ее общую работоспособность. На страницах свойств (которые также называют *вкладками*) можно получить доступ к страницам Performance ("Производительность"), Availability ("Доступность"), Server ("Сервер"), Schema ("Схема"), Data Movement ("Перемещение данных") и Software and Support ("Программное обеспечение и поддержка") для обслуживания базы данных.

На домашней странице базы данных можно ознакомиться с текущей производительностью и состоянием экземпляра базы данных:

- имя экземпляра, версия базы данных, домашний каталог Oracle, параметры восстановления носителей и другие сведения об экземпляре;
- доступность текущего экземпляра;
- важные предупреждения;
- сведения о производительности сеанса и SQL;
- ключевые показатели расходования пространства;
- ссылки для перехода вниз по иерархии (например, LISTENER_<имя_узла>) для получения дополнительных сведений.

Другие инструменты Oracle

Компоненты
> SQL*Plus
Параметры
инициализации
Запуск БД
Остановка БД
Журнал
предупреждений
Представления
производительности

- SQL*Plus предоставляет дополнительный интерфейс для работы с базой данных, с помощью которого можно выполнять следующие действия:
 - осуществлять обслуживание базы данных;
 - выполнять SQL-операторы для запроса, вставки, обновления и удаления данных из базы данных;
- SQL Developer:
 - представляет собой графический интерфейс пользователя для доступа к экземпляру базы данных Oracle;
 - поддерживает работу как с SQL, так и PL/SQL;
 - доступен в установке базы данных Oracle по умолчанию.

ORACLE

© Oracle, 2009. Все права защищены.

Другие инструменты Oracle

Помимо Enterprise Manager для выполнения SQL-операторов можно использовать SQL*Plus и SQL Developer. Эти инструменты позволяют выполнять множество операций по обслуживанию базы данных, а также выбирать, вставлять, обновлять или удалять данные из базы данных.

SQL*Plus – это приложение командной строки, которое используется для применения SQL- и PL/SQL-операторов к базе данных Oracle. Операторы можно вводить как интерактивно, так и с помощью сценариев SQL*Plus. SQL*Plus устанавливается вместе с базой данных и располагается в каталоге \$ORACLE_HOME/bin.

Программу SQL*Plus можно запустить из командной строки или меню "Пуск" клиента Windows.

SQL Developer представляет собой графический интерфейс пользователя для доступа к экземпляру базы данных Oracle. SQL Developer поддерживает языки SQL и PL/SQL. Он доступен в установке базы данных Oracle по умолчанию.

С помощью SQL Developer можно просматривать объекты базы данных, выполнять SQL-операторы и SQL-сценарии, а также редактировать и отлаживать операторы PL/SQL. Можно запускать произвольное количество готовых отчетов, а также создавать собственные.

Примечание. В данном курсе используются инструменты Enterprise Manager и SQL*Plus.

Использование SQL*Plus

SQL*Plus:

- представляет собой инструмент командной строки;
- используется интерактивно или в режиме пакетной обработки.

```
$ sqlplus hr

SQL*Plus: Release 11.2.0.1.0 - Production on Thu Jun 18 05:04:49 2009
Copyright (c) 1982, 2009, Oracle. All rights reserved.
Enter Password: *****

Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - Production
With the Partitioning, Automatic Storage Management, OLAP, Data Mining
and Real Application Testing options

SQL> select last_name from employees;
LAST_NAME
-----
Abel
Ande
...
```

ORACLE

© Oracle, 2009. Все права защищены.

Использование SQL*Plus

Интерфейс командной строки SQL*Plus можно использовать для выполнения команд SQL*Plus, SQL и PL/SQL в следующих целях:

- ввод, изменение, запуск, хранение, извлечение и сохранение SQL-команд и блоков PL/SQL;
- форматирование, вычисление, хранение и печать результатов выполнения запроса;
- отображение определений столбцов для любой таблицы;
- отправка сообщений конечным пользователям и получение ответов;
- администрирование базы данных.

Чтобы запустить SQL*Plus:

1. Откройте окно терминала.
2. В командной строке введите команду SQL*Plus в следующем формате:
`$ sqlplus <userid>/<pwd> or /nolog`
3. Если используется параметр **NOLOG**, следует ввести **CONNECT** и имя пользователя, от имени которого необходимо подключиться.
`SQL> connect <имя_пользователя>`
4. Введите пароль пользователя по запросу. SQL*Plus будет запущен и подключится к базе данных по умолчанию.

Вызов SQL*Plus из сценария оболочки

```
$ ./batch_sqlplus.sh
SQL*Plus: Release 11.2.0.1.0 - Production on Thu Jun 18 05:10:19 2009
Copyright (c) 1982, 2009, Oracle. All rights reserved.

Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - Production
With the Partitioning, Automatic Storage Management, OLAP, Data Mining
and Real Application Testing options

SQL>
COUNT(*)
-----
        107
SQL>
107 rows updated.
SQL>
Commit complete.
SQL> Disconnected from Oracle Database 11g Enterprise Edition Release
11.2.0.1.0 - Production
With the Partitioning, Automatic Storage Management, OLAP, Data Mining
and Real Application Testing options
$
```

Результат

```
# Name of this file: batch_sqlplus.sh
# Count employees and give raise.
sqlplus hr/hr <<EOF
select count(*) from employees;
update employees set salary = salary*1.10;
commit;
quit
EOF
```

ORACLE

© Oracle, 2009. Все права защищены.

Вызов SQL*Plus из сценария оболочки

SQL*Plus можно вызвать из сценария оболочки или BAT-файла, указав команду `sqlplus` и задействовав механизм сценариев операционной системы для передачи параметров.

В приведенном примере команды `SELECT`, `UPATE` и `COMMIT` будут выполнены до того, как SQL*Plus вернет управление операционной системе.

Вызов SQL-сценария из SQL*Plus

script.sql

```
select * from departments where location_id = 1400;  
quit
```

Результат

```
$ sqlplus hr/hr @script.sql  
  
SQL*Plus: Release 11.2.0.1.0 - Production on Thu Jun 18 05:13:42 2009  
Copyright (c) 1982, 2009, Oracle. All rights reserved.  
  
Connected to:  
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - Production  
With the Partitioning, Automatic Storage Management, OLAP, Data Mining  
and Real Application Testing options  
  
DEPARTMENT_ID DEPARTMENT_NAME          MANAGER_ID LOCATION_ID  
-----  
              60 IT                      103        1400  
  
Disconnected from Oracle Database 11g Enterprise Edition Release  
11.2.0.1.0 - Production  
With the Partitioning, Automatic Storage Management, OLAP, Data Mining  
and Real Application Testing options
```

ORACLE

© Oracle, 2009. Все права защищены.

Вызов SQL-сценария из SQL*Plus

Существующий файл SQL-сценария можно открыть из SQL*Plus. Это можно сделать в командной строке, предварительно вызвав SQL*Plus, как показано на слайде. Также это можно сделать из сеанса SQL*Plus с помощью оператора @. Ниже приведен пример запуска сценария из установленного сеанса SQL*Plus:

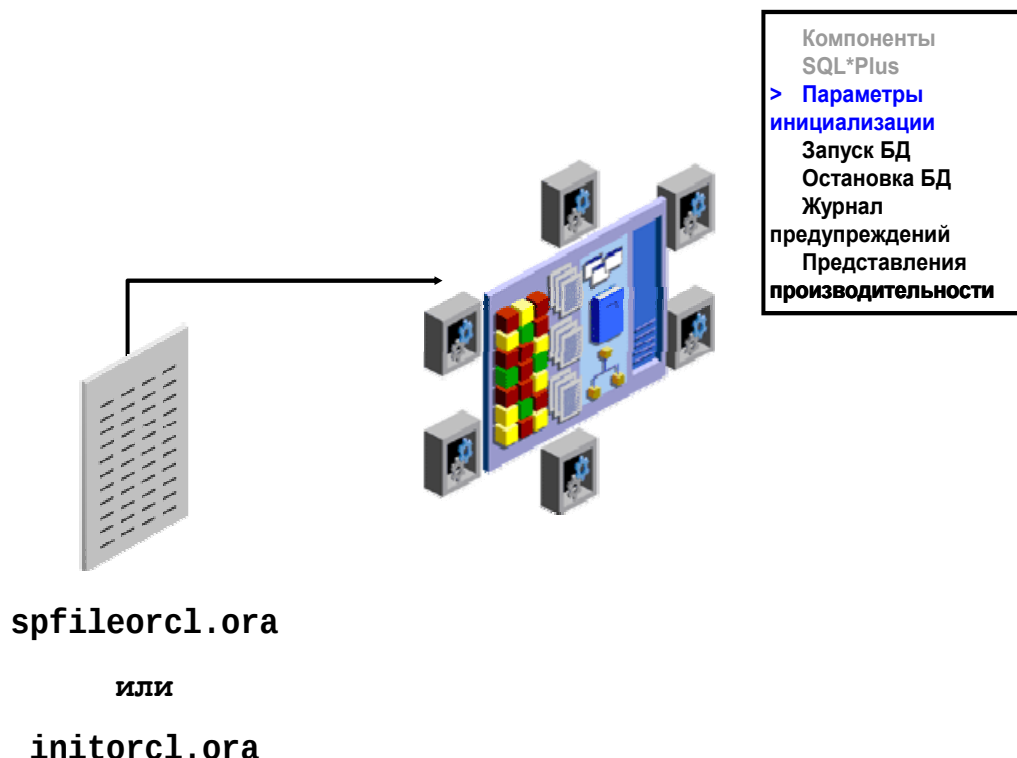
```
SQL> @script.sql
```

Примечание. Расширением по умолчанию для файлов сценариев является .sql.

При сохранении сценария из SQL*Plus с помощью команды save файлу автоматически будет присвоено данное расширение. Сценарии с таким расширением можно выполнять, не указывая расширение файла, как в следующем примере:

```
SQL> @script
```

Файлы параметров инициализации



ORACLE

© Oracle, 2009. Все права защищены.

Файлы параметров инициализации

При запуске экземпляра осуществляется считывание файла параметров инициализации. Существуют два типа файлов параметров.

- **Файлы параметров сервера (SPFILE):** этот тип файла параметров инициализации является предпочтительным. Это двоичный файл, чтение и запись в который осуществляет сервер базы данных. *Самостоятельно изменять этот файл нельзя.* Он расположен на сервере, на котором запущен экземпляр Oracle, и сохраняется вне зависимости от ее остановки или запуска. Именем файла по умолчанию, по которому он проверяется при запуске, является `spfile<SID>.ora`.
- **Текстовый файл параметров инициализации:** этот тип файла параметров инициализации может быть считан сервером БД, но не записывается сервером. Настройки параметров инициализации необходимо задавать и изменять вручную с помощью текстового редактора. Данный файл также хранится независимо от запуска и остановки базы данных. Именем файла по умолчанию (которое будет автоматически использоваться при запуске в случае отсутствия SPFILE) является `init<SID>.ora`.

Рекомендуется создать и использовать SPFILE, так как это позволит динамически обслуживать параметры инициализации.

Примечание. База данных Oracle выполняет поиск файлов инициализации в каталоге Linux `$ORACLE_HOME/dbs`. При наличии ASM файл SPFILE часто располагается в группе дисков ASM. В этом случае файл `init<SID>.ora` с указанием расположения SPFILE должен находиться в каталоге `$ORACLE_HOME/dbs`.

Файлы параметров инициализации (продолжение)

Типы значений параметров инициализации

Сервер базы данных Oracle работает со следующими типами значений параметров инициализации:

- логическое значение
- строка
- целое число
- файл параметров;
- зарезервированное значение
- большое целое число

Производные значения параметров

Некоторые параметры инициализации являются производными, то есть их значения вычисляются из значений других параметров. Обычно изменение значений производных параметров не требуется.

При их изменении указанное значение переопределит вычисленное значение.

Например, по умолчанию значение параметра `SESSIONS` вычисляется из значения параметра `PROCESSES`. Если значение параметра `PROCESSES` изменится, значение по умолчанию параметра `SESSIONS` тоже изменится, если вы не присвоили ему определенное значение вручную.

Зависимые от операционной системы значения параметров

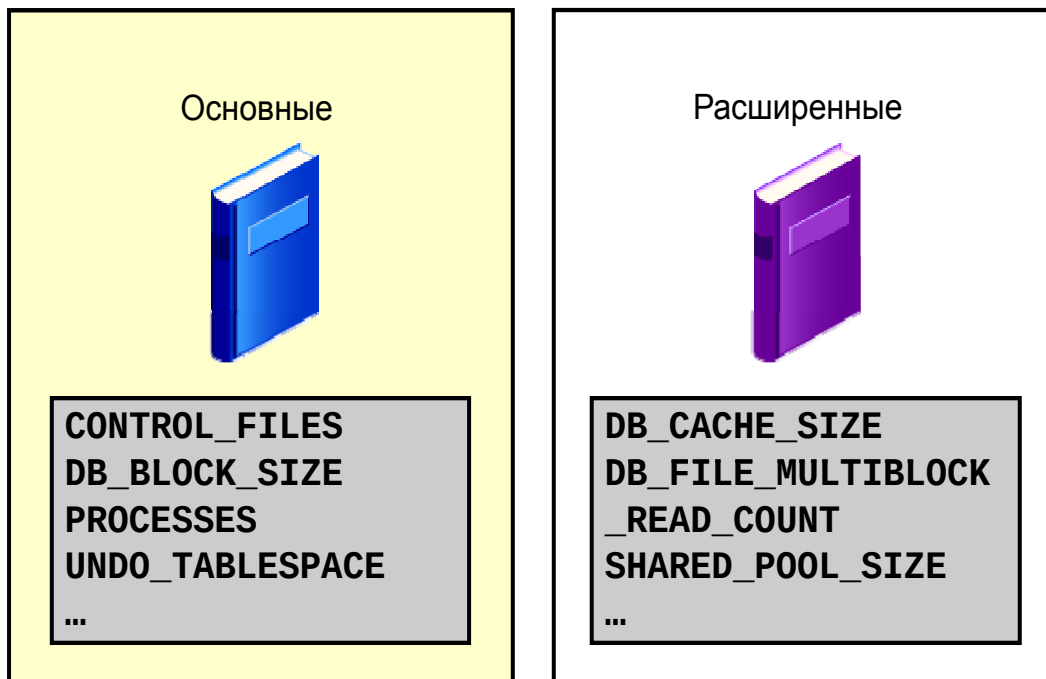
Допустимые значения или диапазоны значений некоторых параметров инициализации зависят от операционной системы хоста. Например, параметр `DB_FILE_MULTIBLOCK_READ_COUNT` определяет максимальное количество блоков, считываемых за одну операцию ввода/вывода при последовательном сканировании. Значение этого параметра будет зависеть от текущей платформы. Размер этих блоков определяется параметром `DB_BLOCK_SIZE`, значение по умолчанию которого также зависит от операционной системы.

Присвоение значений параметрам

Наибольшим образом на производительность системы могут влиять параметры инициализации. Некоторые из них могут накладывать ограничения на потребление пространства, но при этом они не будут влиять на производительность. Например, если параметру `OPEN_CURSORS` присвоено значение `10`, то попытка пользовательского процесса открыть одиннадцатый курсор приведет к ошибке. Другие параметры могут влиять на производительность, но не на общую работоспособность системы. Например, уменьшение значения параметра `OPEN_CURSORS` может снизить производительность, но не остановить работу системы.

Увеличение значений параметров может повысить производительность системы, однако для большинства параметров это приведет к увеличению размера глобальной системной области (SGA). SGA большего размера может позволить достичь пиковой производительности базы данных. SGA слишком большого размера в операционных системах с виртуальной памятью может снизить производительность из-за интенсивной работы с файлом подкачки. Присваивать значения параметрам операционной системы, которые управляют виртуальной памятью, следует с учетом размера SGA. Конфигурация операционной системы также может ограничивать максимальный размер SGA.

Упрощенные параметры инициализации



ORACLE

© Oracle, 2009. Все права защищены.

Упрощенные параметры инициализации

Параметры инициализации делятся на два типа: основные и расширенные.

В большинстве случаев для получения приемлемого уровня производительности базы данных следует установить и настроить около 30 основных параметров. В редких случаях для достижения оптимальной производительности может возникнуть необходимость в настройке расширенных параметров. Существует около 314 расширенных параметров.

Основные параметры предназначены для самостоятельной настройки, чтобы поддерживать высокую производительность базы данных. Все остальные параметры относятся к расширенным.

Примеры основных параметров:

- Определение глобального имени базы данных: DB_NAME и DB_DOMAIN
- Указание области мгновенного восстановления и ее размера: DB_RECOVERY_FILE_DEST и DB_RECOVERY_FILE_DEST_SIZE
- Указание общего размера всех компонентов SGA: SGA_TARGET
- Указание метода табличного пространства управления пространством отмены: UNDO_TABLESPACE
- параметр инициализации COMPATIBLE и необратимая совместимость

Примечание. Некоторые параметры инициализации перечислены на последующих страницах. Для получения полного списка параметров обратитесь к руководству *Oracle Database Reference*.

Параметры инициализации: примеры

Параметр	Определяет
CONTROL_FILES	Имена управляющих файлов
DB_FILES	Максимальное количество файлов базы данных
PROCESSES	Максимальное количество одновременно подключенных пользовательских процессов
DB_BLOCK_SIZE	Стандартный размер блока базы данных для всех табличных пространств
DB_CACHE_SIZE	Размер стандартного блока буферного кэша

ORACLE

© Oracle, 2009. Все права защищены.

Параметры инициализации: примеры

Параметр CONTROL_FILES: указывает имена управляющих файлов. Oracle настоятельно рекомендует создавать мультиплексированные и зеркальные копии управляющих файлов. Диапазон значений: от одного до восьми имен файлов (с путями). Значение по умолчанию: зависит от ОС.

Параметр DB_FILES: определяет максимальное количество файлов базы данных, которые можно открыть в базе данных. Диапазон значений: зависит от ОС. Значение по умолчанию: 200.

Параметр PROCESSES: указывает максимальное количество пользовательских процессов ОС, которые могут быть одновременно подключены к серверу Oracle. Это значение влияет на все фоновые процессы и процессы пользователя. Диапазон значений: от 6 и до значения, зависящего от ОС. Значение по умолчанию: 100.

Параметр DB_BLOCK_SIZE: указывает размер (в байтах) блока базы данных Oracle. Это значение задается при создании базы данных и впоследствии не может быть изменено. Данным параметром определяется стандартный размер блока базы данных. Этот размер будет по умолчанию использоваться во всех табличных пространствах. Диапазон значений: от 2048 до 32768 (зависит от ОС). Значение по умолчанию: 8192.

Параметр DB_CACHE_SIZE: определяет размер стандартного блока буферного кэша. Диапазон значений: как минимум 16 Мбайт. Значение по умолчанию: 0 при установленном значении SGA_TARGET, в противном случае наибольшее значение из 48 Мбайт или (4 Мбайт*sru_count).

Параметры инициализации: примеры



ORACLE

© Oracle, 2009. Все права защищены.

Параметры инициализации: примеры (продолжение)

Параметр SGA_TARGET определяет общий размер всех компонентов SGA. Если параметр SGA_TARGET указан, то размер приведенных ниже пулов будет определен автоматически:

- буферный кэш (DB_CACHE_SIZE);
- разделяемый пул (SHARED_POOL_SIZE);
- большой пул (LARGE_POOL_SIZE);
- пул Java (JAVA_POOL_SIZE);
- пул потоков (STREAMS_POOL_SIZE).

Если данным пулам памяти, которые настраиваются автоматически, присвоены ненулевые значения, данные значения будут использоваться системой автоматического управления совместно используемой памятью Automatic Shared Memory Management (ASMM) в качестве минимальных уровней. Минимальные значения задаются в том случае, если компоненту приложения для нормальной работы требуется определенный минимальный объем памяти. Размер приведенных ниже пулов настраивается вручную и не зависит от ASMM:

- буфер журнала;
- другие буферные кэши (такие как KEEP и RECYCLE), а также блоки других размеров;
- фиксированная SGA и другие внутренние выделения памяти.

Память, выделенная этим пулам, вычитается из общего объема доступной памяти для SGA_TARGET при включенной системе ASMM.

Примечание. Процесс MMON рассчитывает значения автоматически настроенных пулов памяти для поддержки ASMM.

Параметры инициализации: примеры (продолжение)

MEMORY_TARGET определяет общий объем доступной в системе Oracle памяти.

База данных настраивает память согласно значению параметра **MEMORY_TARGET**, уменьшая и увеличивая размеры SGA и PGA по необходимости.

Если в текстовом файле параметров инициализации не указан параметр **MEMORY_MAX_TARGET**, но указано значение для параметра **MEMORY_TARGET**, то база данных автоматически присвоит параметру **MEMORY_MAX_TARGET** значение параметра **MEMORY_TARGET**. Если не указан параметр **MEMORY_TARGET** и при этом указано значение для параметра **MEMORY_MAX_TARGET**, параметру **MEMORY_TARGET** по умолчанию будет присвоено нулевое значение. После запуска можно динамически присваивать параметру **MEMORY_TARGET** ненулевое значение, если оно не превышает значения параметра **MEMORY_MAX_TARGET**. Значение параметра **MEMORY_TARGET** можно изменить с помощью команды **ALTER SYSTEM**. Значения могут находиться в диапазоне от 152 Мбайт до значения параметра **MEMORY_MAX_TARGET**.

Параметры инициализации: примеры

Параметр	Определяет
PGA_AGGREGATE_TARGET	Объем памяти PGA, выделенной всем серверным процессам
SHARED_POOL_SIZE	Размер разделяемого пула (в байтах)
UNDO_MANAGEMENT	Использование управления пространством отмены операций

ORACLE

© Oracle, 2009. Все права защищены.

Параметры инициализации: примеры (продолжение)

Параметр PGA_AGGREGATE_TARGET: определяет объем памяти глобальной программной области (PGA), выделяемой всем связанным с экземпляром серверным процессам. Данная область памяти размещается не в глобальной системной области (SGA). База данных использует объем памяти PGA, указанный данным параметром, в качестве целевого значения. При установке данного параметра объем памяти, занимаемый SGA, будет вычтен из общей памяти системы, доступной экземпляру Oracle. Диапазон значений состоит из целых чисел и букв K, M или G (килобайты, мегабайты или гигабайты). Минимальным значением является 10 Мбайт, а максимальным – (4096 Гбайт - 1). Размер по умолчанию равен 10 Мбайт или 20% от размера SGA (в зависимости от того, что больше).

Параметр SHARED_POOL_SIZE: определяет размер разделяемого пула в байтах.

Разделяемый пул содержит такие объекты, как разделяемые курсоры, хранимые процедуры, управляющие структуры и буферы параллельной обработки сообщений. Диапазон значений: зависит от ОС. Значение по умолчанию: 0, если установлено значение SGA_TARGET, в противном случае 128 Мбайт для 64-битной среды и 48 Мбайт для 32-битной среды.

Параметр UNDO_MANAGEMENT: определяет режим управления пространством отмены операций, который будет использовать система. Если параметру присвоено значение AUTO, то экземпляр будет запущен в автоматическом режиме Automatic Undo Management (AUM). В противном же случае экземпляр будет запущен в режиме Rollback Undo (RBU). В режиме RBU пространство отмены операций выделяется внешним образом как сегменты отката. В режиме AUM пространство отмены операций выделяется внешним образом как табличные пространства отмены операций. Диапазон значений: AUTO или MANUAL. Если параметр UNDO_MANAGEMENT опущен, то при первом запуске экземпляра по умолчанию параметру будет присвоено значение AUTO.

Использование SQL*Plus для отображения параметров

```
SQL> SELECT name , value FROM V$PARAMETER;
```

NAME	VALUE
lock_name_space	2
processes	150
sessions	247
timed_statistics	TRUE
timed_os_statistics	0
...	


```
SQL>SHOW PARAMETER SHARED_POOL_SIZE
```

NAME	TYPE	VALUE
shared_pool_size	big integer	0


```
SQL> show parameter para
```

NAME	TYPE	VALUE
fast_start_parallel_rollback	string	LOW
parallel_adaptive_multi_user	boolean	TRUE
parallel_automatic_tuning	boolean	FALSE
parallel_execution_message_size	integer	16384
parallel_instance_group	string	
...		

ORACLE

© Oracle, 2009. Все права защищены.

Использование SQL*Plus для отображения параметров

На этом слайде представлены примеры использования SQL*Plus для просмотра параметров. Чтобы просмотреть значения различных параметров, можно опросить представление словаря данных V\$PARAMETER. Представление V\$PARAMETER отображает текущие значения параметров для текущего сеанса. Также можно воспользоваться командой SHOW PARAMETER с указанием строки для просмотра параметров, содержащих эту строку.

В нижеприведенном запросе иллюстрируется запрос на отображение параметров и их значений. С помощью фразы WHERE можно указать конкретные параметры:

```
SQL> SELECT name, value FROM V$PARAMETER WHERE name LIKE '%pool%';
```

NAME	VALUE
shared_pool_size	0
large_pool_size	0
java_pool_size	0
streams_pool_size	0
shared_pool_reserved_size	8808038
buffer_pool_keep	
...	

9 rows selected.

Использование SQL*Plus для отображения параметров (продолжение)

Описание представления:

```
SQL> desc V$parameter
```

Name	Null?	Type
NUM		NUMBER
NAME		VARCHAR2(80)
TYPE		NUMBER
VALUE		VARCHAR2(4000)
DISPLAY_VALUE		VARCHAR2(4000)
ISDEFAULT		VARCHAR2(9)
ISSES_MODIFIABLE		VARCHAR2(5)
ISSYS_MODIFIABLE		VARCHAR2(9)
ISINSTANCE_MODIFIABLE		VARCHAR2(5)
ISMODIFIED		VARCHAR2(10)
ISADJUSTED		VARCHAR2(5)
ISDEPRECATED		VARCHAR2(5)
ISBASIC		VARCHAR2(5)
DESCRIPTION		VARCHAR2(255)
UPDATE_COMMENT		VARCHAR2(255)
HASH		NUMBER

Во втором примере иллюстрируется использование команды SQL*Plus **SHOW PARAMETER** для просмотра значений параметров. Данную команду также можно использовать для поиска всех параметров, которые содержат текстовую строку. Например, с помощью приведенной ниже команды можно найти все имена параметров, содержащие строку **db**:

```
SQL> show parameter db
```

NAME	TYPE	VALUE
...		
db_8k_cache_size	big integer	0
db_block_buffers	integer	0
db_block_checking	string	FALSE
db_block_checksum	string	TYPICAL
db_block_size	integer	8192
db_cache_advice	string	ON
db_cache_size	big integer	0
...		

Другие представления, содержащие сведения о параметрах

- **V\$SPPARAMETER**: отображает сведения о содержимом файла параметров сервера. Если файл параметров сервера не использовался для запуска экземпляра, то каждая строка представления будет содержать значение **FALSE** в столбце **ISSPECIFIED**.
- **V\$PARAMETER2**: отображает сведения о параметрах инициализации, влияющие на текущий сеанс, и их значениях. Новый сеанс наследует значения параметров, указанные для экземпляра в представлении **V\$SYSTEM_PARAMETER2**.
- **V\$SYSTEM_PARAMETER**: отображает сведения о параметрах инициализации, которые в настоящий момент действуют для экземпляра.

Изменение значений параметров инициализации

- Статические параметры:
 - могут изменяться только в файле параметров;
 - чтобы новое значение вступило в силу, необходимо выполнить перезапуск экземпляра;
 - существуют примерно для 110 параметров.
- Динамические параметры:
 - могут изменяться при работающей базе данных;
 - могут быть изменены на:
 - уровне сеанса;
 - уровне системы;
 - действительны на протяжении всего сеанса или согласно настройке SCOPE;
 - могут быть изменены с помощью команд ALTER SESSION и ALTER SYSTEM;
 - существуют примерно для 234 параметров.

ORACLE

© Oracle, 2009. Все права защищены.

Изменение значений параметров инициализации

Существует два типа файлов параметров инициализации.

Статические параметры: влияют на экземпляр или на всю базу данных и могут изменяться только путем изменения содержимого файла `init.ora` или файла `SPFILE`. Чтобы изменения статических параметров вступили в силу, необходимо остановить и перезапустить базу данных. Данные параметры нельзя настроить только для текущего экземпляра.

Динамические параметры: могут изменяться при работающей базе данных. Существует два типа динамических параметров:

- *параметры уровня сеанса* оказывают влияние только на текущий сеанс пользователя. Примерами могут выступать параметры поддержки национальных языков (NLS), которые позволяют указать настройки национального языка для операций сортировки, параметры даты и т.д. Данные параметры можно использовать в течение сеанса; после его завершения они также прекратят свое существование.
- *параметры уровня системы* оказывают влияние на всю базу данных и все сеансы. Примерами могут выступать параметры, изменяющие значение `SGA_TARGET`, а также параметры расположения архивного журнала. Действие данных параметров зависит от настройки `SCOPE`. Чтобы сделать эти параметры постоянными, необходимо добавить их в файл `SPFILE` путем указания параметра `SCOPE=both` или вручную изменив файл `PFILE`.

Динамические параметры можно изменять с помощью команд `ALTER SESSION` и `ALTER SYSTEM`.

Изменение значений параметров инициализации (продолжение)

С помощью фразы SET оператора ALTER SYSTEM можно задать или изменить значения параметров инициализации. Необязательный параметр SCOPE определяет область действия изменений:

- **SCOPE=SPFILE:** изменения вносятся только в файл параметров сервера. К текущему экземпляру изменения не применяются. Для динамических и статических параметров изменения станут постоянными и вступят в силу при следующем запуске. Для статических параметров это единственное допустимое значение параметра SCOPE.
- **SCOPE=MEMORY:** изменения применяются только в памяти. Изменения применяются к текущему экземпляру и вступают в силу незамедлительно. Для динамических параметров эффект будет моментальным, но не постоянным, так как файл параметров сервера обновлен не будет. Для статических параметров это значение недопустимо.
- **SCOPE=BOTH:** изменения вносятся и в файл параметров сервера, и в памяти. Изменения применяются к текущему экземпляру и вступают в силу незамедлительно. Для динамических параметров эффект также будет постоянным, так как будет обновлен файл параметров сервера. Для статических параметров это значение недопустимо.

Нельзя указывать значения SCOPE=SPFILE или SCOPE=BOTH в том случае, если экземпляр не был запущен с использованием файла параметров сервера. По умолчанию используется значение SCOPE=BOTH, если файл параметров сервера был использован для запуска экземпляра. Если же для этого использовался текстовый файл параметров инициализации, по умолчанию будет использовано значение MEMORY.

Для динамических параметров также можно указать ключевое слово DEFERRED. В этом случае изменения вступят в силу только для последующих сеансов. Это справедливо только для следующих параметров:

- backup_tape_io_slaves
- recyclebin
- audit_file_dest
- object_cache_optimal_size
- object_cache_max_size_percent
- sort_area_size
- sort_area_retained_size
- olap_page_pool_size

При присвоении параметру SCOPE значения SPFILE или BOTH необязательный параметр COMMENT позволит связать значение параметра с текстовой строкой. Указанный комментарий будет записан в файле параметров сервера.

Изменение значений параметров: примеры

```
SQL> ALTER SESSION  
      SET NLS_DATE_FORMAT = 'mon dd yyyy';
```

Session altered.

```
SQL> SELECT SYSDATE FROM dual;
```

SYSDATE

jun 18 2009

```
SQL> ALTER SYSTEM SET  
SEC_MAX_FAILED_LOGIN_ATTEMPTS=2 COMMENT='Reduce  
from 10 for tighter security.' SCOPE=SPFILE;
```

System altered.

ORACLE

© Oracle, 2009. Все права защищены.

Изменение значений параметров: примеры

Первый оператор слайда иллюстрирует изменение параметра уровня сеанса. Пользователь устанавливает следующий формат даты для сеанса: `mon dd yyyy`. В результате этого действия все запросы, использующие даты, будут отображать их в указанном формате. Параметры уровня сеанса можно легко изменять из приложений с помощью PL/SQL.

Второй оператор изменяет максимальное количество неудачных попыток регистрации, после которых соединение будет прервано. Также в примере присутствует комментарий и указано, что изменения следует внести только в файл параметров сервера. После накопления указанного количества неудачных попыток регистрации соединение будет автоматически прервано серверным процессом. Это не динамический параметр, поэтому для применения изменений потребуется перезапустить экземпляр базы данных Oracle.

Экспресс-тест

Enterprise Manager Database Control можно использовать для одновременного управления несколькими базами данных.

1. Да
2. Нет

ORACLE

© Oracle, 2009. Все права защищены.

Ответ: 2

Экспресс-тест

Большинство параметров базы данных являются динамическими и могут быть изменены без необходимости отключения экземпляра базы данных.

1. Да
2. Нет

ORACLE

© Oracle, 2009. Все права защищены.

Ответ: 1

Запуск и остановка базы данных. Учетные данные

Компоненты
SQL*Plus
Параметры
инициализации
> **Запуск БД**
Остановка БД
Журнал
предупреждений
Представления
производительности

The screenshot shows the 'Startup/Shutdown: Specify Host and Target Database Credentials' dialog box. It is titled 'ORACLE Enterprise Manager 11g Database Control'. The 'Database Instance' is 'orcl.example.com'. The user is logged in as 'SYS'. The dialog has two main sections: 'Host Credentials' and 'Database Credentials'. In 'Host Credentials', the 'Username' is 'oracle' and the 'Password' is masked with dots. In 'Database Credentials', the 'Username' is 'sys', the 'Password' is masked, the 'Database' is 'orcl.example.com', and the 'Connect As' is 'SYSDBA'. There is a 'Save as Preferred Credential' checkbox which is unchecked. A note at the bottom states: 'Note that you need to login to the database as SYSDBA or SYSOPER in order to change the status of the database.' There are 'Cancel' and 'OK' buttons at the bottom right. A green circle with the number '2' is placed over the 'OK' button.

The diagram shows two tabs: 'General' and 'Database Instance'. The 'General' tab has a green upward arrow and a 'Shutdown' button. The 'Database Instance' tab has a red downward arrow and a 'Startup' button. Between the tabs, the text 'или' (or) is written vertically. A green circle with the number '1' is placed next to the 'или' text.

ORACLE

© Oracle, 2009. Все права защищены.

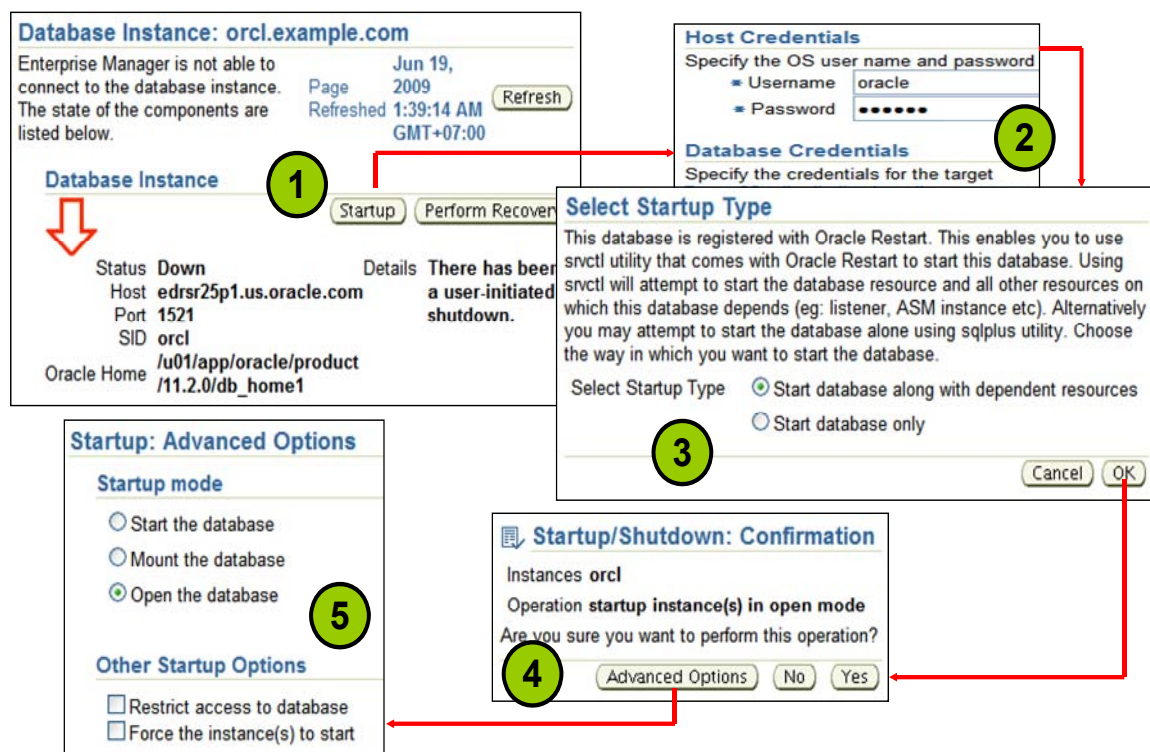
Запуск и остановка базы данных: учетные данные

При нажатии кнопки Startup ("Запуск") или Shutdown ("Остановка") вам будет предложено ввести свои реквизиты для подключения к хосту (компьютеру, на котором расположена база данных) и базе данных. Необходимо ввести учетную запись базы данных, которая имеет привилегии SYSDBA. Введите учетные данные.

После ввода учетных данных будет выведен запрос на ввод метода запуска или остановки. После этого можно нажать кнопку Advanced Options ("Дополнительные параметры") и при необходимости изменить параметры запуска или режим остановки. Также можно нажать кнопку Show SQL ("Показать SQL"), чтобы просмотреть операторы SQL, используемые для запуска или остановки базы данных.

Примечание. Для остановки базы данных с помощью Enterprise Manager по умолчанию используется параметр IMMEDIATE. При выполнении команды SHUTDOWN из SQL*Plus по умолчанию используется параметр NORMAL.

Запуск экземпляра базы данных Oracle



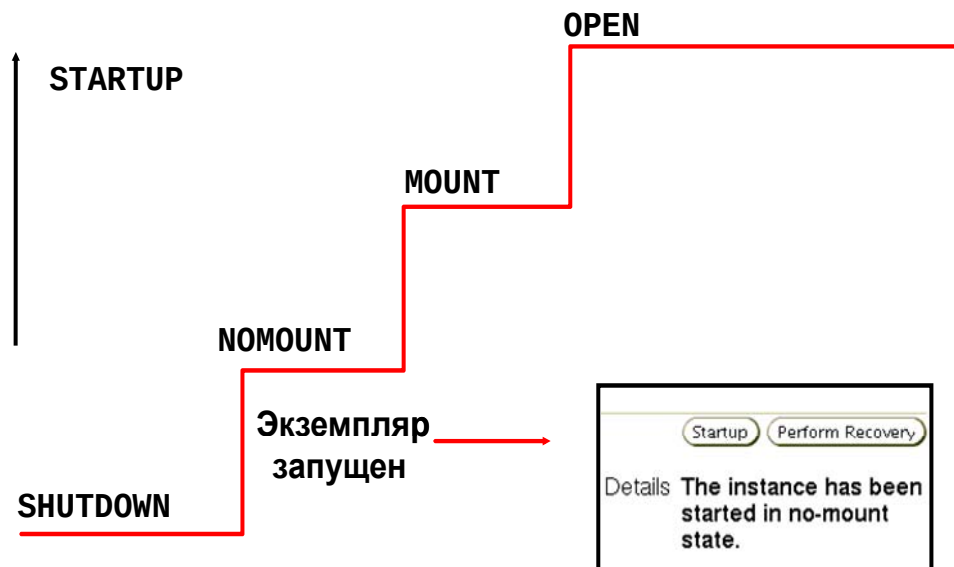
ORACLE

© Oracle, 2009. Все права защищены.

Запуск экземпляра базы данных Oracle

Если база данных не запущена в момент перехода на страницу управления базой данных в Enterprise Manager, нажмите кнопку Startup ("Запуск"). После этого введите реквизиты хоста и при необходимости укажите режим запуска. Если база данных Oracle зарегистрирована в Oracle Restart, то необходимо выбрать в отдельном диалоговом окне средство для запуска экземпляра базы данных: Server Control (SRVCTL) или SQL*Plus. Служебная программа SRVCTL является рекомендованной для Oracle Restart, так как предоставляет возможность запускать зависимые ресурсы.

Запуск экземпляра базы данных Oracle: NOMOUNT



ORACLE

© Oracle, 2009. Все права защищены.

Запуск экземпляра базы данных Oracle: NOMOUNT

При запуске экземпляра базы данных укажите состояние, в котором его следует запустить. Указанные ниже сценарии описывают различные этапы запуска экземпляра.

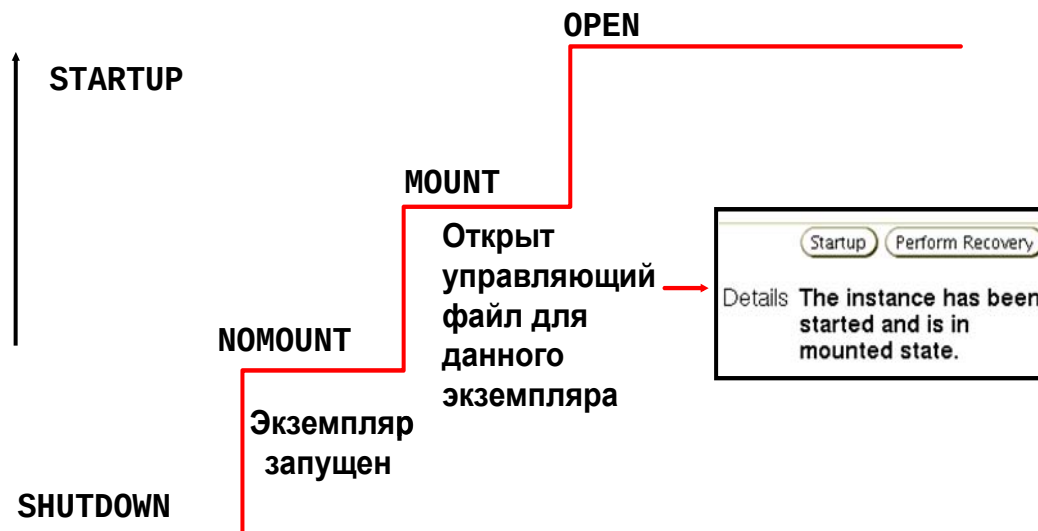
Экземпляр обычно запускается в режиме **NOMOUNT** только при создании базы данных, при повторном создании управляющих файлов или при выполнении определенных сценариев резервного копирования и восстановления.

При запуске экземпляра выполняются следующие действия:

- Поиск файла с определенным именем в `$ORACLE_HOME/dbs` в особой последовательности:
 1. Поиск `spfile<SID>.ora`.
 2. Если файл `spfile<SID>.ora` не найден, осуществляется поиск файла `spfile.ora`.
Если файл `spfile.ora` не найден, осуществляется поиск файла `init<SID>.ora`.
В этом файле хранятся параметры инициализации экземпляра. Установка для параметра `PFILE` значения `STARTUP` переопределяет поведение по умолчанию.
- Выделение SGA
- Запуск фоновых процессов
- Открытие файла `alert_<SID>.log` и файлов трассировки

Примечание. `SID` – это системный идентификатор, который описывает экземпляр (например `ORCL`).

Запуск экземпляра базы данных Oracle: MOUNT



ORACLE

© Oracle, 2009. Все права защищены.

Запуск экземпляра базы данных Oracle:MOUNT

При монтировании базы данных выполняются следующие операции:

- база данных ассоциируется с ранее запущенным экземпляром;
- выполняется поиск и открытие управляющих файлов, указанных в файле параметров;
- производится чтение управляющих файлов для получения сведений об именах и состоянии файлов данных и интерактивных файлах журналов повтора (однако при этом не выполняются проверки на наличие самих файлов данных и интерактивных файлов журналов повтора).

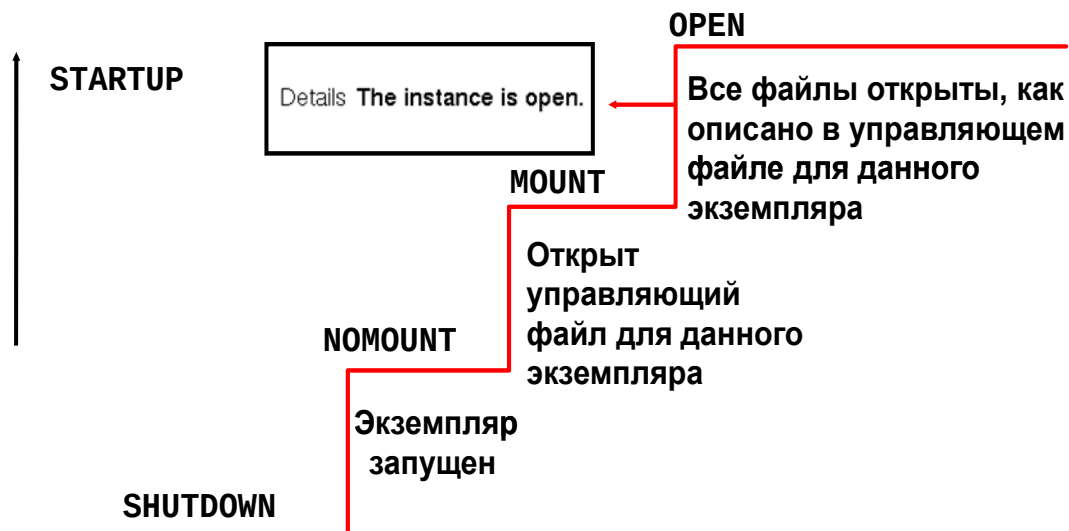
Для выполнения некоторых операций обслуживания следует запустить экземпляр, смонтировать базу данных, но не открывать ее.

Например, базу данных следует смонтировать, но не открывать для выполнения следующих задач:

- переименование файлов данных (файлы данных для табличного пространства, находящегося в автономном режиме, можно переименовывать и при открытой базе данных);
- включение и отключение параметров архивации оперативного файла журнала повторов;
- выполнение полного восстановления базы данных.

Примечание. База данных может оставаться в режиме **MOUNT** даже при поступлении запроса на переход в режим **OPEN**. Причиной этому может быть необходимость в восстановлении базы данных тем или иным способом. Если операция восстановления выполняется в режиме **MOUNT**, журналы повторов будут открыты для чтения. Также будут открыты файлы данных для чтения блоков, необходимых для восстановления, и для записи блоков, если это требуется для восстановления.

Запуск экземпляра базы данных Oracle: OPEN



ORACLE

© Oracle, 2009. Все права защищены.

Запуск экземпляра базы данных Oracle: OPEN

Для нормально функционирующей базы данных подразумевается, что экземпляр был запущен, а база данных смонтирована и открыта. К корректно функционирующей базе данных может подключиться любой действующий пользователь и выполнить с данными стандартные операции.

При открытии базы данных выполняются следующие операции:

- открытие файлов данных;
- открытие оперативных файлов журнала повторов.

Если при открытии базы данных отсутствует любой из файлов данных или оперативных **файлов журнала повторов, сервер Oracle возвращает ошибку.**

На данном заключительном этапе сервер Oracle удостоверяется, что все файлы данных и оперативные файлы журнала повторов могут быть открыты, а также выполняет проверку целостности базы данных. При необходимости фоновый процесс системного монитора (SMON) инициирует операцию восстановления экземпляра.

Экземпляр базы данных можно запустить в ограниченном режиме, в котором доступ к экземпляру имеют только пользователи с полномочиями администратора. Чтобы запустить экземпляр в ограниченном режиме, выберите параметр Restrict access to database ("Ограничить доступ к базе данных") на странице расширенных параметров запуска.

Параметры запуска: примеры

- Использование служебной программы sqlplus:

```
SQL> startup
```

1

```
SQL> startup nomount
```

2

```
SQL> alter database mount;
```

3

```
SQL> alter database mount;
```

4

- Использование служебной программы srvctl с Oracle Restart

```
$ srvctl start database -d orcl -o mount
```

ORACLE

© Oracle, 2009. Все права защищены.

Параметры запуска: примеры

На слайде отображен синтаксис SQL*Plus для запуска базы данных.

- Данная команда запускает экземпляр, ассоциирует с ним файлы базы данных, а затем монтирует и открывает базу данных.
- Данная команда запускает экземпляр, при этом база данных не смонтирована.
- Данная команда монтирует базу данных из состояния **NOMOUNT**.
- Данная команда открывает базу данных из состояния **MOUNT**.

Если база данных включена в Oracle Restart, то служебная программа **srvctl** позволяет запустить экземпляр базы данных. Преимуществом программы **srvctl** является возможность запуска всех необходимых зависимых ресурсов (например, экземпляра ASM, групп дисков ASM и приемника).

Примечание. Программа **srvctl** находится в каталоге **\$ORACLE_HOME/bin** для ПО Grid infrastructure и каталоге **\$ORACLE_HOME/bin** для ПО базы данных Oracle. При запуске базы данных Oracle необходимо использовать программу **srvctl** из ПО баз данных Oracle. При запуске экземпляра ASM или приемника следует использовать программу **srvctl** из ПО Grid Infrastructure.

Остановка экземпляра базы данных Oracle

The screenshot illustrates the steps to shut down an Oracle database instance. It shows four numbered callouts: 1. The 'Shutdown' button in the General tab. 2. The 'Host Credentials' and 'Database Credentials' sections in the 'Specify Host and Target Database Credentials' dialog. 3. The 'Confirmation' dialog asking to perform the shutdown operation. 4. The 'Advanced Shutdown Options' dialog where the shutdown mode is selected.

General

Status **Up**

Up Since Jun 19, 2009 2:02:00 AM GMT+07:00

Instance Name orcl

Version 11.2.0.1.0

Host edrsr25p1.us.oracle.com

Listener LISTENER_edrsr25p1.us.oracle.com

ASM +ASM_edrsr25p1.us.oracle.com

[View All Properties](#)

Startup/Shutdown: Specify Host and Target Database Credentials

Specify the following credentials in order to change the status of the database.

Host Credentials

Specify the OS user name and password to login to target database machine.

* Username oracle

* Password

Database Credentials

Specify the credentials for the target database.

To use OS authentication, leave the user name and password fields blank.

* Username sys

* Password

Database orcl.example.com

* Connect As SYSDBA

Startup/Shutdown: Advanced Shutdown Options

Specify the shutdown mode

☐ Normal [Browse Sessions](#)

☐ Transactional

☒ Immediate

☐ Abort

Startup/Shutdown: Confirmation

Current Status open

Operation shutdown immediate

Are you sure you want to perform this operation?

[Show SQL](#) [Advanced Options](#) [No](#) [Yes](#)

ORACLE

© Oracle, 2009. Все права защищены.

Остановка экземпляра базы данных Oracle

Если при переходе на страницу управления базой данных Enterprise Manager экземпляр уже был запущен, нажмите кнопку Shutdown ("Остановка"), чтобы остановить экземпляр. Будет выведен запрос на подтверждение или ввод учетных данных хоста и базы данных. Нажмите кнопку ОК, чтобы открыть диалоговое окно подтверждения Startup/Shutdown ("Запуск и остановка"). Нажмите кнопку Advanced Options ("Дополнительные параметры"), чтобы выбрать режим остановки: NORMAL, TRANSACTIONAL, IMMEDIATE или ABORT.

Режимы остановки

Режимы остановки	A	I	T	N
Разрешены новые подключения	Нет	Нет	Нет	Нет
Ожидание завершения текущего сеанса	Нет	Нет	Нет	Да
Ожидание завершения текущей транзакции	Нет	Нет	Да	Да
Принудительное создание контрольной точки и закрытие файлов	Нет	Да	Да	Да

Режимы остановки:

- A = ABORT
- I = IMMEDIATE
- T = TRANSACTIONAL
- N = NORMAL

ORACLE

© Oracle, 2009. Все права защищены.

Режимы остановки

Ниже приведен перечень режимов остановки в возрастающем порядке дружелюбности по отношению к текущим операциям:

- **ABORT:** выполняется минимальное количество операций до остановки. Так как этот режим требует выполнения восстановления перед запуском, используйте его только при необходимости. Данный режим обычно используется в ситуациях, когда не срабатывает ни один другой режим остановки, когда возникли проблемы при запуске экземпляра или же когда требуется немедленно остановить экземпляр вследствие чрезвычайного происшествия (например, когда можно ожидать отключения электропитания в любую секунду).
- **IMMEDIATE:** наиболее часто используемый режим. Для всех неподтвержденных транзакций выполняется откат.
- **TRANSACTIONAL:** позволяет завершить транзакции, но не запускает новые.
- **NORMAL:** ожидает завершения сеансов.

По времени, затрачиваемому на выполнение остановки, самым быстрым является режим ABORT, а самым медленным – NORMAL. Режимы NORMAL и TRANSACTIONAL могут потребовать много времени на остановку в зависимости от количества сеансов и транзакций.

Параметры остановки

Перед остановкой:

- неподтвержденные изменения откатываются для режима IMMEDIATE;
- буферный кэш базы данных записывается в файлы данных;
- ресурсы освобождаются.

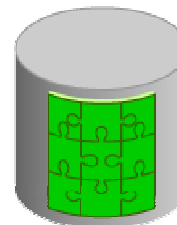
Во время:

**SHUTDOWN
NORMAL
или
SHUTDOWN
TRANSACTIONAL
или
SHUTDOWN
IMMEDIATE**

Перед запуском:

- восстановление экземпляра не выполняется.

Целостная база данных



ORACLE

© Oracle, 2009. Все права защищены.

Параметры остановки

SHUTDOWN NORMAL

Режим **NORMAL** является режимом остановки по умолчанию, если режим не указан. При остановке базы данных в обычном режиме выполняются следующие операции:

- Запрещаются новые подключения.
- Сервер Oracle ожидает отключения всех пользователей до выполнения полной остановки.
- Буферы повторов и базы данных записываются на диск.
- Завершаются фоновые процессы, и SGA выгружается из памяти.
- Сервер Oracle закрывает и размонтирует базу данных перед остановкой экземпляра.
- При следующей загрузке восстановление экземпляра не требуется.

SHUTDOWN TRANSACTIONAL

Режим остановки **TRANSACTIONAL** позволяет сохранить данные клиентов, включая результаты выполняемых в данный момент операций. При остановке базы данных в транзакционном режиме выполняются следующие операции:

- Клиенты не могут создавать новые транзакции для текущего экземпляра.
- Клиент отключается после завершения текущей транзакции.
- После завершения всех транзакций сразу же происходит полная остановка.
- При следующей загрузке восстановление экземпляра не требуется.

Параметры остановки (продолжение)

SHUTDOWN IMMEDIATE

При остановке базы данных в режиме **IMMEDIATE** выполняются следующие операции:

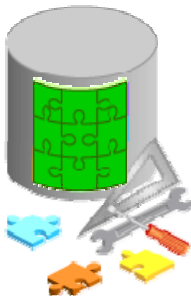
- Обрабатываемый базой данных Oracle в данный момент SQL-оператор завершен не будет.
- Сервер Oracle не будет ожидать отключения подключенных к базе данных пользователей.
- Сервер Oracle выполнит откат активных транзакций и отключит всех подключенных пользователей.
- Сервер Oracle закрывает и размонтирует базу данных перед остановкой экземпляра.
- **При следующей загрузке восстановление экземпляра не требуется.**

Примечание. Режим **IMMEDIATE** является режимом по умолчанию при использовании Enterprise Manager.

Параметры остановки

Перед остановкой:

- измененные буферы не записываются в файлы данных;
- откат неподтвержденных изменений не выполняется.



Во время:

SHUTDOWN ABORT
или
сбои экземпляра;
или
STARTUP FORCE

**Нецелостная база
данных**

Перед запуском:

- оперативные файлы журнала повторов используются для повторного применения изменений;
- сегменты отмены операций используются для отката неподтвержденных изменений;
- ресурсы освобождаются.

ORACLE

© Oracle, 2009. Все права защищены.

Параметры остановки (продолжение)

SHUTDOWN ABORT

Если режимы остановки **NORMAL**, **TRANSACTIONAL** и **IMMEDIATE** не работают, можно прервать работу текущего экземпляра базы данных. При прерывании экземпляра выполняются следующие операции:

- Обработываемый базой данных Oracle в данный момент SQL-оператор немедленно завершается.
- Сервер Oracle не ожидает отключения подключенных к базе данных пользователей.
- Буферы повторов и базы данных не записываются на диск.
- Откат неподтвержденных транзакций не выполняется.
- Работа экземпляра прерывается без закрытия файлов.
- База данных не закрывается и не размонтируется.
- При следующем запуске потребуется выполнить восстановление экземпляра, что будет сделано автоматически.

Примечание. Не рекомендуется выполнять резервное копирование нецелостной базы данных.

Параметры остановки: примеры

- С помощью SQL*Plus:

```
SQL> shutdown
```

1

```
SQL> shutdown transactional
```

2

```
SQL> shutdown immediate
```

3

```
SQL> shutdown abort
```

4

- Использование служебной программы SRVCTL с Oracle Restart

```
$ srvctl stop database -d orcl -o abort
```

ORACLE

© Oracle, 2009. Все права защищены.

Параметры остановки: примеры

На этом слайде представлены примеры использования SQL*Plus и служебной программы SRVCTL для завершения работы базы данных.

1. Эта команда инициирует нормальное завершение работы. База данных завершит работу только после выхода всех пользователей.
2. Эта команда инициирует транзакционное завершение работы. База данных завершит работу только после завершения всех существующих транзакций.
3. Эта команда инициирует немедленное завершение работы. Для незафиксированных транзакций будет выполнен откат.
4. Эта команда инициирует завершение работы shutdown abort.

Если база данных включена в Oracle Restart, то служебная программа SRVCTL позволяет завершить работу экземпляра базы данных.

Примечание. Программа SRVCTL находится в каталоге \$ORACLE_HOME/bin для ПО Grid infrastructure и в каталоге \$ORACLE_HOME/bin для ПО базы данных Oracle. При запуске базы данных Oracle необходимо использовать программу SRVCTL из ПО баз данных Oracle. При запуске экземпляра ASM или приемника следует использовать программу SRVCTL из ПО Grid Infrastructure.

Просмотр журнала предупреждений

Домашняя страница базы данных > область
Related Links ("Связанные ссылки") >
Содержание журнала предупреждений

Компоненты
SQL*Plus
Параметры инициализации
Запуск БД
Остановка БД
> **Журнал предупреждений**
Представления
производительности

Timestamp	Type	Level	Incident ID	Group	Message ID	Message Text
Jun 19, 2009 10:00:16 PM GMT+07:00	NOTIFICATION	16		sqltune	kesaiTuneSqlDrv:5067:3456118459	End automatic SQL Tuning Advisor run for special tuning task "SYS_AUTO_SQL_TUNING_TASK"
Jun 19, 2009 10:00:03 PM GMT+07:00	NOTIFICATION	16		sqltune	kesaiTuneSqlDrv:4555:2579917519	Begin automatic SQL Tuning Advisor run for special tuning task "SYS_AUTO_SQL_TUNING_TASK"
Jun 19, 2009 10:00:00 PM GMT+07:00	NOTIFICATION	16		process start	ksbrdp:3833:3697353022	VKRM started with pid=24, OS id=7929
Jun 19, 2009 10:00:00 PM GMT+07:00	NOTIFICATION	16		process start	ksbs1p_real:2253:2371767696	Starting background process VKRM
Jun 19, 2009 2:07:22 AM GMT+07:00	NOTIFICATION	16		process start	ksbrdp:3833:3697353022	SMCO started with pid=23, OS id=30582
Jun 19, 2009 2:07:22 AM GMT+07:00	NOTIFICATION	16		process start	ksbs1p_real:2253:2371767696	Starting background process SMCO
Jun 19, 2009 2:02:26 AM GMT+07:00	NOTIFICATION	16		process start	ksbrdp:3833:3697353022	CJQ0 started with pid=33, OS id=29846

ORACLE

© Oracle, 2009. Все права защищены.

Просмотр журнала предупреждений

Каждая база данных имеет файл `alert_<sid>.log`. Этот файл расположен на сервере с базой данных и по умолчанию хранится в каталоге `$ORACLE_BASE/diag/rdbms/<ИМЯ_БД>/<SID>/trace`, если был установлен параметр `$ORACLE_BASE`.

Журнал предупреждений базы данных – это хронологический журнал сообщений о следующих аспектах:

- использованные при загрузке нестандартные параметры инициализации;
- все внутренние ошибки (ORA - 600), ошибки повреждения блоков (ORA - 1578), а также ошибки взаимных блокировок (ORA - 60);
- административные операции, например, операторы SQL CREATE, ALTER, DROP DATABASE и TABLESPACE, а также операторы Enterprise Manager или SQL*Plus STARTUP, SHUTDOWN, ARCHIVE LOG и RECOVER;
- некоторые сообщения и ошибки, связанные с функциями разделяемого сервера процессов диспетчера;
- ошибки, возникшие при автоматическом обновлении материализованного представления.

База данных Oracle использует журнал предупреждений для хранения сведений о происшедших событиях. Журнал предупреждений является альтернативой консоли оператора, в которой также отображаются вышеописанные сведения. (Большинство систем также отображают вышеописанные сведения в консоли). Если операция администрирования была выполнена успешно, то она будет записана в журнал предупреждений как завершенная с отметкой времени.

Просмотр журнала предупреждений (продолжение)

Enterprise Manager отслеживает файл журнала предупреждений и уведомляет вас о критических ошибках. Журнал предупреждений также можно использовать для просмотра сведений о некритических ошибках и информационных сообщений. Так как файл журнала предупреждений со временем будет становиться слишком большим, его можно периодически удалять. При следующей попытке базы данных создать запись в журнале предупреждений файл будет создан заново.

Примечание. В каталоге \$ORACLE_BASE/diag/rdbms/<db_name>/<SID>/alert размещена XML-версия журнала предупреждений.

Определение расположения журнала предупреждений с помощью SQL*Plus:

- Подключитесь к базе данных с помощью SQL*Plus (или другого приложения, например SQL Developer).
- Запросите представление V\$DIAG_INFO.

Чтобы просмотреть текстовую версию журнала предупреждений без XML-тегов:

- В результатах опроса представления V\$DIAG_INFO найдите путь, соответствующий записи Diag Trace. Перейдите к указанному каталогу.
- Откройте в текстовом редакторе файл alert_SID.log.

Чтобы просмотреть журнал предупреждений с XML-форматированием:

- В результатах опроса представления V\$DIAG_INFO найдите путь, соответствующий записи Diag Alert. Перейдите к указанному каталогу.
- Откройте в текстовом редакторе файл log.xml.

Использование файлов трассировки

- Каждый сервер и фоновый процесс может создавать записи в связанном с ним файле трассировки.
- Сведения об ошибке также записываются в соответствующий файл трассировки.
- Репозиторий автоматической диагностики (ADR)
 - представляет собой централизованный системный репозиторий для трассировки и регистрации
 - хранит данные диагностики базы данных, такие как:
 - данные трассировки;
 - журнал предупреждений;
 - отчеты монитора пригодности.

ORACLE

© Oracle, 2009. Все права защищены.

Использование файлов трассировки

Каждый сервер и фоновый процесс может создавать записи в связанном с ним файле трассировки. Когда процесс обнаруживает внутреннюю ошибку, он сохраняет сведения об ошибке в своем файле трассировки. Если произошла внутренняя ошибка и сведения о ней были записаны в файл трассировки, администратору следует связаться со службой поддержки Oracle Support Services.

Все имена файлов трассировки, связанных с фоновым процессом, содержат имя создавшего их процесса. Единственным исключением являются файлы трассировки, созданные процессом очереди заданий (Jnnn).

Дополнительные сведения в файлах трассировки могут дать указания по настройке приложений или экземпляра. Фоновые процессы всегда сохраняют такие данные в файлах трассировки, когда это возможно.

Начиная с Oracle Database 11g, для предотвращения, обнаружения, диагностики и устранения проблем используется расширенная инфраструктура диагностики сбоев. В частности, особое внимание уделяется критическим ошибкам, связанным с ошибками в коде базы данных, повреждением метаданных и пользовательских данных.

Использование файлов трассировки (продолжение)

При наступлении критической ошибки ей присваивается номер инцидента. После этого диагностические сведения об ошибке немедленно записываются в файлы трассировки и помечаются соответствующим номером. Эти данные сохраняются в репозитории автоматической диагностики (ADR) – файловом репозитории, расположенном за пределами базы данных, откуда их можно будет извлечь для дальнейшего анализа по номеру инцидента.

ADR представляет собой централизованный системный репозиторий данных трассировки и регистрации, который хранит сведения для диагностики базы данных, такие как данные трассировки, журнал предупреждений, отчеты монитора пригодности и т. д.

Корневой каталог ADR называется *базовым каталогом ADR*. Он определяется параметром инициализации **DIAGNOSTIC_DEST**. Если этот параметр опущен или не указано его значение, то при загрузке база данных присвоит параметру **DIAGNOSTIC_DEST** значение в зависимости от следующих условий:

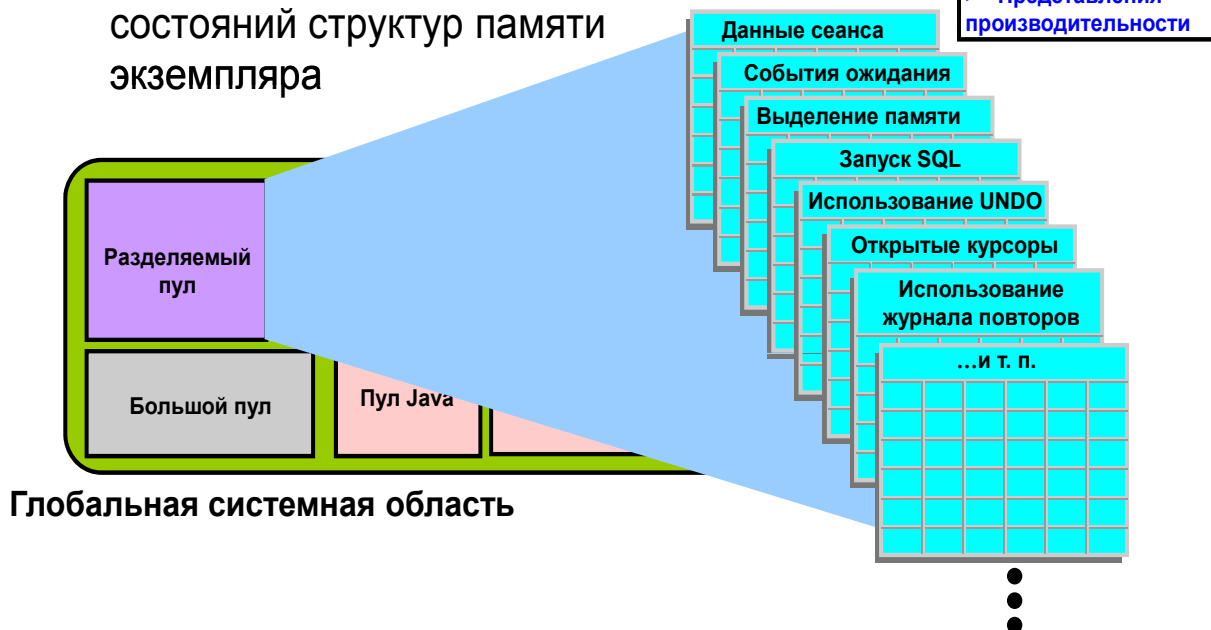
- если значение переменной среды **ORACLE_BASE** установлено, параметру **DIAGNOSTIC_DEST** будет присвоено значение, указанное для переменной **ORACLE_BASE**.
- если значение переменной среды **ORACLE_BASE** не установлено, параметру **DIAGNOSTIC_DEST** будет присвоено значение **ORACLE_HOME/log**.

Корневой каталог ADR размещен в указанном ниже каталоге, точкой отсчета для которого служит базовый каталог ADR:

`./diag/product_type/db_id/instance_id`

Динамические представления производительности

Обеспечивают доступ к сведениям об изменениях состояний структур памяти экземпляра



© Oracle, 2009. Все права защищены.

Динамические представления производительности

База данных Oracle поддерживает более динамические наборы данных, которые отражают функционирование и производительность экземпляра базы данных. Такие динамические представления производительности основываются на виртуальных таблицах, созданных на базе структур памяти и расположенных на сервере базы данных. Таким образом, они не являются обычными таблицами, расположенными в базе данных. С этим связано то, что некоторые представления доступны до монтирования или открытия базы данных.

Динамические представления производительности содержат сведения о следующих аспектах:

- сеансы;
- состояние файлов;
- ход выполнения заданий и задач;
- блокировки;
- состояние резервного копирования;
- использование и выделение памяти;
- параметры системы и сеансов;
- выполнение SQL;
- статистические данные и метрики.

Примечание. Имена динамических представлений производительности также содержатся в представлениях `DICT` и `DICT_COLUMNS`. Динамические представления данных производительности начинаются с префикса 'v\$', а их число превышает 590.

Динамические представления производительности: примеры использования

1 SQL> SELECT sql_text, executions FROM v\$sql
WHERE cpu_time > 200000;

2 SQL> SELECT * FROM v\$session WHERE machine =
'EDRSR9P1' and logon_time > SYSDATE - 1;

3 SQL> SELECT sid, ctime FROM v\$lock
WHERE block > 0;

ORACLE

© Oracle, 2009. Все права защищены.

Динамические представления производительности: примеры использования

Данные представления часто используются в Enterprise Manager, но пользователи также могут просматривать их в случае необходимости. Примеры из слайда дают ответы на следующие вопросы:

1. Выполнение каких SQL-операторов требует более 200000 микросекунд процессорного времени?
2. Какие из текущих сеансов были открыты с компьютера EDRSR9P1 за последний день?
3. Каковы идентификаторы сеансов, в настоящий момент применяющих блокировки от других пользователей, а также сколько времени существуют данные блокировки?

Динамические представления производительности: рекомендации

- Данные представления принадлежат пользователю SYS.
- На различных отрезках времени доступны различные представления:
 - Экземпляр запущен.
 - База данных смонтирована.
 - База данных открыта.
- Чтобы отобразить имена всех представлений, следует запросить представление V\$FIXED_TABLE.
- Такие представления часто называют представлениями "v-доллар".
- Целостность таких представлений не гарантируется ввиду динамической природы данных.

ORACLE

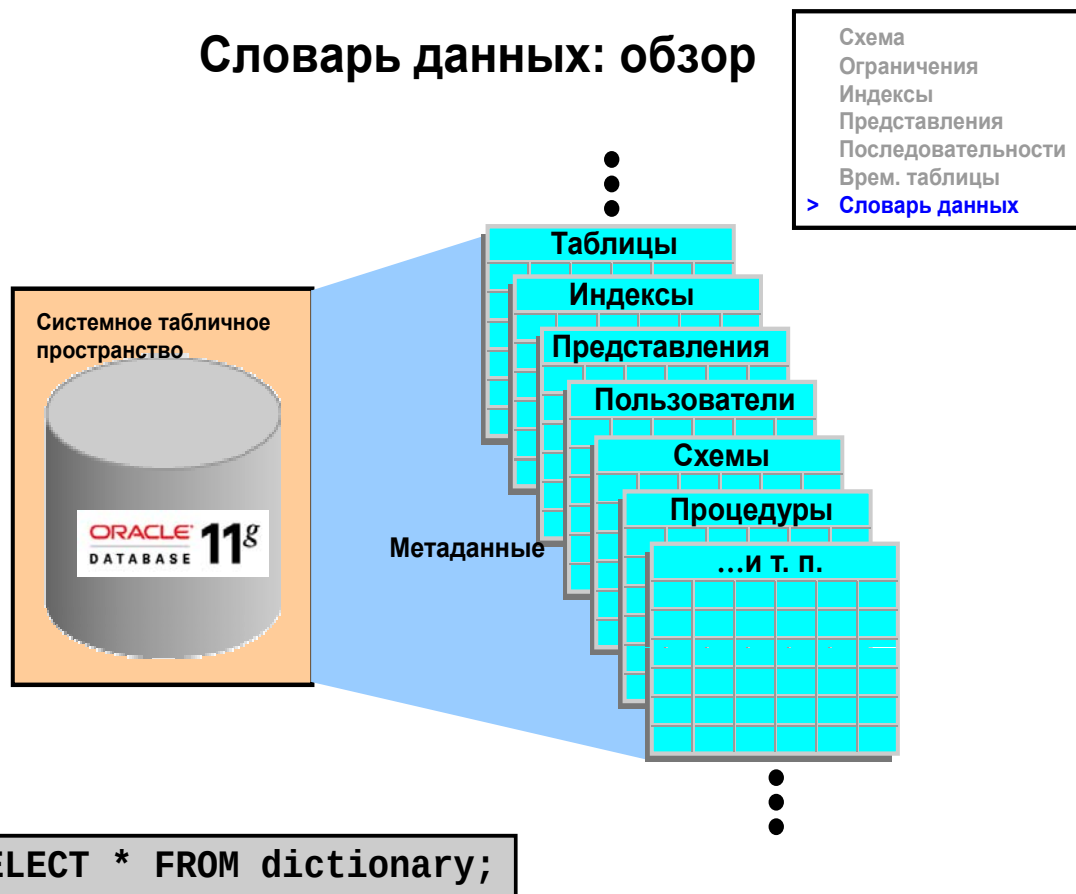
© Oracle, 2009. Все права защищены.

Динамические представления производительности: рекомендации

Некоторые динамические представления содержат данные, не применимые ко всем состояниям базы данных. Например, если экземпляр только что был запущен, но база данных еще не была смонтирована, можно опросить представление V\$BGPROCESS для просмотра списка запущенных фоновых процессов. Однако невозможно опросить представление V\$DATAFILE для просмотра состояний файлов базы данных, так как только во время монтирования базы данных происходит считывание управляющего файла для выявления связанных с базой данных файлов данных.

Некоторые представления V\$ содержат сведения, подобные сведениям из представлений DBA_. Например, сведения из представления V\$DATAFILE подобны сведениям из представления DBA_DATAFILES. Также обратите внимание на то, что имена представлений V\$ обычно указываются в единственном числе, в то время как имена представлений DBA_ указываются в множественном числе.

Словарь данных: обзор



```
SELECT * FROM dictionary;
```

ORACLE

© Oracle, 2009. Все права защищены.

Словарь данных: обзор

Словарь данных Oracle является метаданными базы данных и содержит имена и атрибуты всех объектов базы данных. Все операции создания или изменения каких-либо объектов записываются в словарь данных для отражения изменений. Данные сведения хранятся в базовых таблицах, которые обслуживаются базой данных Oracle. Обращаться к данным таблицам следует через заранее определенные представления, а не напрямую.

Сведения словаря данных:

- используются сервером баз данных Oracle для поиска информации о клиентах, объектах, ограничениях и хранилищах;
- обновляются сервером базы данных Oracle при обновлении объектных структур или описаний;
- доступны любому пользователю для получения сведений о базе данных;
- принадлежат пользователю SYS;
- не должны изменяться непосредственно с помощью SQL.

Примечание. Представление словаря данных **DICTIONARY** (или его синоним **DICT**) содержит имена и описания таблиц и представлений словаря данных. В представлении **DICT_COLUMNS** можно просмотреть столбцы представлений и их определения. Полное определение каждого из представлений см. в справочнике *Oracle Database Reference*. Существует более 1000 представлений, ссылающихся на сотни базовых таблиц.

Представления словаря данных

	Пользователи, имеющие право на доступ	Содержимое	Подмножество	Заметки
DBA_	Администратор БД	Все	Н/П	Могут содержать дополнительные столбцы (только для DBA)
ALL_	Все	Все, что позволяют увидеть пользователю его полномочия	Представления DBA_	Включает в себя собственные объекты пользователя и другие объекты, на просмотр которых у пользователя имеются полномочия
USER_	Все	Все, что принадлежит пользователю	Представления ALL_	Обычно подобны ALL_ за исключением отсутствующего столбца OWNER (для некоторых представлений определены сокращенные имена в виде синонимов PUBLIC).

ORACLE

© Oracle, 2009. Все права защищены.

Представления словаря данных

Префиксы представлений указывают на данные (и их объем), которые пользователь может увидеть.

Глобальные представления с полными сведениями обо всем доступны только пользователям с полномочиями DBA. Они содержат префикс DBA_.

Следующим уровнем полномочий является уровень префикса ALL_, который представляет все объекты, которые может увидеть пользователь с заданными полномочиями независимо от того, является ли он владельцем данных объектов. Например, если пользователю USER_A был предоставлен доступ к таблице, которой владеет пользователь USER_B, то пользователь USER_A сможет увидеть таблицу, указанную в любом представлении ALL_, которое связано с именами таблиц.

Префикс USER_ отражает более узкий диапазон видимости. В представлении такого типа можно увидеть только те объекты, которыми владеет пользователь (то есть объекты, присутствующие в личной схеме пользователя).

В общем случае каждый набор представлений является подмножеством набора представлений с более высокими требованиями к полномочиям как по строкам, так и по столбцам. Не все представления в заданном наборе представлений имеют соответствующие им представления в других наборах представлений.

Это зависит от природы данных представления. Например, существует представление DBA_LOCK, однако представления ALL_LOCK не существует. Причиной этому является то, что только DBA может заинтересоваться сведениями о блокировках. Выбирая нужный набор представлений, убедитесь, что он соответствует вашим потребностям. При наличии полномочий для доступа к представлениям DBA вам может потребоваться обратиться к версии представления USER, так как в результатах отображаются данные об объектах, которыми вы владеете, а добавление других объектов в результирующий набор может быть нежелательным.

Представления словаря данных (продолжение)

Представления DBA_ могут запрашиваться только пользователями с привилегиями SYSDBA или SELECT ANY DICTIONARY.

Не все представления словаря начинаются с префикса DBA_, ALL_ и USER_. Следующие представления или синонимы представлений являются исключениями из этого правила:

- AUDIT_ACTIONS
- CAT
- CHANGE_PROPAGATIONS
- CHANGE_PROPAGATION_SETS
- CHANGE_SETS
- **CHANGE_SOURCES**
- CHANGE_TABLES
- CLIENT_RESULT_CACHE_STATS\$
- CLU
- COLS
- COLUMN_PRIVILEGES
- DATABASE_COMPATIBLE_LEVEL
- **DBMS_ALERT_INFO**
- DBMS_LOCK_ALLOCATED
- DICT
- DICTIONARY
- DICT_COLUMNS
- DUAL
- GLOBAL_NAME
- **IND**
- INDEX_HISTOGRAM
- INDEX_STATS
- LOGSTDBY_UNUNSUPPORTED_TABLES
- NLS_DATABASE_PARAMETERS
- NLS_INSTANCE_PARAMETERS
- NLS_SESSION_PARAMETERS
- **OBJ**
- RECYCLEBIN
- RESOURCE_COST
- ROLE_ROLE_PRIVS
- ROLE_SYS_PRIVS
- ROLE_TAB_PRIVS
- SEQ
- **SESSION_PRIVS**
- SESSION_ROLES
- SM\$VERSION
- SYN
- TABLE_PRIVILEGES
- TABS

Словарь данных: примеры использования

- 1

```
SELECT table_name, tablespace_name
FROM user_tables;
```
- 2

```
SELECT sequence_name, min_value, max_value,
increment_by
FROM all_sequences
WHERE sequence_owner IN ('MDSYS', 'XDB');
```
- 3

```
SELECT USERNAME, ACCOUNT_STATUS
FROM dba_users
WHERE ACCOUNT_STATUS = 'OPEN';
```
- 4

```
DESCRIBE dba_indexes
```

ORACLE

© Oracle, 2009. Все права защищены.

Словарь данных: примеры использования

Примеры запросов дают ответы на следующие вопросы:

1. Каковы имена таблиц (а также имя табличного пространства, в котором они располагаются), которые были созданы в вашей схеме?
2. К каким важным сведениям о последовательностях базы данных у вас есть доступ?
3. Какие пользователи базы данных могут в данный момент подключиться?
4. Какие столбцы содержит представление DBA_INDEXES? Ниже отображена информация о том, какие сведения об индексах базы данных доступны вам. Фрагмент выходных данных результата выполнения данной команды:

```
SQL> DESCRIBE dba_indexes
Name                Null?    Type
-----
OWNER               NOT NULL VARCHAR2(30)
INDEX_NAME          NOT NULL VARCHAR2(30)
INDEX_TYPE                   VARCHAR2(27)
TABLE_OWNER         NOT NULL VARCHAR2(30)
TABLE_NAME          NOT NULL VARCHAR2(30)
```

Экспресс-тест

При использовании Oracle Restart необходимо использовать программу управления сервером `srvctl` вместо SQL*Plus для запуска и остановки экземпляра базы данных.

1. Да
2. Нет

ORACLE

© Oracle, 2009. Все права защищены.

Ответ: 2

Экспресс-тест

Какой словарь можно использовать для поиска имен всех таблиц в базе данных?

1. USER_TABLES
2. ALL_TABLES
3. DBA_TABLES
4. ANY_TABLES

ORACLE

© Oracle, 2009. Все права защищены.

Ответ: 3

Заключение

В ходе прохождения урока вы освоили следующие темы:

- запуск и остановка базы данных Oracle и ее компонентов;
- **использование Oracle Enterprise Manager;**
- обращение к базе данных с помощью SQL*Plus;
- изменение параметров инициализации базы данных;
- описание этапов запуска базы данных;
- описание параметров остановки базы данных;
- просмотр журнала предупреждений;
- обращение к динамическим представлениям производительности.

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is positioned on the right side of a solid red horizontal bar.

© Oracle, 2009. Все права защищены.

Обзор упражнения 4: управление экземпляром Oracle

В данном упражнении рассматриваются следующие темы:

- навигация в Enterprise Manager;
- отображение и изменение параметров инициализации;
- остановка и запуск экземпляра базы данных;
- просмотр журнала предупреждений;
- подключение к базе данных с помощью SQL*Plus.

ORACLE

© Oracle, 2009. Все права защищены.

5

Управление экземпляром ASM

ORACLE®

© Oracle, 2009. Все права защищены.

Цели

По завершении данного урока вы сможете:

- описать преимущества использования ASM;
- осуществлять управление экземпляром ASM;
- создавать и удалять дисковые группы ASM;
- расширять дисковые группы ASM;
- извлекать метаданные ASM с помощью различных служебных программ.

ORACLE

© Oracle, 2009. Все права защищены.

Цели

В этом уроке представлено более подробное описание экземпляра ASM и способов управления им при помощи различных служебных программ.

Преимущества ASM для администраторов

ASM позволяет избежать выполнения следующих действий:

- настройка производительности подсистем ввода-вывода;
- перемещение и реорганизация файлов данных;
- управление именами файлов;
- управление логическими томами;
- управление файловой системой;
- управление кластерной файловой системой;
- управление неформатированными накопителями.

ASM значительно сокращает:

- управление номерами логических блоков:
 - меньшие, большие номера логических блоков;
- зависимость администратора базы данных от системного администратора;
- вероятность возникновения ошибок, связанных с выполнением задач обслуживания вручную.

ORACLE

© Oracle, 2009. Все права защищены.

Преимущества ASM для администраторов

ASM устраняет необходимость выполнения многих задач, которые необходимо выполнять в средах без хранилищ ASM. К ним относятся следующие:

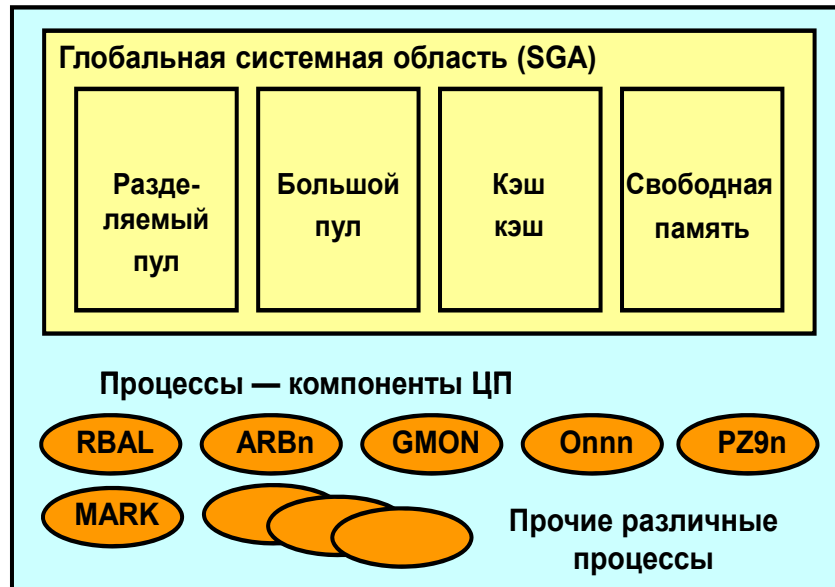
- **Настройка производительности подсистем ввода-вывода:** политика разнесения и зеркалирования всех элементов ASM в сочетании с автоматическими операциями определения баланса означает, что настройка производительности подсистем ввода-вывода, направленная на балансировку использования диска и устранение горячих дисковых точек, не требуется.
- **Перемещение и реорганизация файлов данных:** постоянное изменение размещения файлов данных для удовлетворения требований относительно производительности и ограничений пространства не требуется.
- **Управление именами файлов:** вам больше не требуется определять политику именования файлов и обеспечивать ее применение.
- **Управление логическими томами, файловой системой, кластерной файловой системой и неформатированными накопителями:** вам больше не требуются эти элементы хранения.

ASM предоставляет дополнительные преимущества за счет сокращения усилий в следующих важных областях:

- Временные затраты на управление номерами логических устройств (LUN) сокращаются, так как ASM, как правило, требуются меньшие и большие LUN.
- Зависимость, которая часто существует между администратором базы данных и системным администратором, значительно снижается. Например, вмешательство системного администратора для добавления нового файла данных или перемещения дисковых ресурсов из одной дисковой группы в другую не требуется.
- Вероятность возникновения ошибок, связанных с выполнением задач обслуживания вручную, значительно снижается. Например, использование обычной файловой системой недавно созданного файла данных может приводить к случайному нарушению правила именования файлов, в результате чего данный файл не включается в резервную копию всей остальной базы данных.

Экземпляр ASM

Экземпляр ASM представляет собой сочетание процессов и компонентов памяти для ASM.



Экземпляр ASM

ORACLE

© Oracle, 2009. Все права защищены.

Экземпляр ASM

При каждом запуске ASM или базы данных выделяется общая область памяти, называемая глобальной системной областью (System Global Area — SGA), а также запускаются фоновые процессы Oracle ASM или базы данных. Сочетание фоновых процессов и SGA называется экземпляром Oracle ASM или экземпляром базы данных Oracle. Экземпляр представляет собой сочетание процессов ЦП и ОЗУ запущенной среды ASM.

SGA в экземпляре ASM отличается от SGA в экземпляре базы данных с точки зрения распределения и использования памяти. SGA в экземпляре ASM разделен на четыре основных области следующим образом:

- **Разделяемый пул:** используется для информации метаданных.
- **Большой пул:** используется для параллельных операций.
- **Кэш ASM:** используется для чтения и записи блоков в ходе операций определения баланса.
- **Свободная память:** доступная нераспределенная память.

Минимальный рекомендуемый объем памяти для экземпляра ASM составляет 256 Мбайт. Функция автоматического управления памятью для экземпляра ASM включена по умолчанию и динамически настраивает размер отдельных компонентов памяти SGA. Объем памяти, необходимый для экземпляра ASM, зависит от объема дискового пространства, управляемого ASM.

Вторая часть экземпляра ASM — это фоновые процессы. Экземпляр ASM может иметь множество фоновых процессов; не все фоновые процессы выполняются постоянно.

Компоненты ASM: экземпляр ASM (продолжение)

Фоновые процессы, относящиеся к функции ASM, представлены на следующем слайде. Существуют обязательные фоновые процессы и дополнительные фоновые процессы. К этим процессам могут относиться следующие:

- **ARCn**: процесс архиватора.
- **CKPT**: процесс создания контрольной точки.
- **DBWn**: процесс записи в базу данных.
- **DIAG**: процесс диагностики.
- **Jnnn**: процессы обработки очереди заданий.
- **LGWR**: процесс записи в журнал.
- **PMON**: процесс монитора процессов.
- **PSP0**: процесс порождения процессов.
- **QMn**: процессы монитора очередей.
- **RECO**: процесс восстановления.
- **SMON**: процесс системного монитора.
- **VKTM**: процесс виртуального контролера времени.
- **MMAN**: процесс диспетчера памяти.

Приведенный выше список процессов является неполным. Для экземпляра ASM эти процессы не всегда выполняют те же задачи, которые они выполняли бы в экземпляре базы данных. Например, процесс LGWR в экземпляре базы данных отвечает за копирование векторов изменений из раздела буфера журнала SGA в оперативные журналы повторов на диске. Экземпляр ASM не содержит буфер журнала в SGA и не использует оперативные журналы повторов. Процесс LGWR в экземпляре ASM копирует регистрационную информацию в дисковую группу ASM.

Если ASM входит в состав кластера, дополнительные процессы, связанные с управлением кластером, будут выполняться в экземпляре ASM. К этим процессам относятся следующие:

- **LMON**: глобальный процесс сервисного монитора очереди.
- **LMDn**: глобальные сервисные демоны очереди.
- **LMSn**: глобальные сервисные процессы кэша.
- **LCKn**: процессы блокировки.

Компоненты ASM: экземпляр ASM — основные процессы

Основные процессы экземпляра ASM несут ответственность за действия, относящиеся к ASM.

Процесс	Описание
RBAL	Открывает все файлы устройств в рамках процесса обнаружения и координирует операции повторного определения баланса
ARBn	Один или несколько подчиненных процессов, которые выполняют операции повторного определения баланса
GMON	Несет ответственность за выполнение действий на уровне дисков, таких как удаление или перевод в автономный режим, а также за обеспечение совместимости дисковой группы ASM
MARK	Отмечает выделенные блоки ASM как устаревшие по мере необходимости
Onnn	Один или несколько подчиненных процессов ASM, которые формируют пул соединений с экземпляром ASM для обмена сообщениями
PZ9n	Один или несколько параллельных подчиненных процессов, которые используются для получения данных в кластерной установке ASM из представлений GV\$

ORACLE

© Oracle, 2009. Все права защищены.

Компоненты ASM: экземпляр ASM — основные процессы

Экземпляр ASM использует выделенные фоновые процессы для выполнения большинства функций. Процесс **RBAL** координирует операции повторного определения баланса для групп дисков в экземпляре автоматического управления хранением. Он выполняет глобальное открытие дисков для автоматического управления хранением. Процесс **ARBn** выполняет фактическое перемещение экстенда данных повторного определения баланса в экземпляре автоматического управления хранением. Одновременно может существовать несколько таких процессов с именами **ARB0**, **ARB1** и т. д. Процесс **GMON** поддерживает членство диска в дисковых группах ASM. Процесс **MARK** отмечает выделенные блоки ASM как устаревшие после пропущенной записи на отключенный диск. Процесс **Onnn** представляют серверную часть соединения «клиент-сервер». Эти процессы появляются в момент запуска экземпляра и исчезают после этого. Они образуют пул соединений с экземпляром ASM для обмена сообщениями и отображаются только по мере необходимости. Процесс **PZ9n** представляет один или несколько параллельных подчиненных процессов, которые используются для получения данных, когда ASM одновременно запускается в кластерной конфигурации на нескольких компьютерах.

Параметры инициализации экземпляра ASM

Экземпляр ASM использует небольшой поднабор параметров, которые используются экземпляром базы данных Oracle Database.

```
INSTANCE_TYPE = ASM  
ASM_POWER_LIMIT = 1  
ASM_DISKSTRING = '/dev/sda1', '/dev/sdb*'  
ASM_DISKGROUPS = DATA2, FRA  
ASM_PREFERRED_READ_FAILURE_GROUPS = DATA.FailGroup2  
DIAGNOSTIC_DEST = /u01/app/oracle  
LARGE_POOL_SIZE = 12M  
REMOTE_LOGIN_PASSWORDFILE = EXCLUSIVE
```

ORACLE

© Oracle, 2009. Все права защищены.

Параметры инициализации экземпляра ASM

Экземпляр ASM управляется файлом параметров таким же образом, как и обычный экземпляр базы данных. Набор параметров, как правило, включает следующие параметры:

- **INSTANCE_TYPE** — для всех экземпляров ASM следует задать значение ASM. Это единственный параметр, который необходимо задать. Для экземпляров базы данных задается значение RDBMS.
- **ASM_POWER_LIMIT** контролирует скорость операции повторного определения баланса. Доступны значения в диапазоне от 1 до 11, где 11 — это самая высокая скорость. Если пользователь не задает это значение, устанавливается значение 1.
- **ASM_DISKSTRING** — это значение, которое зависит от операционной системы и используется ASM для ограничения набора дисков, используемых при обнаружении. По умолчанию используется строка pull, и этого достаточно в большинстве случаев. Значение, накладывающее большее количество ограничений, может сокращать время, необходимое ASM для выполнения обнаружения, тем самым оптимизируя монтирование дисковых групп.

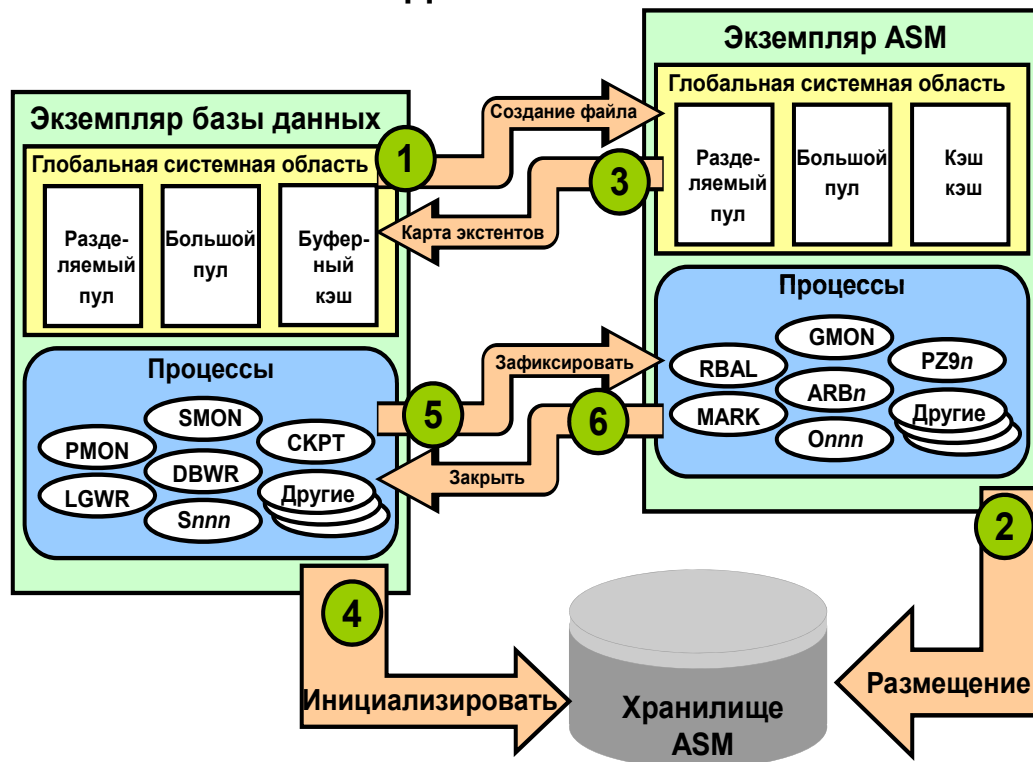
Параметры инициализации экземпляра ASM (продолжение)

- **ASM_DISKGROUPS** — это список имен дисковых групп, которые монтируются экземпляром ASM при запуске или при использовании команды **ALTER DISKGROUP ALL MOUNT**. Oracle Restart может монтировать дисковые группы, если они указаны как зависимости, даже если они не включены в параметр **ASM_DISKGROUPS**.
Значение по умолчанию для этого параметра отсутствует.
- **ASM_PREFERRED_READ_FAILURE_GROUPS** определяет группы сбоев, которые содержат предпочтительный диск для чтения. Этот параметр следует использовать в расширенных или растянутых кластерных базах данных, которые содержат зеркальные копии данных с одной из копий в непосредственной близости от сервера.
- **DIAGNOSTIC_DEST** определяет местоположение домашнего каталога автоматического репозитория диагностики (ADR). В этом каталоге содержатся файлы трассировки, журналы оповещений, основные файлы и файлы инцидентов. Значение по умолчанию этого параметра извлекается из значения **ORACLE_BASE**.
- **LARGE_POOL_SIZE** определяет размер (в байтах) выделяемой динамической памяти большого пула. Выделяемая динамическая память большого пула используется в разделяемых серверных системах для памяти сеансов путем параллельного выполнения для буферов сообщений, а также с помощью процессов резервного копирования для дисковых буферов ввода/вывода. Экземпляр ASM использует функцию автоматического управления памятью, поэтому этот параметр определяет минимальное значение, до которого можно снизить размер большого пула.
- **REMOTE_LOGIN_PASSWORDFILE** определяет, выполняет ли программное обеспечение Oracle проверку файла паролей. Значение по умолчанию — **EXCLUSIVE**.

Восемь параметров, указанных выше, являются единственными нестандартными параметрами, которые создаются для экземпляра ASM. Экземпляр ASM отличается от экземпляра базы данных тем, что не все параметры базы данных действительны для экземпляра ASM. Приблизительно 74 из всех 344 параметров экземпляра база данных можно использовать для экземпляра ASM. Остальные параметры, не указанные на слайде, можно задать по мере необходимости, хотя для большинства установок достаточно стандартных параметров.

Примечание. Функция автоматического управления памятью для экземпляра ASM включена по умолчанию, даже если параметр **MEMORY_TARGET** не задан явным образом. Это единственный параметр, который необходимо задать для полного управления памятью ASM. Oracle настоятельно рекомендует использовать автоматическое управление памятью для ASM.

Взаимодействие между экземплярами базы данных и ASM



ORACLE

© Oracle, 2009. Все права защищены.

Взаимодействие между экземплярами базы данных и ASM

Процесс создания файла точно иллюстрирует взаимодействия, которые происходят между экземплярами базы данных и ASM. Процесс создания файла происходит следующим образом:

1. База данных запрашивает создание файла.
2. Приоритетный процесс ASM создает запись каталога продолжающихся операций (Continuing Operation Directory — COD) и выделяет пространство для новых файлов в дисковой группе.
3. Процесс базы данных ASMB получает карту экстендов для нового файла.
4. Теперь файл открыт, и процесс базы данных инициализирует файл напрямую.
5. После инициализации процесс базы данных запрашивает завершение процедуры создания файла. Это приводит к тому, что приоритетный процесс ASM очищает запись COD и отмечает файл как созданный.
6. Подтверждение создания файла приводит к неявному закрытию файла. Экземпляр базы данных должны повторно открыть файл для выполнения дальнейших операций ввода-вывода.

Этот пример позволяет подчеркнуть два важных момента касательно архитектуры ASM:

- Экземпляр базы данных и экземпляр ASM координируются для совместной работы. Экземпляр базы данных должны взаимодействовать с ASM для отображения файлов базы данных в экстендах ASM. Экземпляр базы данных также получает постоянный поток сообщений, связанных с операциями ASM (такими, как повторное определение баланса дисковой группы), которые могут блокировать или перемещать экстенды ASM.
- Входные/выходные операции базы данных не осуществляются через экземпляр ASM. В действительности база данных выполняет операции ввода/вывода непосредственно с файлами ASM, как показано в действии 4 на слайде.

Экземпляр ASM: динамические представления производительности

Экземпляр ASM служит для размещения таблиц метаданных на основе памяти, представленных в виде динамических представлений производительности.

- Служебные программы ASM осуществляют доступ для извлечения только метаданных с помощью языка SQL
- Содержит различные специализированные представления, относящиеся к ASM:

V\$ASM_ALIAS	V\$ASM_ATTRIBUTE	V\$ASM_CLIENT
V\$ASM_DISK	V\$ASM_DISK_IOSTAT	V\$ASM_DISK_STAT
V\$ASM_DISKGROUP	V\$ASM_DISKGROUP_STAT	V\$ASM_FILE
V\$ASM_OPERATION	V\$ASM_TEMPLATE	

ORACLE

© Oracle, 2009. Все права защищены.

Экземпляр ASM: динамические представления производительности

Одной из основных функций любого экземпляра является хранения таблиц метаданных на основе памяти. Эти таблицы начинаются с префикса X\$ и обычно не документируются. Серии динамических представлений производительности, которые начинаются с префикса V\$, используются для отображения настроенного представления данных, содержащихся в таблицах памяти X\$. Информация представлена как доступная только для чтения администраторам, обладающим соответствующими полномочиями. Эта информация извлекается из ASM с помощью языка SQL. На представленном выше слайде приведен список наиболее распространенных динамических представлений производительности, которые содержат метаданные, относящиеся к ASM. Существует несколько сотен дополнительных динамических представлений производительности, но большинство из этих представлений будут пустыми, поскольку для их использования необходимо монтирование управляющего файла базы данных экземпляром базы данных. Экземпляры ASM не монтируют управляющие файлы базы данных. Полный список динамических представлений производительности см. в руководстве *Oracle Database Reference 11g Release 2 (11.2)* (Справочное руководство по базе данных Oracle 11g версии 2 (11.2)).

Системные полномочия ASM

- Экземпляр ASM не включает словарь данных, поэтому единственным способом подключения к ASM является использование этих системных полномочий.

Полномочие ASM	Группа полномочий (рекомендуется)	Привилегия
SYSASM	OSASM (asmadmin)	Полные административные полномочия
SYSDBA	OSDBA (asmdba)	Доступ к данным, сохраненным в ASM, и SYSASM в текущей версии
SYSOPER	OSOPER (asmoper)	Ограниченные полномочия для запуска и остановки экземпляра ASM вместе с набором неdestructивных команд ALTER DISKGROUP

- Пользователь SYS автоматически создается с полномочием SYSASM.

ORACLE

© Oracle, 2009. Все права защищены.

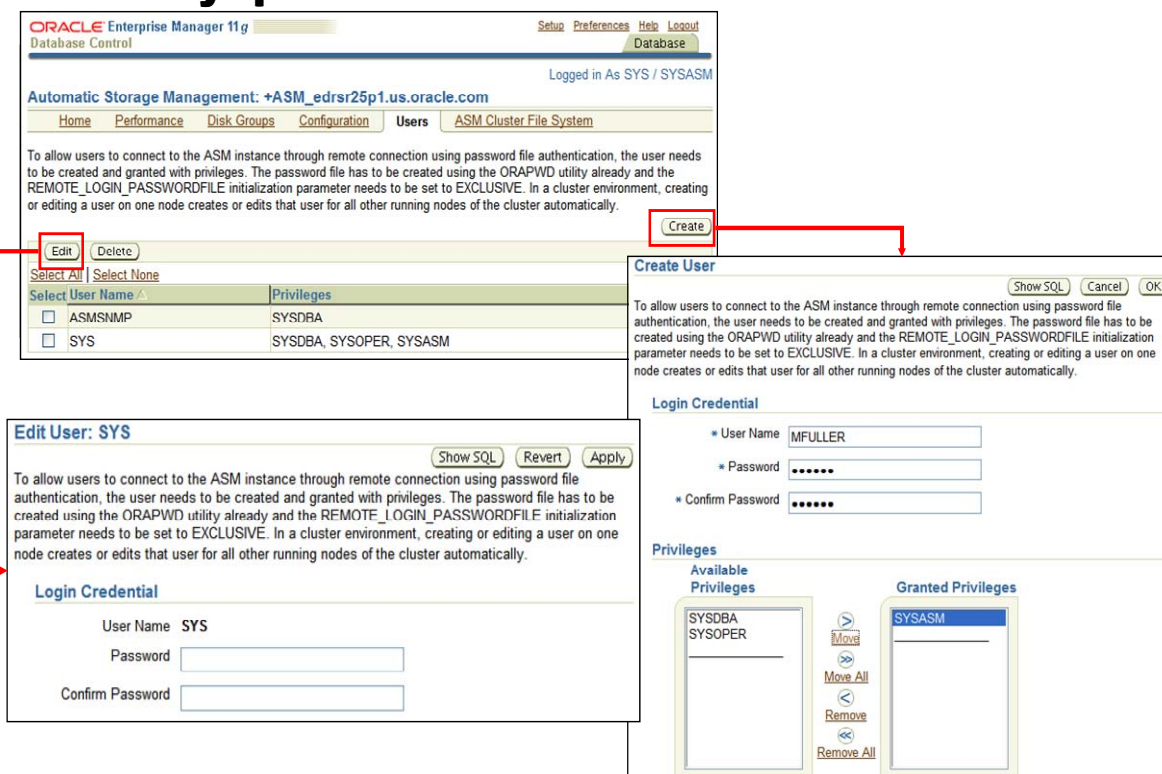
Системные полномочия ASM

Экземпляр ASM не включает словарь данных, поэтому единственным способом подключения к ASM является использование одного из трех системных полномочий: SYSASM, SYSDBA или SYSOPER. Эти системные полномочия ASM представлены в следующем списке.

- **SYSASM:** это полномочие предоставляет полное административное полномочие на экземпляр ASM.
- **SYSDBA:** это полномочие предоставляет доступ к данным, сохраненным в ASM, и в текущей версии предоставляет административные полномочия SYSASM.
- **SYSOPER:** это полномочие предоставляет возможность запуска и остановки экземпляров ASM вместе с набором неdestructивных команд ALTER DISKGROUP. Другие команды, такие как CREATE DISKGROUP, не разрешены.

Если функция ASM установлена, группы операционных систем используются для проверки подлинности полномочий SYSASM, SYSDBA и SYSOPER. Эти группы называются OSASM, OSDBA и OSOPER, соответственно, в служебной программе Oracle Universal Installer (OUI), то есть группа OSASM — это группа операционной системы, которой предоставляется полномочие SYSASM. Рекомендуемые имена групповых значений OSASM, OSDBA и OSOPER — asmadmin, asmdba и asmoper, соответственно. Поэтому SYSASM — это имя, используемое базой данных, OSASM — это имя, используемое служебной программой OUI, а asmadmin — это имя, используемое операционной системой. Все они относятся к одной и той же группе пользователей. После первоначального создания экземпляра ASM sys и asmnsmp являются единственными определенными пользователями ASM.

Использование Enterprise Manager для управления пользователями ASM



ORACLE

© Oracle, 2009. Все права защищены.

Использование Enterprise Manager для управления пользователями ASM

Enterprise Manager позволяет управлять пользователями, которые осуществляют доступ к экземпляру ASM посредством удаленного соединения (путем аутентификации с помощью файла паролей). Эти пользователи резервируются исключительно для экземпляра ASM.

Эта функциональная возможность доступна только при подключении в качестве пользователя SYSASM. Она скрыта, если вы подключены как пользователи SYSDBA или SYSOPER.

- При нажатии кнопки «Create» (Создать) отображается страница «Create User» (Создать пользователя).
- При нажатии кнопки «Edit» (Правка) отображается страница «Edit User» (Изменить пользователя).
- Для удаления созданных пользователей следует нажать кнопку «Delete» (Удалить).

Примечание. Для выполнения входа в ASM с использованием роли SYSASM щелкните ссылку «Preferences» (Параметры) в верхней части страницы, а затем щелкните ссылку «Preferred Credentials» (Предпочтительные учетные данные). Затем в списке целевых типов появится тип «ASM». Щелкните значок «Set Credentials» (Задать учетные данные) рядом с целевым типом ASM, чтобы задать пароль и учетную запись, обладающую полномочием SYSASM. Чтобы изменение вступило в силу, возможно, потребуется выполнить выход и повторных вход в Database Control.

Запуск и остановка экземпляров ASM с помощью SQL*Plus

Процедура использования SQL*Plus для запуска и остановки экземпляров ASM аналогична процедуре запуска и остановки экземпляров базы данных.

```
$ . oraenv
ORACLE_SID = [orcl] ? +ASM
The Oracle base for ORACLE_HOME=/u01/app/oracle/product/11.2.0/grid is
/u01/app/oracle
$ sqlplus / AS SYSASM
SQL*Plus: Release 11.2.0.1.0 - Production on Wed Jul 8 20:46:46 2009
Copyright (c) 1982, 2009, Oracle. All rights reserved.
Connected to an idle instance.
SQL> startup
ASM instance started

Total System Global Area 284565504 bytes
Fixed Size 1336028 bytes
Variable Size 258063652 bytes
ASM Cache 25165824 bytes
ASM diskgroups mounted
ASM diskgroups volume enabled
SQL> shutdown abort
```

ORACLE

© Oracle, 2009. Все права защищены.

Запуск и остановка экземпляров ASM с помощью SQL*Plus

При использовании SQL*Plus пользователь запускает экземпляр ASM с помощью команды **STARTUP** аналогично запуску экземпляра базы данных Oracle. При запуске экземпляра ASM обратите внимание на следующее:

- Для подключения к экземпляру ASM с помощью SQL*Plus задайте для переменной среды **ORACLE_SID** идентификатор ASM SID. Идентификатор ASM SID для базы данных одного экземпляра — **+ASM**, а идентификатор SID ASM для узла Oracle RAC — **+ASMномер_узла**, где *номер_узла* — это номер данного узла. Скрипт **oraenv** задает переменные **ORACLE_BASE**, **ORACLE_SID**, **ORACLE_HOME** и **PATH**.
- Файл параметров инициализации должен содержать следующую запись:
INSTANCE_TYPE = ASM

Этот параметр указывает на то, что запускается экземпляр ASM, а не экземпляр базы данных.

- Когда вы выполняете команду **STARTUP**, а не пытаетесь смонтировать и открыть базу данных, эта команда пытается смонтировать дисковые группы, указанные параметром инициализации **ASM_DISKGROUPS**. Если вы не ввели значение для **ASM_DISKGROUPS**, впоследствии вы можете смонтировать дисковые группы с помощью команды **ALTER DISKGROUP...MOUNT**.

Запуск и остановка экземпляров ASM с помощью SQL*Plus (продолжение)

В следующем списке представлено описание параметров команды STARTUP, имеющих отношение к ASM.

- **FORCE:** подает команду SHUTDOWN ABORT для экземпляра ASM перед его перезапуском.
- **MOUNT или OPEN:** монтирует дисковые группы, указанные в параметре инициализации ASM_DISKGROUPS. Это выполняется по умолчанию, если параметры команды не указаны.
- **NOMOUNT:** запускает экземпляр ASM без монтажа какой-либо дисковой группы.
- **RESTRICT:** запускает экземпляр в ограниченном режиме. Предложение RESTRICT можно использовать в сочетании с предложениями MOUNT, NOMOUNT и OPEN.

В ограниченном режиме экземпляры базы данных не могут использовать дисковые группы. То есть базы данных не могут открывать файлы, которые относятся к этой дисковой группе. Кроме того, если дисковая группа смонтирована экземпляром в ограниченном режиме, эта дисковая группа не может быть смонтирована любым другим экземпляром в кластере. Ограниченный режим позволяет выполнять задачи по обслуживанию в дисковой группе без вмешательства со стороны клиентов. Операции повторного определения баланса, которые выполняются при работе дисковой группы в ограниченном режиме, аннулируют обмен сообщениями карты экстендов для блокировки и разблокировки между экземплярами ASM в кластерной среде. Это оптимизирует общую эффективность операций повторного определения баланса. По завершении периода обслуживания необходимо явным образом размонтировать дисковую группу и повторно смонтировать ее в обычном режиме.

Процесс завершения работы ASM инициируется при выполнении команды SHUTDOWN в SQL*Plus. Перед выполнением этой команды убедитесь, что для переменных среды ORACLE_SID и ORACLE_HOME заданы такие значения, которые позволят подключиться к экземпляру ASM.

Oracle настоятельно рекомендует завершить работу всех экземпляров базы данных, которые используют экземпляр ASM, перед попыткой завершения работы экземпляра ASM.

В следующем списке приведено описание параметров команды SHUTDOWN, которые имеют отношение к ASM.

- **NORMAL:** ASM ожидает завершения любого выполняемого процесса SQL, прежде чем размонтировать все дисковые группы и завершить работу экземпляра ASM. Перед завершением работы экземпляр ASM ожидает отключения всех пользователей, подключенных в настоящее время, от экземпляра. Если какие-либо экземпляры подключены к экземпляру ASM, команда SHUTDOWN возвращает ошибку, а экземпляр ASM продолжает работать. NORMAL — это режим завершения работы по умолчанию.
- **IMMEDIATE or TRANSACTIONAL:** ASM ожидает завершения любого выполняемого процесса SQL, прежде чем размонтировать все дисковые группы и завершить работу экземпляра ASM. ASM не будет ожидать отключения подключенных к экземпляру пользователей. Если какие-либо экземпляры подключены к экземпляру ASM, команда SHUTDOWN возвращает ошибку, а экземпляр ASM продолжает работать.
- **ABORT:** экземпляр ASM незамедлительно завершает работу отключается без последовательного демонтажа дисковой группы. Это приводит к восстановлению при следующем запуске ASM. Если какой-либо экземпляр базы данных подключен к экземпляру ASM, работа экземпляра базы данных прерывается.

Примечание. Формы завершения работы NORMAL, IMMEDIATE и TRANSACTIONAL не применяются при наличии подключенных экземпляров RDBMS. Возвращается следующая ошибка:

```
ORA-15097: cannot SHUTDOWN ASM instance with connect RDBMS instance
```

Запуск и остановка экземпляров ASM с помощью srvctl

Служебную программу Server Control (srvctl) можно использовать для запуска и остановки экземпляров ASM.

```
$ . oraenv
ORACLE_SID = [orcl] ? +ASM
The Oracle base for
  ORACLE_HOME=/u01/app/oracle/product/11.2.0/grid is
  /u01/app/oracle
$ srvctl start asm -o mount
$ srvctl stop asm -f
```

Служебную программу Sever Control (srvctl) можно использовать для проверки состояния экземпляров ASM.

```
$ srvctl status asm
ASM is running on edrsr25p1
```

ORACLE

© Oracle, 2009. Все права защищены.

Запуск и остановка экземпляров ASM с помощью srvctl

Служебную программу Server Control (srvctl) можно использовать для запуска и остановки экземпляров ASM вместе с другими ресурсами, которые управляются распределенной инфраструктурой. Служебную программу `srvctl` можно найти в расположении `ORACLE_HOME/bin` для распределенной инфраструктуры и в местоположении `ORACLE_HOME/bin` установки базы данных. При управлении ASM, прослушивателями или функцией Oracle Restart следует использовать служебную программу `srvctl`, расположенную в разделе `ORACLE_HOME` распределенной инфраструктуры. Служебную программу `srvctl` можно использовать для управления ASM следующими способами:

- Запустите экземпляр ASM.
`srvctl start asm [-o <start_option>]`
`<start_option>` — это одна из действительных функций запуска экземпляра (FORCE, MOUNT, OPEN, NOMOUNT или RESTRICT) (необязательно)
- Остановка экземпляра ASM.
`srvctl stop asm [-o <stop_option>] -f`
`<stop_option>` — одна из действительных функций завершения работы экземпляра (NORMAL, IMMEDIATE, TRANSACTIONAL или ABORT) (необязательно);
-f используется для принудительного выполнения операции
- Создайте отчет о состоянии экземпляра ASM.
`srvctl status asm`

Запуск и остановка экземпляров ASM с помощью asmcmd

Служебная программа asmcmd предоставляет интерфейс командной строки для ASM без использования языка SQL.

```
$ . oraenv
ORACLE_SID = [orcl] ? +ASM
The Oracle base for ORACLE_HOME=/u01/app/oracle/product/11.2.0/grid is
/u01/app/oracle
$ asmcmd
Connected to an idle instance.
ASMCMD> startup
ASM instance started

Total System Global Area  284565504 bytes
Fixed Size                  1336028 bytes
Variable Size              258063652 bytes
ASM Cache                   25165824 bytes
ASM diskgroups mounted
ASM diskgroups volume enabled
ASMCMD> shutdown --abort
ASM instance shut down
Connected to an idle instance.
```

ORACLE

© Oracle, 2009. Все права защищены.

Запуск и остановка экземпляров ASM с помощью asmcmd

Метаданные ASM для администрирования находятся в динамических представлениях производительности, которые содержатся в экземпляре ASM. Доступ к этим представлениям в служебной программе ASM обычно осуществляется с помощью языка SQL. Требование относительно знания языка SQL усложняет курс обучения ASM, переводя его на новый уровень, а обучение языку SQL не входит в список стандартных требований при обучении системных администраторов или администраторов хранилищ. Служебная программа asmcmd предоставляет среду по типу псевдооболочки, которая допускает использование синтаксиса UNIX для выполнения обычных административных задач ASM. Ее можно использовать для управления экземплярами Oracle ASM, дисковыми группами, управления доступом к файлам для дисковых групп, файлами и каталогами в дисковых группах, шаблонами для дисковых групп и томами.

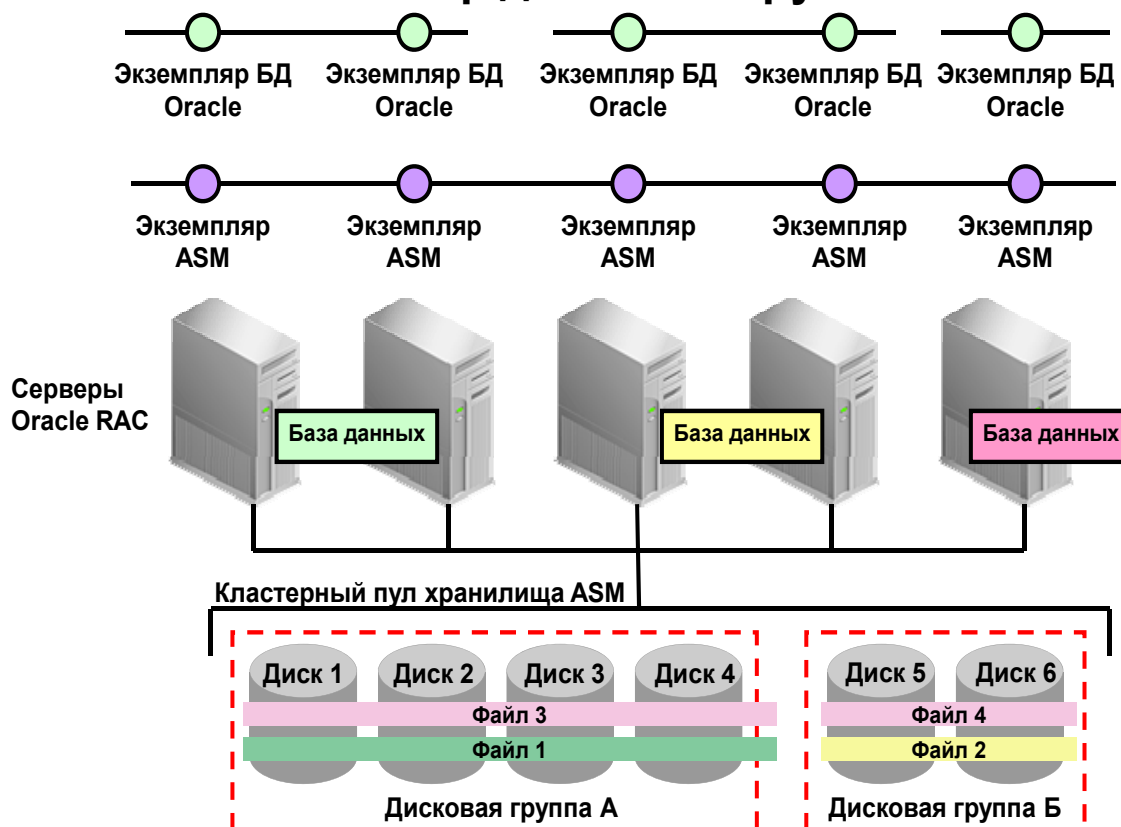
Вы можете использовать служебную программу asmcmd для запуска и завершения работы экземпляра ASM. Поддерживаемые функции запуска:

- nomount (для запуска экземпляра ASM без монтажа дисковых групп)
- restrict (для запуска экземпляра ASM и ограничения использования базы данных)
- pfile <pfile.ora> (для запуска экземпляра ASM с настраиваемым параметром PFILE)

Поддерживаемые функции завершения работы:

- immediate (немедленное завершение работы)
- abort (прерывание всех выполняемых операций)

Обзор дисковых групп



ORACLE

© Oracle, 2009. Все права защищены.

Обзор дисковых групп

Дисковая группа — это логическое объединение одного или нескольких дисков, которыми ASM управляет как коллекцией. Каждая дисковая группа содержит метаданные, связанные с ней. Дисковую группу ASM можно, в принципе, рассматривать как аналог логического тома в стандартной сети хранения данных.

Файлы выделяются из пространства в дисковой группе. Содержимое файлов, которые хранятся в дисковой группе, равномерно распределяется («разносится») по дискам в дисковой группе для устранения горячих точек и обеспечения равномерной работы на дисках. Каждый ASM-файл полностью содержится в одной дисковой группе. Однако дисковая группа может содержать файлы, которые относятся к нескольким базам данных, а одна база данных может использовать различные файлы из множества дисковых групп.

Ключевым атрибутом дисковой группы является параметр избыточности. Существуют три возможных параметра избыточности дисковой группы:

- Внешняя избыточность, когда ASM не обеспечивает какое-либо зеркалирование и диски считаются очень надежными.
- Обычная избыточность, когда ASM поддерживает 2-стороннее зеркалирование по умолчанию в целях обеспечения целостности данных для менее надежных хранилищ.
- Высокая избыточность, когда ASM поддерживает 3-стороннее зеркалирование по умолчанию для еще более надежного обеспечения целостности данных.

ASM поддерживает создание до 63 дисковых групп, однако для большинства установок обычно требуется значительно меньшее количество.

Диски ASM

Диски ASM:

- являются устройствами хранения, которые выделяются для дисковых групп ASM;
- доступны посредством обычных интерфейсов ОС;
- должны быть доступны для чтения и записи владельцем ASM;
- должны быть доступны для всех узлов в кластере;
- могут обладать различными именами ОС или путями в различных узлах;
- могут представлять собой:
 - целый физический диск или раздел на физическом диске;
 - диск или раздел из массива хранения;
 - логические тома (LV) или логические блоки (LUN);
 - сетевые файлы (NFS).

ORACLE

© Oracle, 2009. Все права защищены.

Диски ASM

Дисковые группы ASM содержат один или несколько дисков ASM.

Диски ASM должны быть доступны для чтения и записи владельцем ASM из всех узлов в кластере.

Диск ASM может быть разделом на диске, но Oracle настоятельно рекомендует не допускать совместного использования с другими приложениями физических дисков, используемых ASM. Это связано с тем, что для обеспечения оптимальной работы политики разнесения и зеркалирования ASM производительность подсистем ввода-вывода дисков ASM в дисковой группе должна быть единообразной и стабильной. Совместный доступ к устройствам ASM и других приложений усложняет обеспечение единообразной и стабильной производительности диска.

Не требуется использование одинаковых имен устройств O/S дисков ASM на различных узлах в кластере. ASM определяет членов дисковой группы, считывая заголовки дисков ASM.

Как правило, диски ASM представляют собой неформатированные LUN из массива хранения, предложенного ASM. Кроме того, диски ASM могут также быть файлами на удаленном регистраторе NFS.

Выделенные блоки

ASM-диски разделены на выделенные блоки (AU):

- Размер AU настраивается при создании дисковой группы.
- **Размер AU по умолчанию — 1 Мбайт.**
 - Такой размер достаточно мал для того, чтобы обеспечивать кэширование базой данных, и достаточно велик для эффективного последовательного доступа
- Допустимые размеры AU:
 - 1, 2, 4, 8, 16, 32 или 64 Мбайт
 - Большие AU можно применять при выполнении сценариев сверхбольшой базы данных (VLDB) или при использовании специализированного оборудования для хранения

ORACLE

© Oracle, 2009. Все права защищены.

Выделенные блоки

Пространство на ASM-диске разделено на выделенные блоки (AU). Размер AU по умолчанию составляет один мегабайт, и этого достаточно мало для того, чтобы не создать горячую точку, и достаточно много для того, чтобы обеспечить эффективный последовательный доступ. Размер AU можно задать при создании дисковой группы. Вы не можете изменить размер AU для дисковой группы. Большие AU можно применять при выполнении сценариев сверхбольшой базы данных (VLDB) или при использовании специализированного оборудования для хранения. Если к AU постоянно осуществляется доступ, ядро базы данных кэширует этот блок для более эффективного доступа.

Файлы ASM

Файлы ASM:

- Являются набором экстентов ASM, состоящих из выделенных блоков
 - Экстенты с переменным размером обеспечивают поддержку больших файлов
- Отображаются как обычные файлы для ядра базы данных
- Обладают именами файлов, которые начинаются с «+»
 - Например, +DATA/orcl/datafile/system.256.689832921
- Могут быть связаны с дополнительным псевдонимом файла
 - Например, +DATA/dbfiles/mydb/system01.dbf
- Равномерно распределены по дискам в дисковой группе
- Зеркалируются в соответствии с политиками, определенными в дисковой группе

ORACLE

© Oracle, 2009. Все права защищены.

Файлы ASM

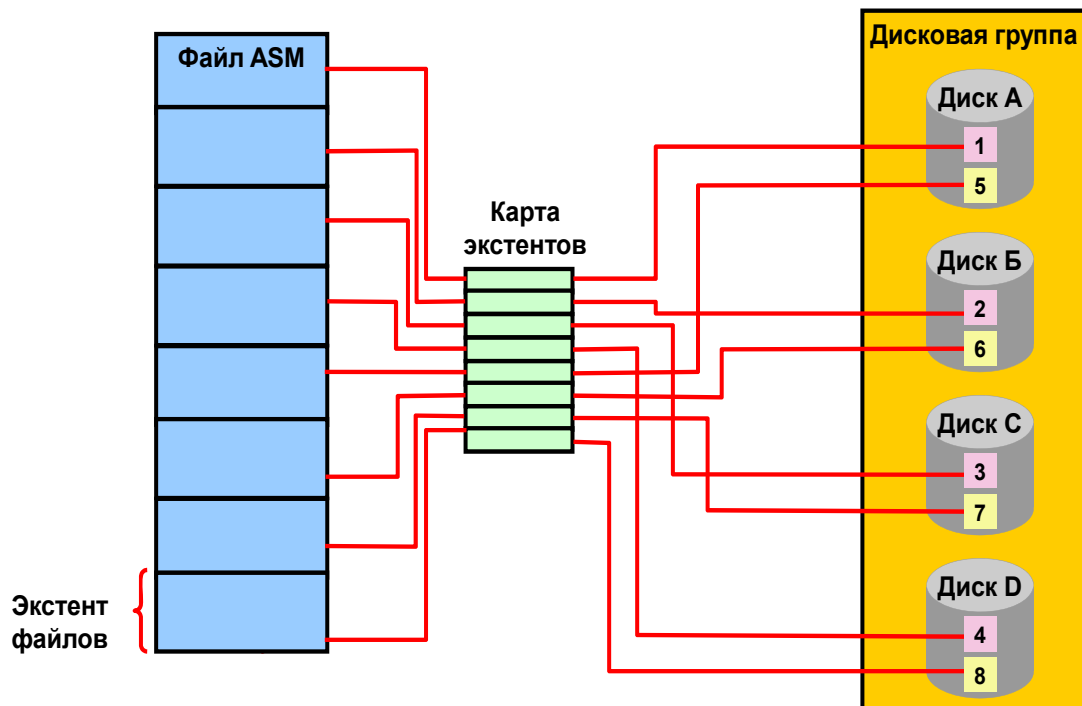
ASM предоставляет набор файлов для использования клиентами ASM. Файл ASM состоит из набора выделенных блоков и отображается как обычный файл для ядра базы данных.

Каждый файл ASM имеет уникальное имя, сгенерированное системой. На слайде показан пример полного имени файла ASM. Полное имя файла ASM представляет собой иерархию, которая начинается с комбинации знака «плюс» и имени дисковой группы. После имени дисковой группы указывается имя базы данных и тип файла. Последний элемент состоит из имени тега, номера файла, номера инкарнации. Можно при необходимости создать псевдоним, чтобы предоставить администраторам более понятный и удобный способ обращения к файлу ASM.

Файлы равномерно распределены по дискам ASM в дисковой группе с помощью политики повсеместного разнесения и зеркалирования (SAME).

ASM обеспечивает непосредственную поддержку большинства типов файлов, которые имеют отношение к базе данных, таких как файлы данных, файлы журналов, управляющие файлы, резервные копии RMAN и т. д. До выпуска Oracle Database 11g версии 2 функция ASM поддерживала только файлы, имеющие отношение к базе данных Oracle, поэтому ее нельзя было использовать для хранения и управления файлами трассировки и журналами предупреждений ASCII, двоичными файлами Oracle, Oracle Cluster Registry (OCR) и диском выбора кластеров. Oracle Database 11g версия 2 устраняет это ограничение, предоставляя средства запуска файловой системы общего назначения посредством ASM.

Карты экстентов



ORACLE

© Oracle, 2009. Все права защищены.

Карты экстентов

ASM отслеживает формат файла с помощью метаданных, которые называются картой экстентов. Карта экстентов представляет собой таблицу, которая сопоставляет экстенды данных в файле с выделенными блоками на диске.

Между экстендами файла и выделенными блоками существует следующее взаимоотношение. Экстент содержит:

- Один AU для первых 20 000 экстентов (0–19 999)
- Четыре AU для следующих 20 000 экстентов (20 000–39 999)
- Шестнадцать AU для экстентов после 40 000

Экстенды с переменным размером в сочетании с большими выделенным блоками можно использовать для размещения очень больших файлов ASM.

Детализация размещения

ASM разделяет разнесение для балансировки нагрузки и разнесение для задержки:

- Крупномодульное разнесение концентрирует **выделенные блоки для балансировки нагрузки.**

– Пример:



ORACLE

© Oracle, 2009. Все права защищены.

Детализация размещения

В ASM разнесение преследует две основные цели:

- Балансировка нагрузки ввода/вывода на всех дисках в дисковой группе.
- Улучшение задержки I/O.

Крупномодульное разнесение распределяет выделенные блоки по дискам в дисковой группе. Это обеспечивает балансировку нагрузки для дисковых групп. При выделении файла ASM равномерно распределяет выделенные блоки по всем дискам. Иногда распределение не может быть абсолютно равномерным, но со временем будет наблюдаться тенденция к выравниванию. На указанной выше схеме представлен файл с пятью выделенными блоками, разнесенными по пяти дискам во внешней дисковой группе избыточности, которая в сумме содержит восемь дисков.

Для первых 20 000 экстенгов размер экстенга равен размеру AU. Для экстенгов с 20 000 до 40 000 всегда выделяются 8 наборов экстенгов за раз, причем размер экстенга равен $4 \times \text{размер выделенного блока}$. Если размер AU составляет 1 Мбайт, это означает, что размер файла ASM будет увеличиваться на 32 Мбайт за раз ($8 \times 4 \times 1 \text{ Мбайт}$). Если для файла применяется крупномодульное разнесение, он разносится по 8 наборам экстенгов с разнесением по 1 выделенному блоку. Разнесение всегда выполняется на уровне AU, а не на уровне экстенга. Таким образом, каждый выделенный блок файла с крупномодульным разнесением находится на другом диске, отличном от диска, на котором размещался предыдущий выделенный блок данного файла, вне зависимости от размера файла. Для экстенгов после 40 000 по-прежнему выделяются 8 экстенгов за раз, но размер экстенга равен $16 \times \text{размер выделенного блока}$.

Детализированное разнесение

При детализированном разнесении разнесенные блоки размером 128 Кбайт распределяются по группам выделенных блоков для улучшения задержки.

- Дисковая группа с 8 дисками и внешней избыточностью
- Размер AU по умолчанию — 1 Мбайт
- Первый экстенст 1 Мбайт записывается как разнесения 128 Кбайт по 8 выделенным блокам



ORACLE

© Oracle, 2009. Все права защищены.

Детализированное разнесение

Детализированное разнесение разделяет экстенсты данных по разделам размером 128 Кбайт и предоставляется для улучшения задержки для определенных типов файлов путем распределения нагрузки для каждого экстенста на ряде дисков. Детализированное разнесение используется по умолчанию для управляющих файлов и интерактивных файлов журнала повтора.

В схеме на этой странице представлен принцип выполнения детализированного разнесения. В этом примере первый экстенст нового файла размером 1 Мбайт занимает первые 128 Кбайт 8 различных выделенных блоков, распределенных по восьми дискам в дисковой группе. Следовательно, операции чтения или записи одного мегабайта распространяются по восьми дискам вместо одного.

Детализированное разнесение

Пример

- Дисковая группа с 8 дисками и внешней избыточностью
- Размер AU по умолчанию — 1 Мбайт
- Следующий экстенст 1 Мбайт записывается как разнесения 128 Кбайт по тем же 8 выделенным блокам, пока они не будут заполнены



ORACLE

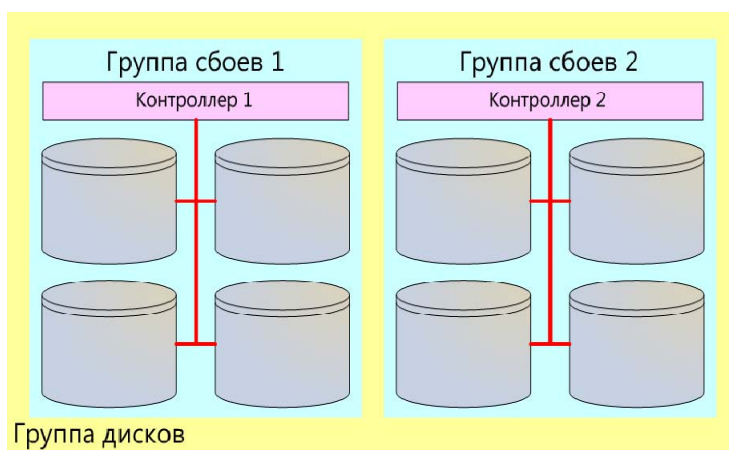
© Oracle, 2009. Все права защищены.

Детализированное разнесение (продолжение)

Продолжая предыдущий пример, следующий экстенст размером один мегабайт распределяется по вторым 128 килобайтам каждого из этих выделенных блоков. Эта схема выполняется до тех пор, пока первый набор выделенных блоков не будет заполнен и не будет выделен другой набор.

Группы отказа ASM

- Набор дисков, использующих общие ресурсы, отказ которых следует допускать
- Зеркалированные копии экстенгов хранятся в отдельных группах отказа
- **Оборудование для хранения определяет границы групп отказа**
 - Пример основан на выделении дисковых контроллеров:



ORACLE

© Oracle, 2009. Все права защищены.

Группы отказа ASM

В рамках дисковой группы диски могут быть собраны в группы отказа. Группы отказа — это способ хранения; администратор базы данных указывает аппаратные границы, с учетом которых выполняется зеркалирование ASM.

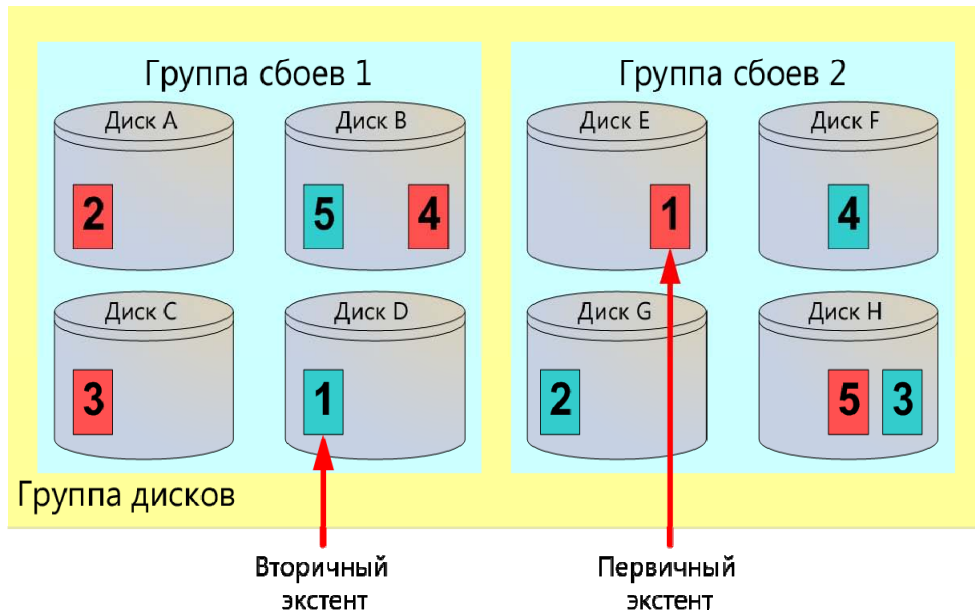
Например, все диски, подключенные к одному дисковому контроллеру, должны быть указаны как относящиеся к общей группе отказа. Это приводит к зеркалированию экстенгов файлов на дисках, подключенных к отдельным контроллерам. Кроме того, администратор может настроить ASM для выбора политики группы отказа по умолчанию. Политика по умолчанию заключается в том, что каждый отдельный диск относится к своей собственной группе отказа.

Вы можете группировать диски в группы отказа, используя любые критерии по мере необходимости. Группы отказа можно использовать для защиты от отказа отдельных дисков, дисковых контроллеров, сетевых компонентов ввода/вывода и даже целых систем хранения данных. Как правило, администратор должен анализировать среду хранения данных и упорядочивать группы отказа для устранения последствий определенных сценариев сбоя.

Администратор базы данных или хранилища должен определить, какая конфигурация группы отказа лучше всего подходит для данной установки.

Пример разнесения и зеркалирования

Обычная дисковая группа избыточности, включающая в сумме восемь дисков, распределяется по двум группам отказа.



ORACLE

© Oracle, 2009. Все права защищены.

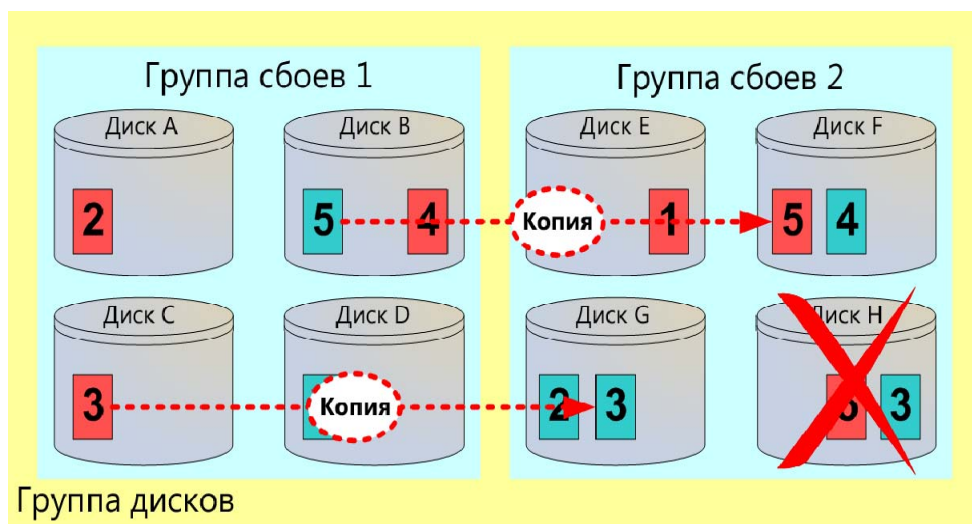
Пример разнесения и зеркалирования

Схема на данной странице — это пример разнесения и зеркалирования в обычной дисковой группе избыточности. Красные блоки — это файл с пятью экстентами, которые разносятся по пяти из восьми дисков в дисковой группе. Синие блоки — это зеркалированные копии экстентов файла. Обратите внимание, что вне зависимости от распределения экстентов по различным дискам и группам отказа каждый экстенст обладает ровно одной копией в каждой группе отказа.

При выделении файла основные экстенты выделяются для обеспечения производительности, а дополнительные копии — для обеспечения целостности. По этой причине все операции считывания базы данных по умолчанию выполняются для основных экстентов.

Пример отказа

Если происходит отказ диска Н, содержащиеся на нем экстененты создаются повторно на работающих дисках из сохранившихся зеркал.



ORACLE

© Oracle, 2009. Все права защищены.

Пример отказа

Давайте расширим пример, представленный на предыдущей странице, и представим, что произошел отказ диска Н и содержащиеся на нем данные более не доступны.

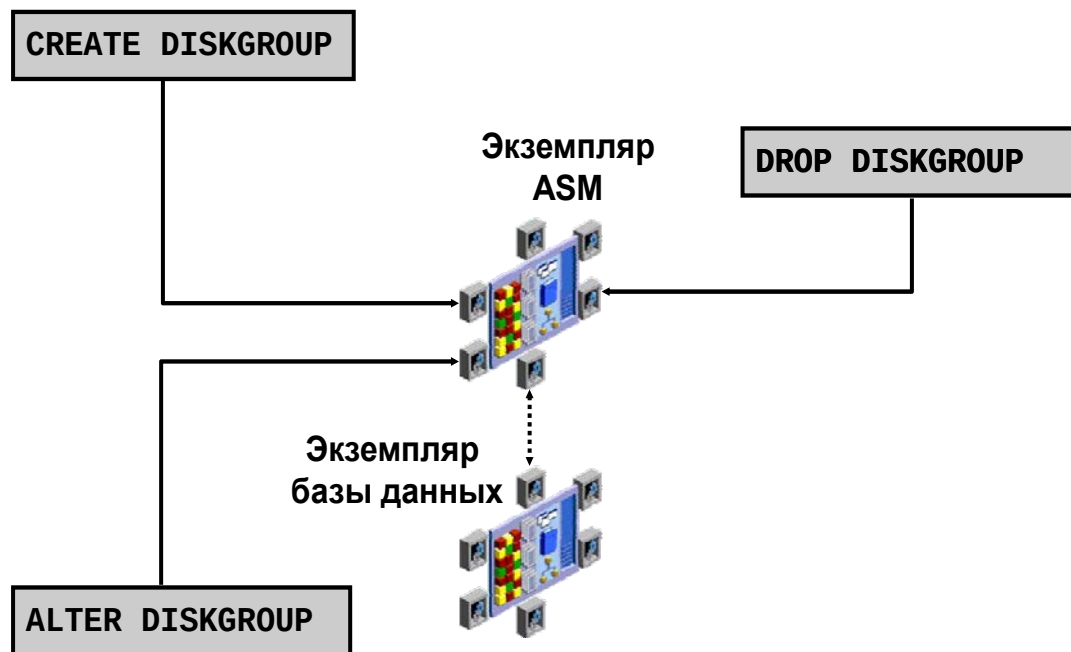
При возникновении такого отказа необходимо, чтобы все экстененты, содержащиеся на отказавшем диске, были восстановлены и скопированы на другой диск.

Экстененты «три» и «пять» копируются из оставшейся копии в свободную область на другом диске в одну и ту же группу отказа. В этом примере экстенент «пять» копируется с диска В на диск F, а диск «три» копируется с диска С на диск G.

Последнее действие, происходящее после отказа диска, заключается в том, что ASM удаляет отказавший диск из дисковой группы.

Удаление диска по сути инициирует ту же самую процедуру, одна в этом случае экстененты, содержащиеся на удаляемом диске, сначала копируются в доступное альтернативное местоположение.

Управление дисковыми группами



ORACLE

© Oracle, 2009. Все права защищены.

Управление дисковыми группами

Основная цель экземпляра ASM заключается в управлении дисковыми группами и защите данных. Экземпляры ASM также сообщают структуру файлов экземплярам базы данных. Таким образом, экземпляры базы данных могут напрямую обращаться к файлам, хранящимся в дисковых группах.

Существует несколько административных команд дисковой группы. Для всех них требуется полномочие **SYSASM** или **SYSDBA**, и все они должны подаваться экземпляром ASM. Вы можете добавлять новые дисковые группы. Можно также изменять существующие дисковые группы для добавления новых дисков, удаления существующих и выполнения множества других операций. Вы можете удалять существующие дисковые группы.

Создание и удаление дисковых групп с помощью SQL*Plus

```
$ . oraenv
ORACLE_SID = [orcl] ? +ASM
The Oracle base for ORACLE_HOME=/u01/app/oracle/product/11.2.0/grid is
/u01/app/oracle
$ sqlplus / AS SYSASM
SQL*Plus: Release 11.2.0.1.0 - Production on Wed Jul 8 20:46:46 2009
Copyright (c) 1982, 2009, Oracle. All rights reserved.
..
SQL> CREATE DISKGROUP dgroupA NORMAL REDUNDANCY
FAILGROUP controller1 DISK
    '/devices/A1' NAME diskA1 SIZE 120G FORCE,
    '/devices/A2',
FAILGROUP controller2 DISK
    '/devices/B1',
    '/devices/B2';
```

```
SQL> DROP DISKGROUP dgroupA INCLUDING CONTENTS;
```

ORACLE

© Oracle, 2009. Все права защищены.

Создание и удаление дисковых групп

Предположим, что при выполнении операции обнаружения дисков ASM в каталоге /dev были обнаружены следующие диски: A1, A2, B1 и B2. Также предположим, что диски A1 и A2 находятся на других дисковых контроллерах отдельно от дисков B1 и B2. В первом примере на слайде показано, каким образом следует сконфигурировать дисковую группу с именем DGROUPA с двумя группами отказа: CONTROLLER1 и CONTROLLER2.

В примере для дисковой группы также используется атрибут избыточности по умолчанию — NORMAL REDUNDANCY. Можно по желанию также указать имя и размер диска. Если эта информация не указана, ASM создает имя по умолчанию и пытается определить размер диска. Если не удастся определить размер, возвращается ошибка. FORCE указывает на то, что данный диск следует добавить в указанную дисковую группу, хотя диск уже отформатирован как член дисковой группы ASM. В случае использования функции FORCE для диска, который не отформатирован как член дисковой группы ASM, возвращается ошибка.

Вы можете удалить дисковую группу со всеми файлами, как показывает вторая инструкция, представленная на слайде. Во избежание случайного удаления следует указать функцию INCLUDING CONTENTS, если дисковая группа все еще содержит какие-либо файлы помимо внутренних метаданных ASM. Для удаления необходимо смонтировать дисковую группу. После проверки закрытия всех файлов дисковых групп группа и все диски удаляются из дисковой группы. Затем выполняется перезапись заголовка каждого диска для удаления информации о форматировании ASM.

Добавление дисков в дисковые группы

```
ALTER DISKGROUP dgroupA ADD DISK  
  '/dev/sde1' NAME A5,  
  '/dev/sdf1' NAME A6,  
  '/dev/sdg1' NAME A7,  
  '/dev/sdh1' NAME A8;
```

```
ALTER DISKGROUP dgroupA ADD DISK '/devices/A*';
```

Форматирование диска

Определение повторного баланса дисковой группы

ORACLE

© Oracle, 2009. Все права защищены.

Добавление дисков в дисковые группы

В этом примере показана процедура добавления дисков в дисковую группу. Вы выполняете команду `ALTER DISKGROUP ADD DISK`, чтобы добавить диски. Первая инструкция добавляет четыре новых диска в дисковую группу `DGROUPA`.

Вторая инструкция демонстрирует взаимодействие строк обнаружения.. Представьте следующую конфигурацию:

`/devices/A1` является членом дисковой группы `DGROUPA`.

`/devices/A2` является членом дисковой группы `DGROUPA`.

`/devices/A3` является членом дисковой группы `DGROUPA`.

`/devices/A4` является диском-кандидатом.

Вторая команда добавляет `A4` в дисковую группу `DGROUPA`. Другие диски игнорируются, несмотря на то что они соответствуют критериям обнаружения. Это связано с тем, что они уже входят в дисковую группу `DGROUPA`. На схеме показано, что при добавлении диска в дисковую группу экземпляр ASM проверяет возможность обращения к диску и его использования. Затем выполняется форматирование и повторное определение баланса. Процесс повторного определения баланса занимает много времени, так как он переносит экстенды из всех файлов на новый диск.

Примечание. Процесс повторного определения баланса не блокирует какие-либо операции базы данных. Основное влияние процесс повторного определения баланса оказывает на нагрузку ввода/вывода системы. Чем выше мощность процесса повторного определения баланса, тем больше нагрузка ввода/вывода. Следовательно, для операций ввода/вывода базы данных доступна меньшая полоса пропускания ввода/вывода.

Прочие команды ALTER

Удаление диска из dgroupA:

```
ALTER DISKGROUP dgroupA DROP DISK A5;
```

Добавление и удаление диска с помощью одной команды:

```
ALTER DISKGROUP dgroupA  
DROP DISK A6  
ADD FAILGROUP controller3  
DISK '/dev/sdi1' NAME A9;
```

Отмена операции удаления диска:

```
ALTER DISKGROUP dgroupA UNDROP DISKS;
```

ORACLE

© Oracle, 2009. Все права защищены.

Прочие команды ALTER

Первая инструкция, представленная на слайде, представляет процесс удаления одного из дисков из дисковой группы DGROUPA. Вторая инструкция показывает процесс добавления и удаления диска с помощью одной команды. Значительным преимуществом в этом случае является то, что процесс повторного определения баланса начинается только после завершения выполнения команды. Третья инструкция показывает процесс отмены операции удаления диска. Команда UNDROP применяется только к дискам, ожидающим удаления; она не влияет на завершённые операции удаления.

Следующая инструкция позволит повторно определить баланс дисковой группы DGROUPB по мере необходимости:

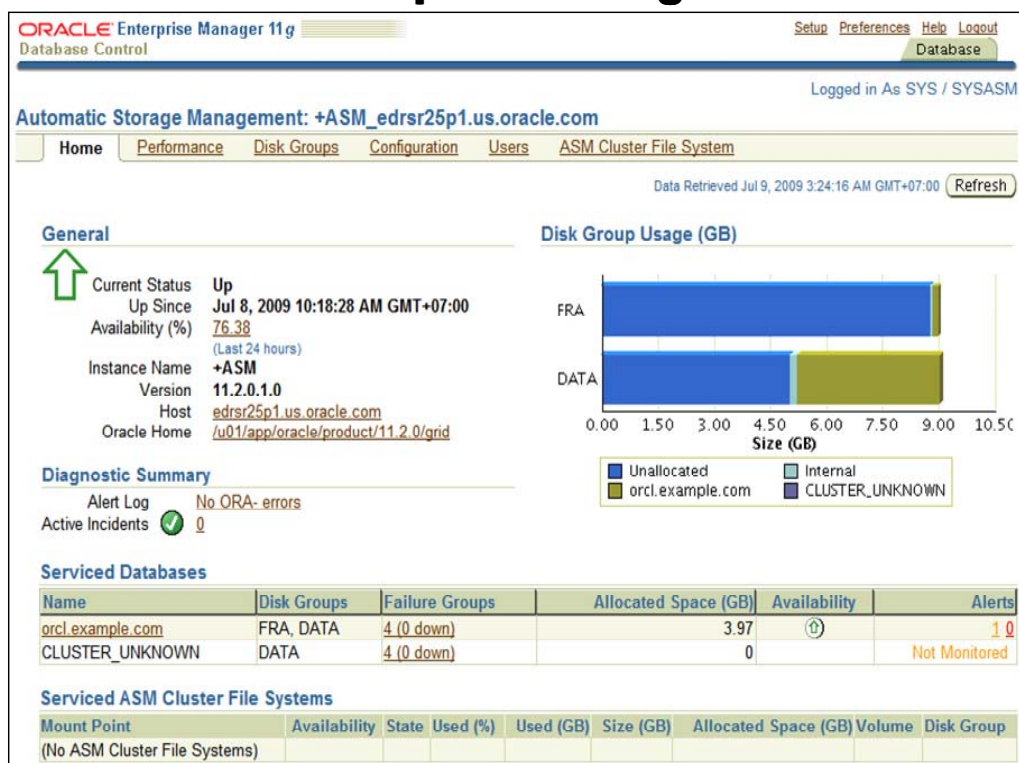
```
ALTER DISKGROUP dgroupB REBALANCE POWER 5;
```

Эта команда обычно не требуется, поскольку данная процедура выполняется автоматически при добавлении, удалении или изменении размера дисков. Однако она может потребоваться, если необходимо использовать предложение POWER для переопределения скорости по умолчанию, которая задается параметром инициализации ASM_POWER_LIMIT. Чтобы изменить уровень мощности текущей операции повторного определения баланса, следует повторно ввести команду с указанием нового уровня. Установка нулевого уровня мощности приводит к тому, что операция повторного определения баланса останавливается, пока команда не будет вызвана повторно явным или неявным образом. Следующая инструкция размонтирует DGROUPA:

```
ALTER DISKGROUP dgroupA DISMOUNT;
```

Параметры MOUNT и DISMOUNT позволяют предоставлять или запрещать доступ экземпляров базы данных к одной или нескольким дисковым группам. Возможность монтирования и размонтирования вручную является полезной в кластеризованной среде ASM, поддерживающей один экземпляр, если произошел сбой этого экземпляра на другом узле.

Управление ASM с помощью Enterprise Manager



ORACLE

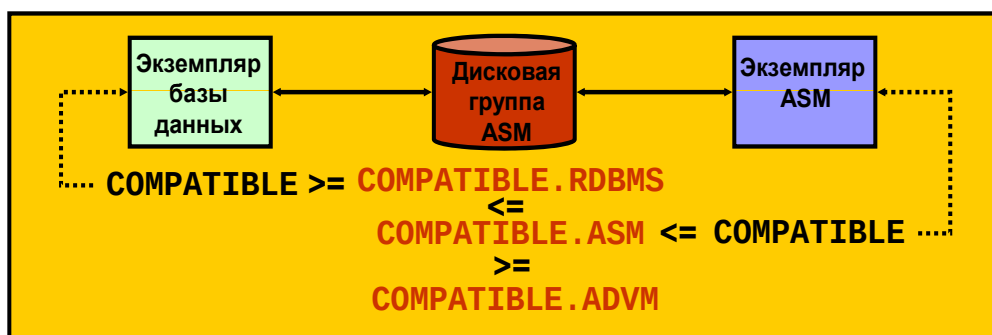
© Oracle, 2009. Все права защищены.

Управление ASM с помощью Enterprise Manager

Oracle Enterprise Manager (EM) — это семейство средств управления компании Oracle, охватывающее базы данных, промежуточное ПО, приложения, сети, ИТ-инфраструктуру и многое другое. Enterprise Manager предоставляет среду на основе браузера, которая является эффективной альтернативой для быстрого выполнения наиболее распространенных задач администрирования ASM.

Совместимость дисковой группы ASM

- Совместимость каждой дисковой группы контролируется независимо от других дисковых групп:
 - Совместимость ASM управляет дисковой структурой метаданных ASM.
 - Совместимость СУРБД управляет минимальным уровнем клиента потребителя.
 - Совместимость ADVM определяет, может ли дисковая группа содержать тома Oracle ASM.
- Процесс определения совместимости дисковой группы является необратимым.



ORACLE

© Oracle, 2009. Все права защищены.

Совместимость дисковой группы ASM

Существуют три вида совместимости, относящиеся к дисковым группам ASM: обработка постоянных структур данных, которые описывают дисковую группу, возможности клиентов (потребителей дисковых групп), а также возможность содержания томов в дисковой группе. Эти атрибуты называются *совместимостью ASM*, *совместимостью СУРБД* и *совместимостью ADVM*, соответственно. Совместимость каждой дисковой группы контролируется независимо от других дисковых групп. Это необходимо для поддержки разнородных сред с дисковыми группами и из Oracle Database 10g, и из Oracle Database 11g. Эти три параметра совместимости являются атрибутами каждой дисковой группы ASM:

- Совместимость СУРБД относится к минимальной совместимой версии экземпляра СУРБД, которая допускает монтирование дисковой группы экземпляром. Эта совместимость определяет формат сообщений, которыми обмениваются экземпляры ASM и базы данных (СУРБД). Экземпляр ASM может поддерживать различные клиенты СУРБД, работающие с различными параметрами совместимости. Значение параметра версии, совместимой с базой данных, каждого экземпляра должно превышать или равняться значению параметра совместимости СУРБД всех дисковых групп, используемых этой базой данных. Экземпляры базы данных, как правило, запускаются посредством домашней страницы Oracle, отличной от страницы запуска экземпляра ASM. Это означает, что экземпляр базы данных может работать под управлением другой версии программного обеспечения. При первом подключении экземпляра базы данных к экземпляру ASM согласовывается максимальная версия, которую поддерживают оба экземпляра.

Совместимость дисковой группы ASM (продолжение)

Настройка параметра совместимости базы данных, версия программного обеспечения базы данных и настройка совместимости СУРБД дисковой группы определяет, может ли экземпляр базы данных монтировать данную дисковую группу.

- Совместимость ASM относится к настройке постоянной совместимости, которая управляет форматом структур данных для метаданных ASM на диске. Уровень совместимости ASM дисковой группы должен всегда превышать уровень совместимости СУРБД этой же дисковой группы или соответствовать ему. Совместимость ASM относится только к формату метаданных ASM. Формат содержимого файла зависит от экземпляра базы данных. Например, для параметра совместимости ASM дисковой группы можно установить значение 11.0, тогда как для параметра совместимости СУРБД может быть установлено значение 10.1. Это означает, что дисковая группа может находиться под управлением только программного обеспечения ASM версии 11.0 или выше, в то время как любой клиент базы данных, использующий программное обеспечение версии 10.1 или выше, может использовать эту дисковую группу.
- Атрибут совместимости ADVM определяет, может ли дисковая группа содержать тома Oracle ASM в дисковой группе. Необходимо задать значение 11.2 или выше. Перед заданием этого атрибута следует установить для `COMPATIBLE .ASM` значение 11.2 или выше. Кроме того, необходимо загрузить драйверы томов ADVM.

Уровень совместимости дисковой группы необходимо повышать только в том случае, если произошло изменение постоянных дисковых структур или процедур обмена сообщениями по протоколу. Однако повышение уровня совместимости дисковой группы — это необратимая операция. Для задания совместимости дисковой группы можно использовать команду `CREATE DISKGROUP` или `ALTER DISKGROUP`.

Примечание. Помимо параметров совместимости дисковой группы параметр *compatible* (параметр *compatible version* базы данных) определяет включенные функции; он применяется к экземпляру базы данных или экземпляру ASM в зависимости от параметра *instance_type*. Например, установка значения 10.1 делает невозможным использование любых функций, реализованных в Oracle Database 11g (оперативный/автономный режим работы дисков, экстенды переменных и т. д.).

Атрибуты дисковой группы ASM

Имя	Свойство	Значения	Описание
au_size	Создать, изменить	1 2 4 8 16 32 64MB	Размер выделенных дисков в дисковой группе
compatible.rdbms	Создать, изменить	Действительная версия база данных	Формат сообщений, передаваемых между базой данных и ASM
compatible.asm	Создать, изменить	Действительная версия экземпляра ASM	Формат структур метаданных ASM на диске
compatible.advm	Создать, изменить	Действительная версия экземпляра ASM	Допускает тома Oracle ASM в дисковой группе
disk_repair_time	Создать, изменить	0 М до 2 ³² D	Период времени перед удалением диска после перехода в автономный режим (OFFLINE)
template.tname. redundancy	Изменить	UNPROTECT MIRROR HIGH	Избыточность указанного шаблона
template.tname. stripe	Изменить	COARSE FINE	Разнесение атрибута указанного шаблона

```
CREATE DISKGROUP DATA2 NORMAL REDUNDANCY
DISK '/dev/sda1', '/dev/sdb1'
ATTRIBUTE 'compatible.asm'='11.2';
```

ORACLE

© Oracle, 2009. Все права защищены.

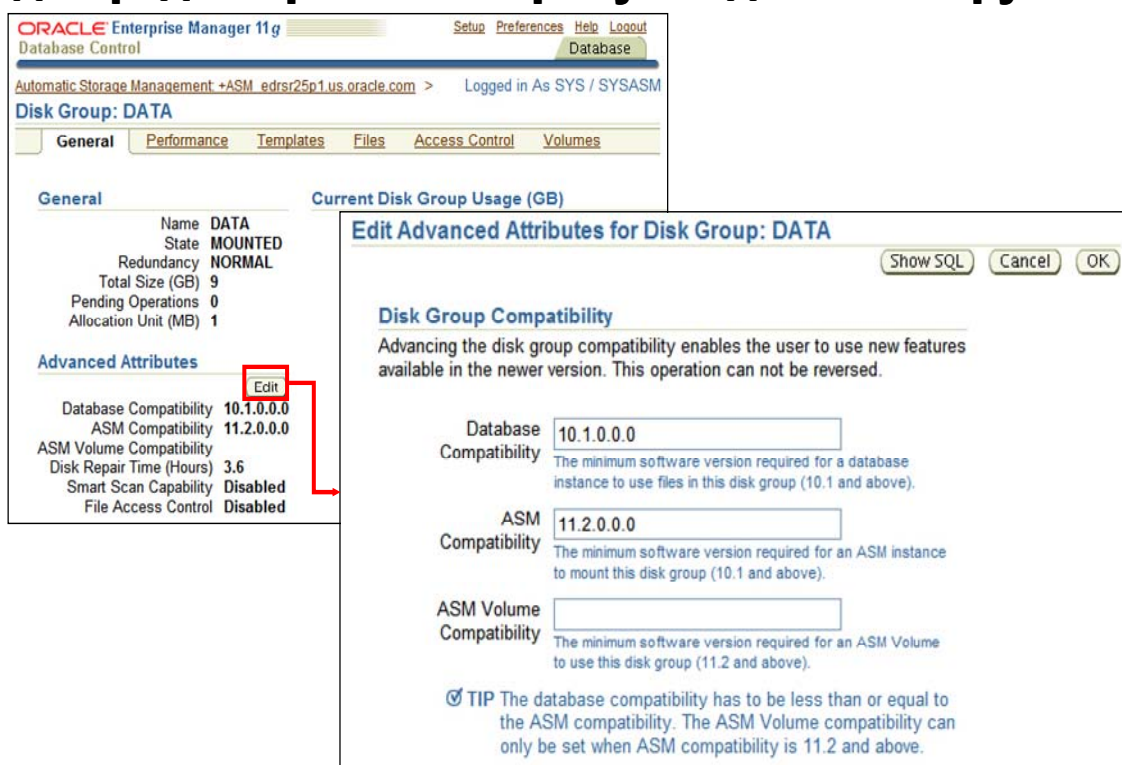
Атрибуты дисковой группы ASM

Когда вы создаете или изменяете дисковую группу ASM, вы можете изменить ее атрибуты с помощью нового предложения **ATTRIBUTE** команд **CREATE DISKGROUP** и **ALTER DISKGROUP**. Краткое описание этих атрибутов приведено в таблице, которая представлена на слайде:

- ASM позволяет использовать различные размеры выделенных блоков (AU), которые указываются при создании дисковой группы. Размер выделенных блоков может составлять 1, 2, 4, 8, 16, 32 или 64 Мбайта.
- Совместимость RDBMS: для получения дополнительной информации см. раздел «Совместимость дисковой группы ASM».
- Совместимость ASM: для получения дополнительной информации см. раздел «Совместимость дисковой группы ASM».
- Значение для параметра **DISK_REPAIR_TIME** можно указать в следующих единицах: минуты (мин.), часы (ч) или дни (дн.). Если вы не укажете единицы, по умолчанию будут использоваться часы. Если вы не укажете атрибут, по умолчанию используется значение «3,6 ч». Вы можете переопределить этот атрибут с помощью инструкции **ALTER DISKGROUP ... DISK OFFLINE**.
- Можно также указать атрибут избыточности указанного шаблона.
- Можно также указать атрибут разнесения указанного шаблона.

Примечание. Для каждой указанной дисковой группы можно просмотреть все заданные атрибуты с помощью фиксированного представления **V\$ASM_ATTRIBUTE**.

Использование диспетчера Enterprise Manager для редактирования атрибутов дисковой группы



ORACLE

© Oracle, 2009. Все права защищены.

Использование диспетчера Enterprise Manager для редактирования атрибутов дисковой группы

Enterprise Manager предоставляет простой способ хранения и извлечения параметров среды, связанных с дисковой группой.

Вы можете задать совместимые атрибуты на странице «Create Disk Group» (Создание дисковой группы) и «Edit Disk Group Advanced Attributes» (Редактирование расширенных атрибутов дисковых групп). Атрибут `disk_repair_time` добавляется только на страницу редактирования расширенных атрибутов дисковых групп.

Примечание. Для экземпляров ASM до версии 11g ASM для параметром совместимости **ASM** и совместимости клиента по умолчанию используется значение 10.1. Для экземпляров ASM версии 11g по умолчанию используется значение параметра совместимости ASM 11.2 и значение параметра совместимости базы данных 10.1.

Извлечение метаданных ASM

- С помощью SQL*Plus:

```
SQL> SELECT f.type, f.redundancy, f.striped, f.modification_date,  
a.system_created, a.name FROM v$asm_alias a, v$asm_file f WHERE  
a.file_number = f.file_number and a.group_number = f.group_number  
and type='DATAFILE';
```

TYPE	REDUND	STRIPE	MODIFICAT	S	NAME
DATAFILE	MIRROR	COARSE	08-JUL-09	Y	SYSTEM.256.689832921
DATAFILE	MIRROR	COARSE	08-JUL-09	Y	SYSAUX.257.689832923
..					

- С помощью asmcmd:

```
ASMCMD> ls -l +DATA/orcl/datafile
```

Type	Redund	Striped	Time	Sys	Name
DATAFILE	MIRROR	COARSE	JUL 08 21:00:00	Y	SYSTEM.256.689832921
DATAFILE	MIRROR	COARSE	JUL 08 21:00:00	Y	SYSAUX.257.689832923
..					

ORACLE

© Oracle, 2009. Все права защищены.

Извлечение метаданных ASM

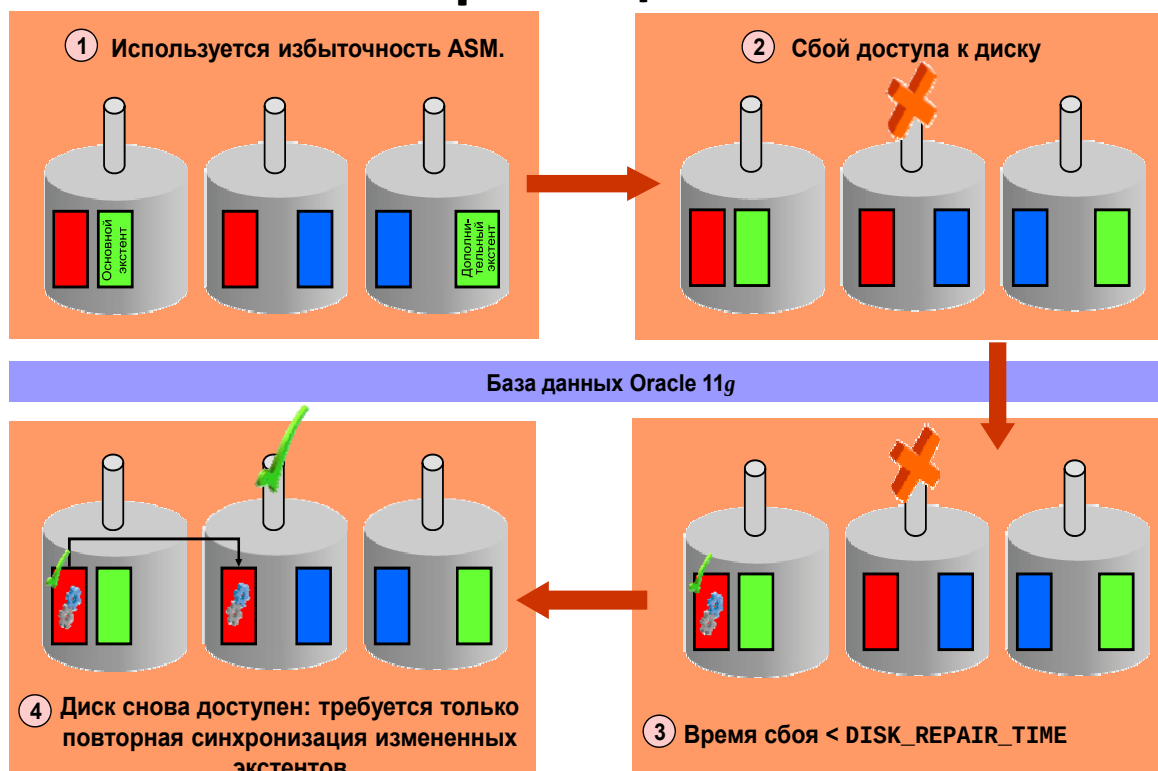
Экземпляр ASM служит для размещения таблиц метаданных на основе памяти, представленных в виде динамических представлений производительности. Эти данные можно запросить с помощью SQL*Plus, asmcmd или Enterprise Manager.

Для использования SQL*Plus требуется знание языка SQL; при этом может потребоваться объединение нескольких динамических представлений производительности для получения соответствующей информации. В первом примере на слайде представлено объединение v\$asm_file и v\$asm_alias для отображения метаданных, относящихся к файлам данных базы данных. Если этот запрос выполняется для экземпляра ASM, возможно получение файлов данных из нескольких баз данных в зависимости от способа оформления синтаксиса. Необходимо использовать дополнительные условия фильтрации для ограничения выводимых результатов только одной базой данных.

Служебная программа asmcmd обладает возможностью подключения к экземпляру ASM и получения метаданных без знания языка SQL. Она использует систему обозначений, аналогичную системе обозначений UNIX. Во втором примере на этом слайде показано использование asmcmd для получения тех же метаданных, что и в примере SQL. Еще одним преимуществом данного примера является то, что выходные данные ограничиваются файлами данных только из одной базы данных, так как указан путь, содержащий имя базы данных orcl, а также тип файла datafile. Поэтому для содержимого, отображаемого как категории в asmcmd, требуются условия фильтрации SQL с использованием предложения WHERE для получения аналогичного результата.

Примечание. Служебная программа Enterprise Manager Database Control позволяет отобразить большую часть метаданных ASM путем перехода между различными веб-страницами ASM.

Обзор быстрой зеркальной повторной синхронизации ASM



ORACLE

© Oracle, 2009. Все права защищены.

Обзор быстрой зеркальной повторной синхронизации ASM

Быстрая зеркальная повторная синхронизация ASM значительно сокращает время, необходимое для повторной синхронизации неустойчивого отказа диска. Когда диск переходит в автономный режим вследствие неустойчивого отказа, ASM отслеживает экстенды, которые были изменены во время простоя. После устранения неустойчивого отказа ASM можно быстро ресинхронизировать только дисковые экстенды ASM, которые были затронуты во время простоя.

При использовании этой функции предполагается, что содержимое затронутых дисков ASM не было повреждено или изменено.

После отказа пути к диску ASM диск ASM переходит в автономный режим, но не удаляется, если был задан атрибут `DISK_REPAIR_TIME` для соответствующей дисковой группы. Значение этого атрибута определяет длительность отключения диска, допускаемую ASM при сохранении возможности повторной синхронизации по завершении ремонта.

Примечание. Механизм отслеживания использует один бит для каждого измененного выделенного блока. Это гарантирует крайнюю эффективность механизма отслеживания.

Экспресс-тест

Какой параметр требуется для экземпляра ASM?

1. INSTANCE_TYPE
2. ASM_DISKGROUPS
3. LARGE_POOL_SIZE
4. Ни один вариант из вышеуказанных

ORACLE

© Oracle, 2009. Все права защищены.

Ответ: 1

Экспресс-тест

Детализированное разнесение по умолчанию используется для _____ и _____.

1. файлов данных
2. **управляющие файлы**
3. временных файлов
4. оперативных журналов повторов
5. SPFILE

ORACLE

© Oracle, 2009. Все права защищены.

Ответы: 2, 4

Заключение

В ходе прохождения урока вы научились:

- управлять экземпляром ASM с помощью SQL*plus, asmcmd и Enterprise Manager;
- **создавать и удалять дисковые группы ASM;**
- определять атрибуты совместимости ASM;
- расширять дисковые группы ASM;
- сравнивать методы извлечения метаданных ASM.

ORACLE

© Oracle, 2009. Все права защищены.

Обзор упражнения 5: управление экземпляром ASM

В данном упражнении рассматриваются следующие темы:

- создание дисковой группы ASM с помощью `asmcmd`;
- удаление дисковой группы ASM с помощью EM;
- просмотр метаданных ASM.

ORACLE

© Oracle, 2009. Все права защищены.

6

Настройка сетевой среды Oracle

ORACLE

© Oracle, 2009. Все права защищены.

Цели

По завершении данного урока вы сможете:

- Использовать Enterprise Manager для:
 - создания дополнительных прослушивателей
 - **создания псевдонимов служб Oracle Net Service**
 - настройки обработки отказов (failover) для времени соединения
 - управления прослушивателем Oracle Net
- Использовать `tnsping` для проверки связности сети **Oracle Net**
- Определять, когда следует использовать разделяемые серверы, а когда — выделенные серверы

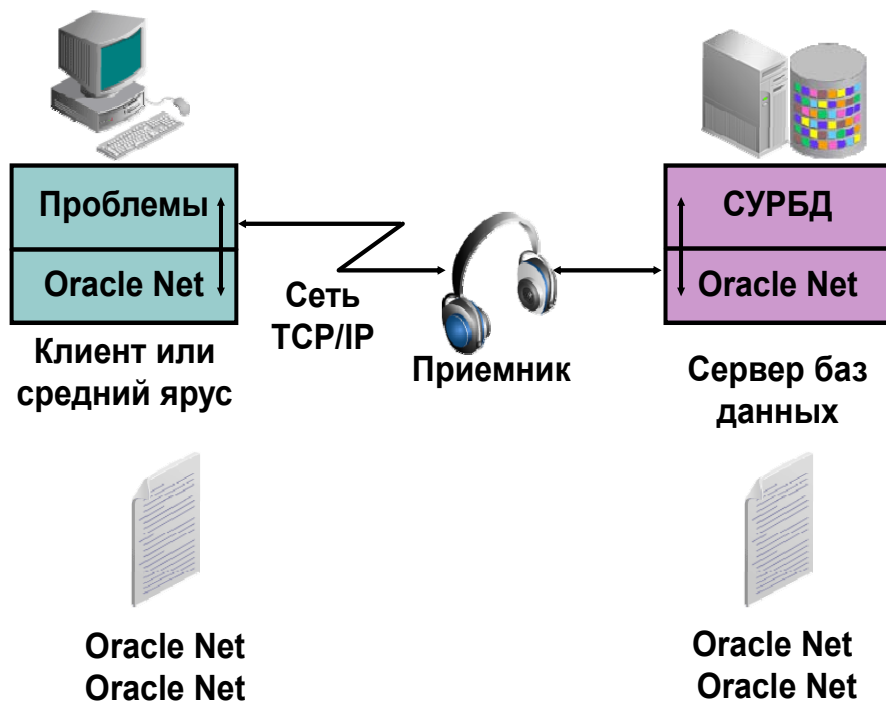
ORACLE

© Oracle, 2009. Все права защищены.

Ресурсы

- *Руководство администратора Oracle Database Net Services*
- *Справочная документация по службам Oracle Database Net Services*

Oracle Net Services



ORACLE

© Oracle, 2009. Все права защищены.

Oracle Net Services

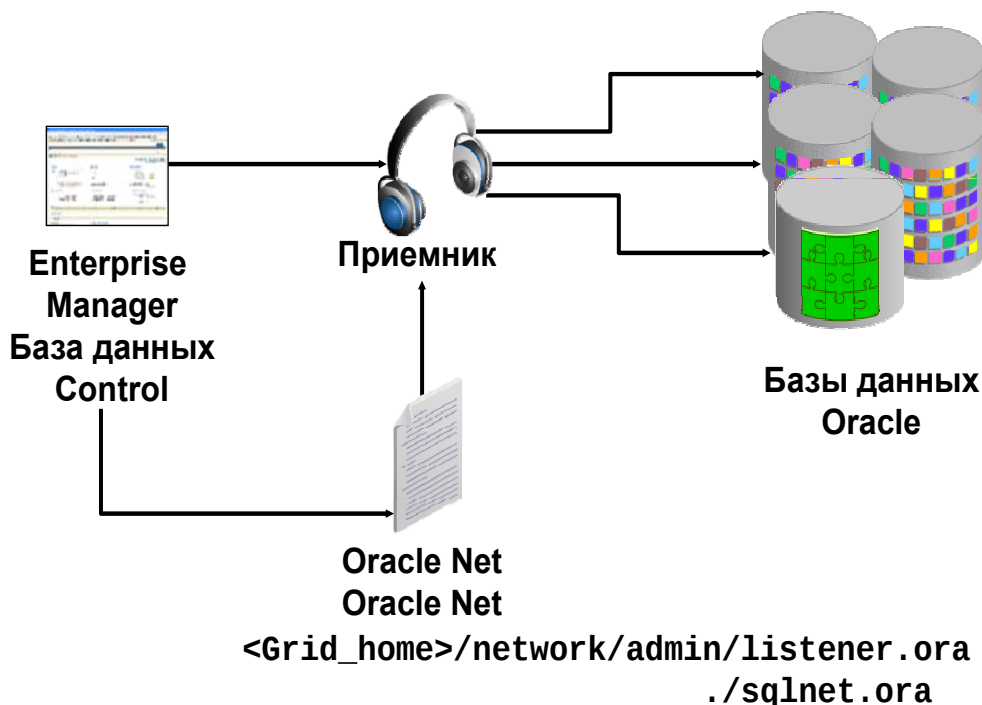
Службы Oracle Net Services обеспечивают сетевые соединения между клиентом или приложением среднего яруса и сервером Oracle. После установления сетевого соединения Oracle Net действует как «курьер», передающий данные между клиентским приложением и сервером баз данных. Он устанавливает и обеспечивает соединение между клиентским приложением и сервером баз данных, а также передает сообщения между ними. Oracle Net (или другой компонент, выполняющий функции Oracle Net, например Java Database Connectivity) размещается на каждом из компьютеров, которым необходимо обращаться к серверу базы данных.

На клиентском компьютере Oracle Net в фоновом режиме обеспечивает соединения с базой данных.

На сервере баз данных Oracle Net запускает активный процесс прослушивателя *Oracle Net Listener*, который контролирует соединения базы данных с внешними приложениями.

Чаще всего Oracle Net Services используется для приема входящих соединений с базой данных. Можно настроить дополнительные сетевые службы, чтобы разрешить доступ к внешним библиотекам кода (EXTPROC) и подключать экземпляр Oracle к другим источникам данных (не Oracle, а, например, Sybase, Informix, DB2 или SQL Server) с помощью гетерогенных служб Oracle.

Прослушиватель Oracle Net



ORACLE

© Oracle, 2009. Все права защищены.

Прослушиватель Oracle Net

Прослушиватель Oracle Net (или просто *прослушиватель*) — это шлюз экземпляра Oracle, который обрабатывает все внешние (нелокальные) соединения пользователей. Один прослушиватель может одновременно обслуживать несколько экземпляров базы данных и тысячи клиентских соединений.

Использование Enterprise Manager — один из способов доступа к прослушивателю. Можно контролировать конфигурацию текущего прослушивателя, а также любые общие параметры, например защиту паролем или пути к файлам журнала.

Опытные администраторы также могут настроить Oracle Net Services, при необходимости вручную изменив файлы конфигурации, с помощью стандартного текстового редактора операционной системы — `vi` or `gedit`.

Примечание. Если установлена распределенная инфраструктура для автономного сервера, прослушиватель Oracle Net Listener запускается из каталога установки программы, который называется `<Grid_home>`. Прослушиватель должен выполняться из этого каталога установки программы, чтобы обеспечить возможность подключения экземпляра ASM. Он также используется по умолчанию для прослушивания всех экземпляров базы данных, установленных на этом же сервере.

Установка сетевых соединений

Чтобы установить соединение с клиентом или средним ярусом с помощью Oracle Net, клиенту должны быть известны:

- **имя хоста, на котором работает прослушиватель;**
- **имя протокола, за которым следит прослушиватель;**
- **имя протокола, который используется прослушивателем;**
- **имя службы, которую обрабатывает прослушиватель.**



ORACLE

© Oracle, 2009. Все права защищены.

Установка сетевых соединений

Чтобы приложение могло установить соединение со службой с помощью прослушивателя Oracle Net, оно должно располагать информацией о данной службе, к которой относятся: адрес или имя хоста, на котором размещен прослушиватель, протокол, который принимает прослушиватель, а также порт, за которым прослушиватель следит. Если прослушиватель определен, приложению остается сообщить имя службы, с которой необходимо установить соединение.

Разрешение имен Oracle Net — это процесс определения вышеупомянутой информации о соединении.

Установка соединения



ORACLE

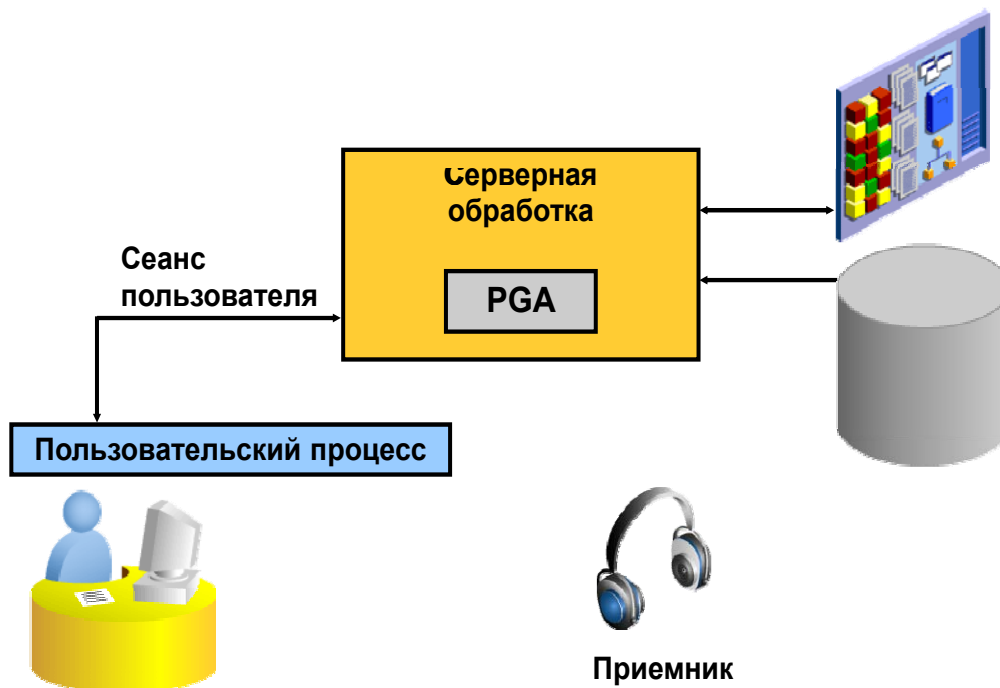
© Oracle, 2009. Все права защищены.

Установка соединения

После завершения разрешения имен Oracle Net запрос соединения передается от приложения пользователя или среднего яруса (далее *процесс пользователя*) прослушивателю. Прослушиватель принимает пакет **CONNECT** и проверяет, запрашивает ли данный пакет **CONNECT** действительное имя службы Oracle Net.

Если имя службы не запрашивается (например, в случае запроса `tnsping`), прослушиватель подтверждает запрос соединения и больше операций не выполняет. При запросе неверного имени службы прослушиватель передает пользовательскому процессу код ошибки.

Сеансы пользователей



ORACLE

© Oracle, 2009. Все права защищены.

Сеансы пользователей

Если в пакете **CONNECT** запрашивается допустимое имя службы, прослушиватель порождает новый процесс для обеспечения соединения. Этот процесс называется *серверным процессом*. Прослушиватель подключается к данному процессу и передает данные инициализации, включая информацию об адресе для пользовательского процесса. После этого прослушиватель уже не обрабатывает соединение, этим занимается серверный процесс.

Серверный процесс проверяет реквизиты для аутентификации (обычно пароль), и если они действительны, создается сеанс пользователя.

Выделенный серверный процесс: после создания сеанса серверный процесс выступает на сервере в качестве агента пользователя. Серверный процесс служит для:

- синтаксического разбора и выполнения SQL-операторов, переданных посредством приложения;
- проверки буферного кэша базы данных на наличие блоков данных, которые необходимы для выполнения SQL-операторов;
- считывания нужных блоков данных из файлов данных на диске в буферный кэш базы данных в SGA (если блоки еще не присутствуют в SGA);
- управления сортировкой. Область сортировки — это область памяти, которая используется для сортировки и располагается в части памяти, связанной с глобальной программной областью (PGA);
- возвращения результатов в такой форме, в которой приложение сможет обрабатывать данные;
- считывания параметров аудита и отчета процессу пользователя об адресате аудита.

Средства настройки и управления сетью Oracle

- Страница администрирования Enterprise Manager Net Services
- Oracle Net Manager
- **Oracle Net Configuration Assistant**
- Командная строка



ORACLE

© Oracle, 2009. Все права защищены.

Средства настройки и управления сетью Oracle

Для управления и настройки сети Oracle используются следующие средства и приложения:

- **Enterprise Manager:** предоставляет интегрированную среду для конфигурирования и управления службами Oracle Net Services. С помощью Enterprise Manager можно настроить службы Oracle Net Services для любого домашнего каталога Oracle на нескольких файловых системах, а также управлять прослушивателями.
- **Oracle Net Manager:** предоставляет графический интерфейс пользователя (GUI), позволяющий настроить Oracle Net Services для домашнего каталога Oracle на локальном клиентском или серверном хосте. С помощью Oracle Net Manager можно настроить Oracle Net Services для домашнего каталога Oracle на локальном клиентском или серверном хосте. Oracle Net Manager можно использовать для настройки следующих компонентов:
 - **Разрешение имен:** определение простых имен, идентификаторов соединения, а также их отображения в дескрипторы соединения, что позволяет определять сетевое расположение и идентифицировать службы. Oracle Net Manager поддерживает настройку дескрипторов соединения в локальных файлах `tnsnames.ora` или централизованной службе каталогов.
 - **Методы разрешения имен:** настройка различных способов разрешения идентификаторов соединений в дескрипторы соединений.

Средства настройки и управления сетью Oracle (продолжение)

- **Профили:** настройка предпочтений для включения и конфигурирования компонентов Oracle Net на стороне клиента или сервера.
- **Прослушиватели:** создание и конфигурирование прослушивателей для приема пользовательских соединений.
- **Oracle Net Configuration Assistant:** запускается программой Oracle Universal Installer при установке программного обеспечения Oracle. Oracle Net Configuration Assistant позволяет настроить адрес протокола прослушивателя и сведения о службах для базы данных Oracle. При обычной установке базы данных Oracle Net Configuration Assistant автоматически настраивает прослушиватель LISTENER с адресом прослушивания протокола TCP/IP для базы данных. При нестандартной установке Oracle Net Configuration Assistant предлагает настроить имя прослушивателя и выбрать адрес протокола. Oracle Net Configuration Assistant следует использовать для исходной настройки сети после установки базы данных. Следовательно, конфигурирование и администрирование сетей можно осуществлять с помощью Oracle Enterprise Manager и Oracle Net Manager.
- **Командная строка:** используется для запуска, остановки и просмотра состояния процесса прослушивателя. Запустить или остановить прослушиватель может пользователь ОС (в данном курсе это oracle). Если прослушиватель не запущен, Enterprise Manager использовать невозможно.

Служебная программа управления прослушивателями

Для управления прослушивателями Oracle Net можно использовать программу командной строки `lsnrctl` (также используется в EM).

```
$ . oraenv
ORACLE_SID = [orcl] ? +ASM
$ lsnrctl
LSNRCTL for Linux: Version 11.2.0.1.0 - Production on 30-JUN-2009 00:47:01
Copyright (c) 1991, 2009, Oracle. All rights reserved.
Welcome to LSNRCTL, type "help" for information.

LSNRCTL> help
The following operations are available
An asterisk (*) denotes a modifier or extended command:
start          stop          status
services       version       reload
save_config    trace        spawn
change_password quit         exit
set*           show*
```

ORACLE

© Oracle, 2009. Все права защищены.

Служебная программа управления прослушивателями

После запуска экземпляра процесс прослушивателя устанавливает канал связи с БД Oracle. После этого прослушиватель может принимать запросы на соединение с базой данных.

Служебная программа управления прослушивателями позволяет контролировать прослушиватели. С помощью `lsnrctl` можно:

- запустить прослушиватель;
- остановить прослушиватель;
- проверить состояние прослушивателя;
- инициализировать прослушиватель повторно с помощью параметров файла конфигурации;
- динамически настраивать несколько прослушивателей;
- изменить пароль прослушивателя.

Служебная программа использует следующий синтаксис команд:

```
LSNRCTL> команда [имя_прослушивателя]
```

Если выполнить команду `lsnrctl`, не указав определенный прослушиватель или не выполнив команду `SET CURRENT_LISTENER`, будет обработан прослушиватель по умолчанию (с именем `LISTENER`). Если необходимо обработать прослушиватель `LISTENER`, аргумент *имя_прослушивателя* можно не указывать. На слайде показан пример допустимых команд `lsnrctl`.

Примечание. Служебная программа `lsnrctl` расположена и на домашней странице распределенной инфраструктуры и на домашней странице базы данных Oracle. Перед использованием необходимо задать для переменных среды соответствующую домашнюю страницу.

Синтаксис служебной программы управления прослушивателями

Команды управления прослушивателями можно выполнять в командной строке или строке ввода LSNRCTL.

- Синтаксис командной строки:

```
$ lsnrctl <имя команды>
$ lsnrctl start
$ lsnrctl status
```

- Синтаксис строки ввода:

```
LSNRCTL> <имя команды>
LSNRCTL> start
LSNRCTL> status
```

ORACLE

© Oracle, 2009. Все права защищены.

Синтаксис служебной программы управления прослушивателями

Команды служебной программы `lsnrctl` можно выполнять в строке ввода самой служебной программы (синтаксис строки ввода) или в командной строке. Ниже показаны две одинаковые команды, но в одной используется синтаксис командной строки, а в другой — строки ввода служебной программы.

Синтаксис командной строки:

```
$ lsnrctl start
```

Синтаксис строки ввода:

```
$ lsnrctl
LSNRCTL for Linux: Version 11.2.0.1.0 - Production on 30-JUN-
2009 01:00:01
Copyright (c) 1991, 2009, Oracle. Все права защищены.
Welcome to LSNRCTL, type "help" for information.
LSNRCTL> start
```

Синтаксис командной строки обычно используется для выполнения отдельных команд или команд сценария. Если требуется выполнить несколько команд `lsnrctl` подряд, синтаксис строки ввода будет удобнее. Обратите внимание на отсутствие аргумента `listener_name` — команда остановки обработает прослушиватель с именем `LISTENER`. Также синтаксис строки ввода используется при защите прослушивателя паролем.

Синтаксис служебной программы управления прослушивателями (продолжение)

Важно помнить, что если используется не прослушиватель LISTENER, а прослушиватель с другим именем, необходимо добавить его имя в команду или же воспользоваться командой SET CURRENT_LISTENER. Предположим, используется прослушиватель custom_lis. Вот два примера остановки прослушивателя custom_lis с помощью строки ввода:

```
LSNRCTL> stop custom_lis
Connecting to
(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=host01)(PORT=55
21)))
The command completed successfully
```

Выполнение следующих команд приведет к тому же результату:

```
LSNRCTL> set cur custom_lis
Current Listener is custom_lis
LSNRCTL> stop
Connecting to
(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=host01)(PORT=55
21)))
The command completed successfully
```

Примечание. В предыдущем синтаксисе current_listener сокращается до cur.

Использование командной строки принесет тот же результат:

```
$ lsnrctl stop custom_lis
LSNRCTL for Linux: Version 11.2.0.1.0 - Production on 30-
JUN-2009 01:01:53
Copyright (c) 1991, 2009, Oracle. Все права защищены.
Connecting to
(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=host01)(PORT=55
21)))
The command completed successfully
```

Использование SRVCTL для запуска и остановки прослушивателя

Если программа перезапуска Oracle Restart настроена на отслеживание прослушивателя, следует использовать SRVCTL для управления этим прослушивателем.

- **Пример синтаксиса:**

```
$ srvctl -h
$ srvctl start listener
$ srvctl stop listener
$ srvctl start listener -l mylistener
$ srvctl status listener
```

ORACLE

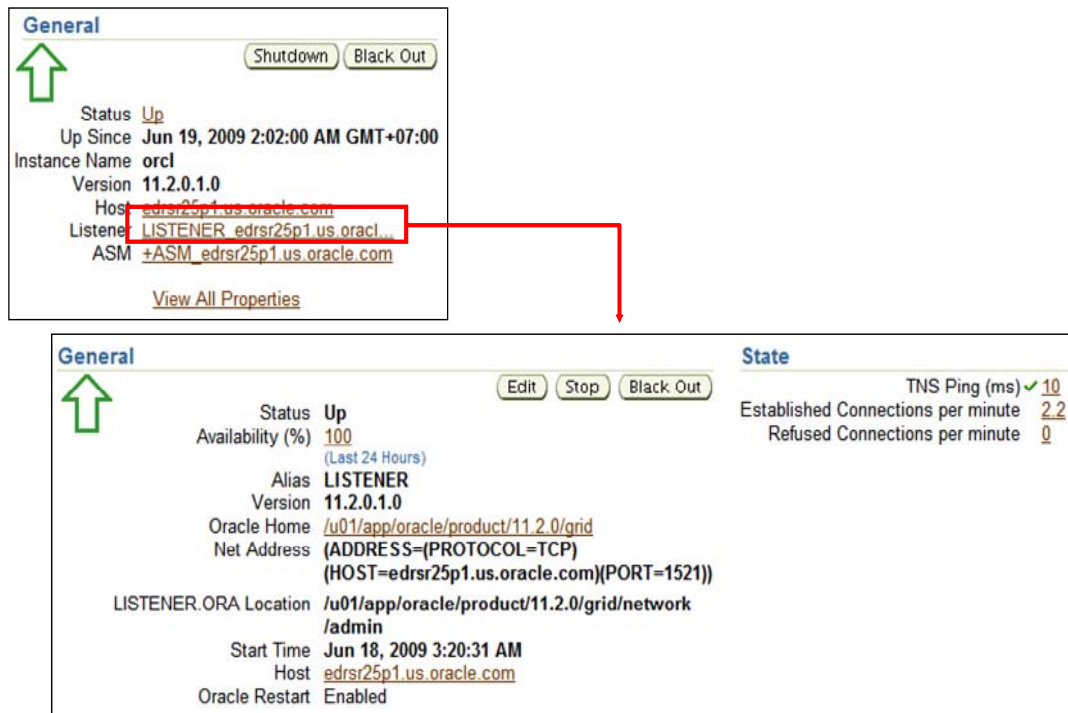
© Oracle, 2009. Все права защищены.

Использование SRVCTL для запуска и остановки прослушивателя

Можно запустить, остановить и просмотреть состояние любого прослушивателя, который управляется Oracle Restart, с помощью SRVCTL. Например:

- Для отображения справки с помощью команд, доступных в SRVCTL: `srvctl -h`
- Для запуска прослушивателя по умолчанию: `srvctl start listener`
- Для остановки прослушивателя по умолчанию: `srvctl stop listener`
- Для запуска прослушивателя с именем mylistener: `srvctl start listener -l mylistener`
- Для отображения состояния прослушивателя по умолчанию: `srvctl status listener`

Домашняя страница прослушивателя



ORACLE

© Oracle, 2009. Все права защищены.

Домашняя страница прослушивателя

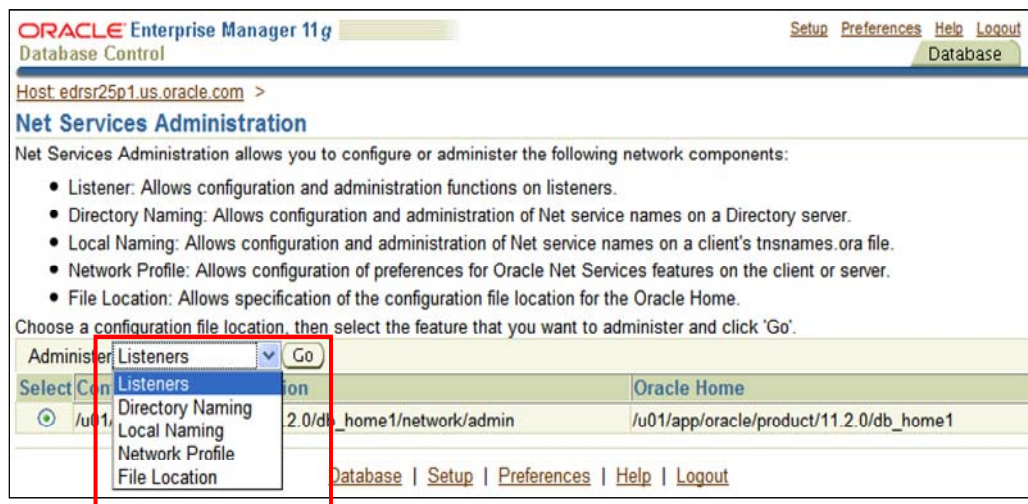
На домашней странице базы данных Enterprise Manager откройте ссылку «Listener» (Прослушиватель) для доступа к домашней странице прослушивателя.

На этой странице можно просмотреть:

- состояние и доступность прослушивателя за последние 24 часа;
- версию прослушивателя и домашний каталог Oracle;
- первый адрес для прослушивания;
- расположение файлов конфигурации, используемых для запуска прослушивателя;
- сведения о времени запуска и хосте прослушивателя;
- состояние Oracle Restart.

Чтобы запустить прослушиватель, перейдите к домашней странице БД и щелкните имя прослушивателя, чтобы открыть его домашнюю страницу. Нажмите кнопку «Stop» (Остановить), чтобы остановить прослушиватель, если он запущен, или же «Start» (Запустить) для запуска незапущенного прослушивателя. Выполните вход в систему хоста в качестве пользователя ОС с правами на запуск и остановку прослушивателей.

Страница администрирования Net Services



ORACLE

© Oracle, 2009. Все права защищены.

Страница администрирования Net Services

На странице администрирования Net Services можно настроить Oracle Net Services для любых домашних каталогов Oracle, размещенных в нескольких файловых системах. Также на ней доступны стандартные функции администрирования прослушивателей. Средства администрирования Net Services позволяют настроить и администрировать следующие компоненты:

- **Прослушиватели:** добавление, удаление, запуск и остановка прослушивателя, а также изменение его характеристик трассировки и протоколирования. Также можно просмотреть отчет о состоянии управления прослушивателем.
- **Разрешение имен на сервере каталогов:** определение простых имен, идентификаторов соединения, а также их отображения в дескрипторы соединения, что позволяет определять сетевое расположение и идентифицировать службы. Хранение служб БД, служб Net Services и псевдонимов Net Services в централизованной службе каталогов.
- **Локальное разрешение имен:** хранение имен Net Services в файле `tnsnames.ora`.
- **Профили:** настройка параметров `sqlnet.ora`.
- **Расположение файлов:** изменение расположения файлов конфигурации Net Services.

Создание прослушивателя

Net Services Administration

Net Services Administration allows you to:

- Listener: Allows configuration and
- Directory Naming: Allows configura
- Local Naming: Allows configura
- Network Profile: Allows configura
- File Location: Allows specification

Choose a configuration file location, then:

Administer: **Listeners** Go

Select Configuration File Location

Configuration File Location	Oracle Home
/u01/app/oracle/product/11.2.0/db_home1/network/admin	/u01/app/oracle/product/11.2.0/db_home1

Net Services Administration: Host Login

Host: edrsr25p1.us.oracle.com

Oracle Home: /u01/app/oracle/product/11.2.0/db_home1

* Username: oracle

* Password:

☐ Save as Preferred Credential

Cancel Login

Listeners: /u01/app/oracle/product/11.2.0/db_home1/network/admin

A listener process is identified by the listening end-points ('Host' and 'Port'), along with the other parameters like, logging and tracing levels, log/trace directories etc. All these parameters are defined in the "Listener Parameter File" (listener.ora). This page shows the status of a listener as "Started" only when the listener is running, and has been started using the "Listener Parameter File" at the same location as shown above.

Create Listener

Cancel OK

General Authentication Logging & Tracing Static Database Registration Other Services

* Listener Name: LISTENER0

Addresses

Listener must have at least one address. If address is changed, listener will be stopped before applying changes.

Add

Select Protocol Protocol Details

(No items found.)

ORACLE

© Oracle, 2009. Все права защищены.

Создание прослушивателя

Чтобы создать прослушиватель Oracle Net, на странице свойств прослушивателя в области «Related Links» (Связанные ссылки) выберите пункт «Net Services Administration» (Администрирование Net Services). Далее выполните следующие шаги:

1. В раскрывающемся списке «Administer» (Администрирование) выберите пункт «Listeners» (Прослушиватели) и нажмите кнопку «Go» (Перейти).
2. Если вы не ввели предпочтительные учетные данные хоста, появляется страница «Host Login» (Доступ к хосту). Введите имя пользователя и пароль, а затем щелкните «Login» (Вход).
3. Нажмите кнопку «Create» (Создать).
4. Введите имя прослушивателя. На данном сервере имя должно быть уникальным. Щелкните «Add» (Добавить), чтобы добавить адрес прослушивателя. У каждого прослушивателя должен быть как минимум один адрес.

Добавление адресов прослушивателей

The screenshot displays the Oracle Net Configuration Assistant interface. The 'Add Address' window is open, showing the 'Protocol' set to 'TCP/IP', 'Port' as '1522', and 'Host' as 'edrsr25p1.us.oracle.com'. The 'Create Listener' window is also open, showing the 'Listener Name' as 'LISTENER0'. The 'Listeners' table at the bottom shows the status of the listener as 'Stopped'. Numbered callouts indicate the sequence of steps for adding and starting a listener.

ORACLE

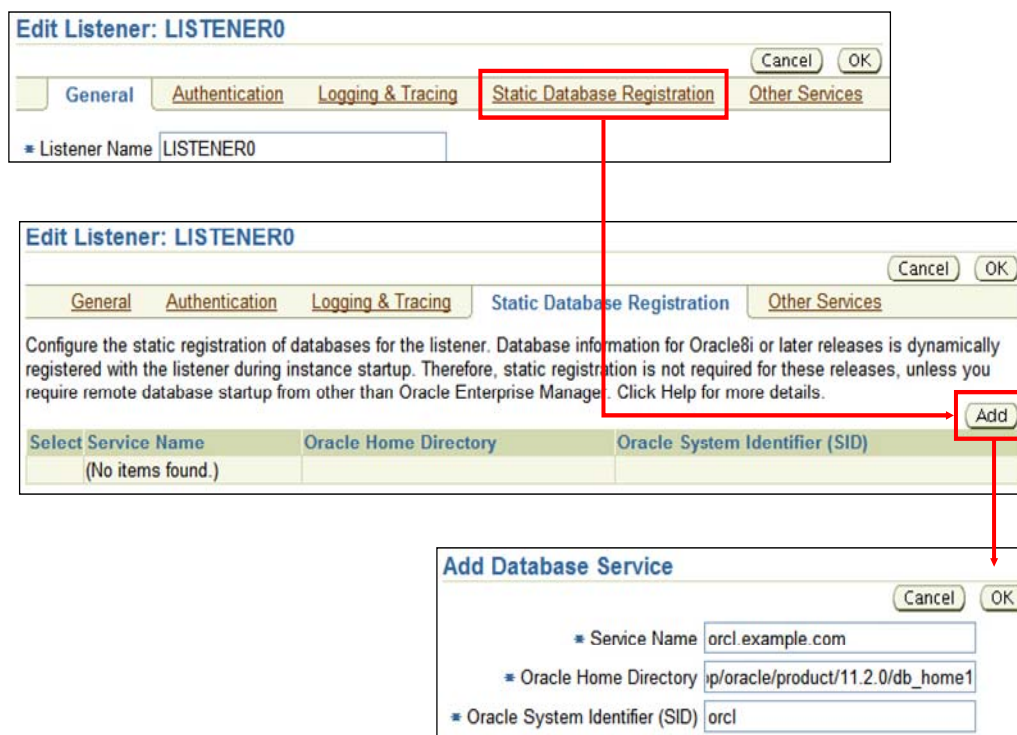
© Oracle, 2009. Все права защищены.

Добавление адресов прослушивателей

После создания прослушивателя создается адрес прослушивателя:

5. Выберите сетевой протокол. По умолчанию выбран самый распространенный из протоколов TCP/IP. Кроме него доступны протокол Internal Process Communication (IPC) — обычно используется для подключения локальных приложений (располагающихся на сервере базы данных); внешние библиотеки кода (EXTPROC) и TCP/IP с поддержкой SSL.
Примечание. Протокол EXTPROC настраивается на вкладке «Other Services» (Другие службы).
6. Укажите порт, за которым будет следить прослушиватель. По умолчанию Oracle Net использует порт 1521. Если выбрать другой порт, потребуется дополнительная настройка прослушивателя или экземпляра.
7. Введите имя или IP-адрес сервера, на котором будет запускаться прослушиватель.
8. Все остальные шаги настройки прослушивателя являются необязательными. Нажмите кнопку «OK», чтобы сохранить адрес. Единственными обязательными параметрами являются адрес прослушивания и имя.
9. На странице «Create Listener» (Создать прослушиватель) изучите информацию о только что созданном адресе и щелкните «OK», чтобы сохранить изменения.
10. Чтобы запустить новый прослушиватель, в раскрывающемся списке «Action» (Действия) выберите пункт «Start/Stop» (Запустить/остановить), после чего нажмите кнопку «Go» (Перейти).

Регистрация служб базы данных



ORACLE

© Oracle, 2009. Все права защищены.

Регистрация служб базы данных

Чтобы прослушиватель перенаправлял клиентские соединения к экземпляру, прослушивателю необходимо получить имя этого экземпляра, а также расположение его каталога `ORACLE_HOME`. Прослушиватель может найти эту информацию двумя способами:

- **Динамическая регистрация служб:** экземпляры Oraclei и более поздних версий автоматически регистрируются на прослушивателе по умолчанию сразу при запуске базы данных. Для этого прослушивателя дополнительная настройка не требуется.
- **Статическая регистрация служб:** в ранних версиях БД Oracle регистрация на прослушивателе не происходит автоматически, поэтому файл конфигурации прослушивателя должен содержать список всех баз данных, которые обслуживаются прослушивателем.

В новых версиях также можно выбрать статическую регистрацию, если:

- используется нестандартный порт прослушивателя (не 1521), а нужно, чтобы экземпляры регистрировались с портом по умолчанию;
- для используемого приложения требуется статическая регистрация служб.

Чтобы добавить статическую службу БД, на странице «Edit Listener» (Редактирование прослушивателя) выберите пункт «Static Database Registration» (Статическая регистрация базы данных), после чего нажмите кнопку «Add» (Добавить). Введите имя службы (точно так же, как глобальное имя БД `<ИМЯ_БД> . <ДОМЕН_БД>`), путь к каталогу `ORACLE_HOME` и SID (он совпадает с именем экземпляра). Нажмите кнопку "OK". Чтобы изменения вступили в силу, необходимо перезагрузить (с помощью команды `RELOAD`) или перезапустить прослушиватель.

Регистрация служб базы данных (продолжение)

Имена служб

Параметр инициализации `SERVICE_NAMES` задает одно или несколько имен, с помощью которых клиент может устанавливать соединение с базой данных. Экземпляр регистрирует имена своих служб в прослушивателе. Когда клиент запрашивает службу, прослушиватель определяет, каким экземплярам принадлежит требуемая служба, после чего направляет клиента к нужному экземпляру.

Можно указать несколько имен служб, чтобы различать используемые базы данных, например:

```
SERVICE_NAMES = sales.example.com, eurosales.example.com
```

Также с помощью имен служб можно различать одну службу, доступную в двух разных БД после репликации.

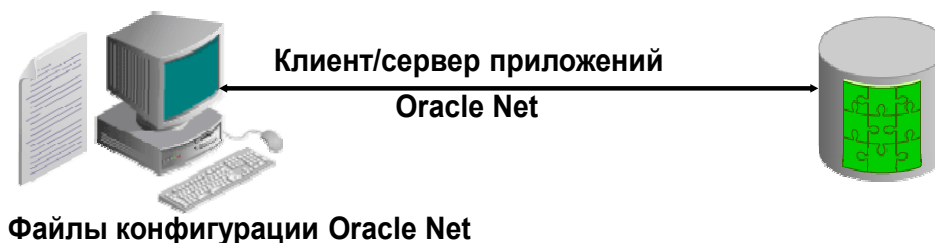
Если не указать имена службы с доменом, Oracle определит их при помощи параметра `DB_DOMAIN`. Если параметр `DB_DOMAIN` не указан, то для неопределенных значений `SERVICE_NAMES` домен также не будет определен.

При обработке клиентского соединения прослушиватель пробует сопоставить значение этого параметра и параметра `SERVICE_NAME` в дескрипторе клиентского соединения. Если в дескрипторе клиентского соединения используется параметр `SID`, прослушиватель не пытается сопоставить данные значения. Значение данного параметра обычно получается на основе параметров `DB_NAME` и `DB_DOMAIN` (`DB_NAME.DB_DOMAIN`) из файла параметров инициализации. Также параметр может содержать любое допустимое имя, с помощью которого клиент идентифицирует службу.

Методы разрешения имен

Oracle Net поддерживает несколько методов разрешения информации о соединении:

- Разрешение имен Easy Connect: используется строка соединения TCP/IP
- **Локальное разрешение имен: используется локальный файл конфигурации**
- Разрешение имен на сервере каталогов: используется центральный сервер каталогов с поддержкой LDAP
- Внешнее разрешение имен: используется сторонняя служба разрешения имен



ORACLE

© Oracle, 2009. Все права защищены.

Методы разрешения имен

Oracle Net поддерживает следующие методы разрешения имен:

- **Разрешение имен Easy Connect:** при данном методе клиенты могут соединяться с сервером БД Oracle с помощью строки соединения TCP/IP, которая состоит из имени хоста, после которого могут быть также указаны порт и имя службы:
`CONNECT имя_пользователя/пароль@хост[:порт]/имя_службы`
Данный метод разрешения имен не требует настройки.
- **Локальное разрешение имен:** при данном методе разрешения имен дескрипторы соединения (различаемые по именам сетевых служб) хранятся на стороне клиента в локальном файле конфигурации `tnsnames.ora`.
- **Разрешение имен на сервере каталогов:** для доступа к службе БД данный метод сохраняет идентификаторы соединения для обращения к службе БД на центральном сервере каталогов, который поддерживает протокол облегченного доступа к каталогам (LDAP).
- **Внешнее разрешение имен:** при данном методе имена служб хранятся в поддерживаемой сторонней службе разрешения имен. Поддерживаемые сторонние службы:
 - служба внешнего разрешения имен Network Information Service (NIS);
 - Distributed Computing Environment (DCE) Cell Directory Services (CDS)

Easy Connect

- Используется по умолчанию
- Не требует настройки на стороне клиента
- Поддерживает только протокол TCP/IP (без SSL)
- Не поддерживает такие дополнительные функции соединения:
 - настройки обработки отказов (failover) по времени соединения;
 - маршрутизация от источника;
 - балансирование нагрузки.

```
SQL> CONNECT hr/hr@db.us.oracle.com:1521/dba11g
```



Файлы конфигурации Oracle Net не используются

ORACLE

© Oracle, 2009. Все права защищены.

Easy Connect

При использовании Easy Connect вся необходимая для соединения с Oracle Net информация передается в строке соединения. Строки соединения Easy Connect имеют следующий вид:

<имя_пользователя>/<пароль>@<имя_хоста>:<порт
прослушивателя>/<имя службы>

Порт прослушивателя и имя службы являются необязательными аргументами. Если не указать порт прослушивателя, Oracle Net предполагает, что используется стандартный порт 1521. Если не указать имя службы, Oracle Net использует имя службы БД, аналогичное имени хоста в строке соединения.

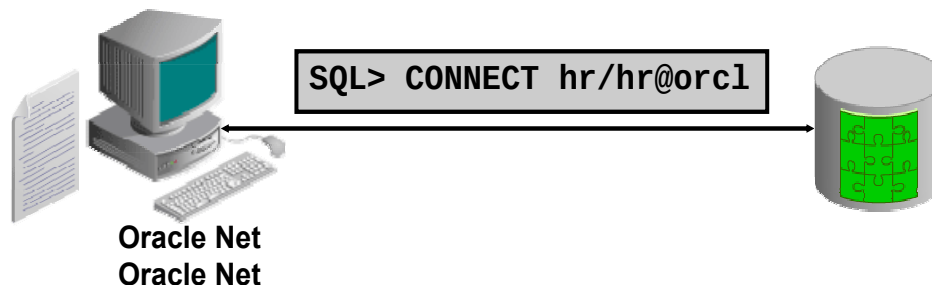
Допустим, прослушиватель использует протокол TCP для прослушивания порта 1521, а экземпляр имеет следующие значения параметров: SERVICE_NAMES=db и DB_DOMAIN=us.oracle.com. Тогда строку соединения, показанную на слайде, можно сократить:

```
SQL> connect hr/hr@db.us.oracle.com
```

Примечание. Параметр инициализации SERVICE_NAMES может принимать список значений, разделенных запятыми. Для работы этого сценария только одно из них должно быть равно db.

Локальное разрешение имен

- Требуется файл разрешения имен на стороне клиента
- Поддерживаются все протоколы Oracle Net
- Поддерживаются дополнительные параметры соединения:
 - настройки обработки отказов (failover) по времени соединения;
 - маршрутизация от источника;
 - балансирование нагрузки.



ORACLE

© Oracle, 2009. Все права защищены.

Локальное разрешение имен

При локальном разрешении имен пользователь указывает псевдоним службы Oracle Net. Oracle Net проверяет этот псевдоним в локальном списке известных служб и при обнаружении совпадения преобразовывает псевдоним в информацию о хосте, протоколе, порте и имени службы.

Одним из преимуществ локальных имен является то, что пользователям БД необходимо помнить только краткий псевдоним, а не длинную строку соединения, которая требуется для Easy Connect.

Локальный список известных служб хранится в текстовом файле конфигурации:

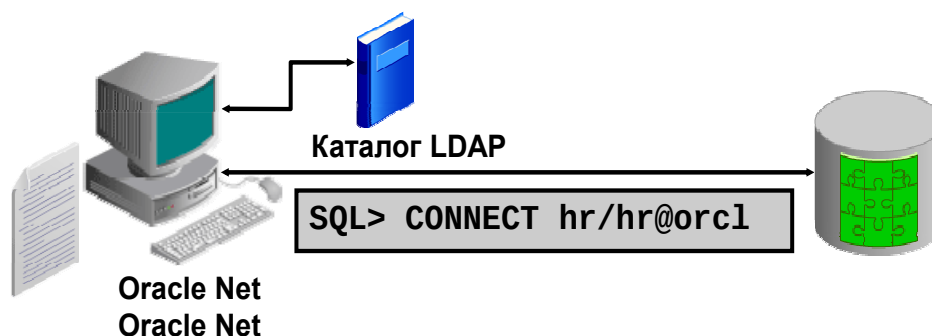
<oracle_home>/network/admin/tnsnames.ora

Это стандартный каталог для файла `tnsnames.ora`; для изменения этого каталога используется переменная `TNS_ADMIN`.

Локальное разрешение имен удобно в организациях, в которых редко меняется конфигурация служб Oracle Net.

Разрешение имен на сервере каталогов

- Требуется протокол LDAP и загрузка информации о разрешении имен Oracle Net:
 - Oracle Internet Directory;
 - службы Microsoft Active Directory.
- Поддерживаются все протоколы Oracle Net
- Поддерживаются дополнительные параметры соединения



ORACLE

© Oracle, 2009. Все права защищены.

Разрешение имен на сервере каталогов

При разрешении имен на сервере каталогов имен пользователь указывает псевдоним службы Oracle Net. Oracle Net ищет этот псевдоним во внешнем списке известных служб и при обнаружении совпадения преобразует псевдоним в информацию о хосте, протоколе, порте и имени службы. Как и при локальном разрешении имен, пользователям БД требуется помнить только краткий псевдоним.

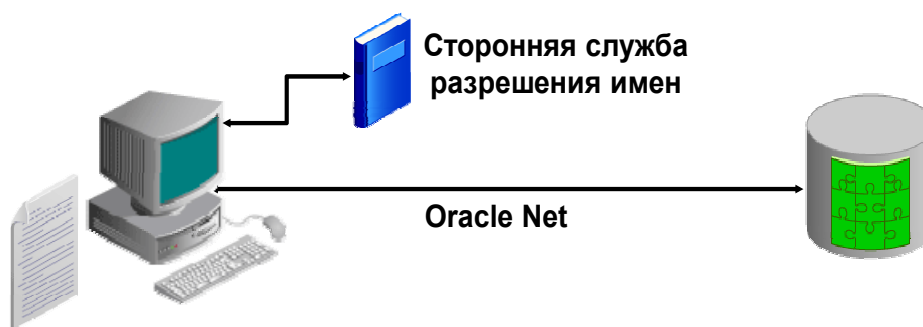
Одним из преимуществ данного метода является то, что имя службы может использоваться для соединения пользователя сразу же после добавления этой службы в каталог LDAP.

При локальном разрешении имен администратору БД необходимо сначала распространить обновленные файлы `tnsnames.ora` с новой информацией об именах служб, после чего пользователи получают возможность соединяться с новыми или измененными службами.

Разрешение имен на сервере каталогов удобно в организациях, в которых часто меняется конфигурация служб Oracle Net.

Внешнее разрешение имен

- Используется сторонняя служба разрешения имен.
- Она включает в себя:
 - служба внешнего разрешения имен Network Information Service (NIS);
 - Distributed Computing Environment (DCE) Cell Directory Services (CDS)



ORACLE

© Oracle, 2009. Все права защищены.

Внешнее разрешение имен

При данном методе имена служб Net Service хранятся в поддерживаемой сторонней службе разрешения имен. Поддерживаемые сторонние службы:

- служба внешнего разрешения имен Network Information Service (NIS);
- Distributed Computing Environment (DCE) Cell Directory Services (CDS)

Концептуально метод внешнего разрешения имен аналогичен разрешению имен на сервере каталогов.

Настройка псевдонимов служб

The screenshot shows the Oracle Net Services configuration interface. At the top, the 'Local Naming' section is active, showing the path `/u01/app/oracle/product/11.2.0/db_home1/network/admin`. Below this, the 'Create Net Service Name' dialog is open. In this dialog, the 'General' tab is selected, and the 'Net Service Name' is set to 'testorcl'. Under 'Database Information', the 'Use Service Name' radio button is selected, and the 'Service Name' is 'orcl.example.com'. The 'Add Address' dialog is also open, showing the 'TCP/IP' protocol, port '1522', and host 'edrsr25p1.us.oracle.com'. Red arrows highlight the 'Create' button in the 'Create Net Service Name' dialog, the 'Add' button in the 'Add Address' dialog, and the 'OK' button in the 'Add Address' dialog. The 'Addresses' table at the bottom is empty.

ORACLE

© Oracle, 2009. Все права защищены.

Настройка псевдонимов служб

Чтобы создать локальный псевдоним службы Oracle Net, в раскрывающемся списке «Administer» (Администрирование) выберите пункт «Local Naming» (Локальное разрешение имен) на странице администрирования Net Services и нажмите кнопку «Go» (Перейти). Затем нажмите кнопку «Create» (Создать).

Чтобы настроить псевдонимы служб для разрешения имен в сервере каталогов, выберите пункт «Directory Naming» (Разрешение имен в сервере каталогов), а не «Local Naming» (Локальное разрешение имен).

Примечание. Если разрешение имен на сервере каталогов не настроено, пункт «Directory Naming» (Разрешение имен в сервере каталогов) выбрать невозможно. Разрешение имен на сервере каталогов рассматривается в курсе *Управление идентификаторами с помощью Oracle Enterprise*, а также в руководстве *Oracle Advanced Security Administration*.

На странице создания имени сетевой службы в поле «Net Service Name» (Имя службы) введите уникальное имя. (Это имя будут вводить пользователи, когда необходимо использовать этот псевдоним.) Введите имя службы или системный идентификатор (SID) базы данных, с которой устанавливается соединение, затем нажмите кнопку «Add» (Добавить), чтобы добавить адрес для имени службы.

Для введенного адреса укажите протокол, порт и имя хоста, используемого прослушивателем.

Дополнительные параметры соединения

При использовании локального разрешения имен или разрешения имен на сервере каталогов Oracle Net поддерживает следующие дополнительные параметры соединения:

- настройки обработки отказов (failover) по времени соединения;
- маршрутизация от источника;
- балансирование нагрузки.

Addresses

Edit Remove Reorder Move To Top Go Add

Select	Protocol	Protocol Details
☒	TCP/IP	Host edrsr25p1.us.oracle.com Port 1522
☐	TCP/IP	Host edrsr25p1.us.oracle.com Port 1521

Connect-time Failover and Client Load Balancing
Configure whether addresses are tried randomly or sequentially during connections to the service. This setting is applicable only if there are more than one addresses configured.

☐ Try each address, in order, until one succeeds
☒ Try each address randomly, until one succeeds
☐ Try one address, selected at random
☐ Use each address in order until destination is reached
☐ Use only the first address

ORACLE

© Oracle, 2009. Все права защищены.

Дополнительные параметры соединения

Если к службе БД можно обратиться по нескольким адресам протоколов прослушивателя, эти адреса можно указать в порядке использования. Они могут выбираться произвольно или последовательно. Если доступно несколько прослушивателей, например в конфигурациях Oracle Real Application Clusters (RAC), Oracle Net может использовать преимущества обработки отказов прослушивателей (failover) и маршрутизации от источника Oracle Connection Manager.

Если включена функция *failover по времени соединения*, псевдоним может содержать несколько адресов прослушивателя. Если первый адрес недоступен, выбирается второй адрес. Oracle Net перебирает адреса в приведенном порядке, пока не найдет действующий прослушиватель или пока не переберет все адреса, не найдя ни одного действительного. Функция прозрачного восстановления приложений Transparent Application Failover (TAF) — это функция на стороне клиента, позволяющая клиентам возобновить соединение с функционирующими БД в случае сбоя экземпляра БД. С помощью извещений сервер запускает обратные вызовы TAF со стороны клиента.

При использовании *балансирования нагрузки* Oracle Net произвольно выбирает адрес из списка адресов. Функция балансирования нагрузки соединений во время выполнения повышает производительность за счет распределения активных соединений среди диспетчеров. В среде RAC используется балансирование нагрузки пула соединений, при котором активные соединения распределяются между несколькими экземплярами.

Дополнительные параметры соединения (продолжение)

Маршрутизация от источника обеспечивается служебной программой Oracle Connection Manager, которая выполняет роль прокси-сервера для трафика Oracle Net и обеспечивает безопасную маршрутизацию этого трафика через брандмауэр. Oracle Net обрабатывает адреса как список ретрансляторов, то есть подключается по первому адресу и запрашивает передачу с этого адреса на следующий — и так до достижения конечной точки. Это отличается от обработки отказов или балансирования нагрузки тем, что для любого соединения используются все адреса.

Параметр	Дополнительная функциональность
По очереди проверяются все адреса, пока не будет найден действительный.	Обработка отказов (Failover)
Произвольно по очереди проверяются все адреса, пока не будет найден действительный.	Обработка отказов (Failover) балансирование нагрузки.
Проверяется один произвольно выбранный адрес.	Балансирование нагрузки
По очереди используются все адреса, пока не будет достигнута точка назначения.	Маршрутизация от источника
Используется только первый адрес.	Нет

Проверка связности сети Oracle Net

Служебная программа `tnsping`, которая проверяет псевдонимы служб Oracle Net:

- обеспечивает соединение между клиентом и прослушивателем Oracle Net;
- не проверяет доступность запрошенной службы;
- поддерживает разрешение имен Easy Connect:

```
tnsping host01.example.com:1521/orcl
```

- поддерживает локальное разрешение имен и разрешение имен на сервере каталогов:

```
tnsping orcl
```

ORACLE

© Oracle, 2009. Все права защищены.

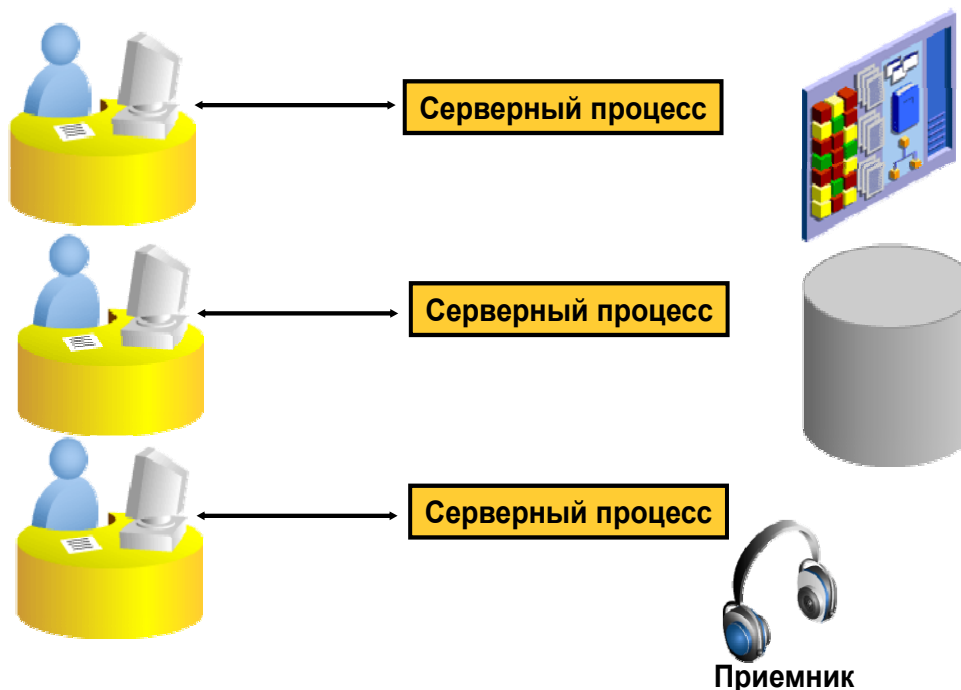
Проверка связности сети Oracle Net

`tnsping` в Oracle Net является аналогом служебной программы проверки запросов `ping` для протокола TCP/IP. С ее помощью можно провести быстрый тест скорости сетевого пути. Например, введите в окне командной строки `tnsping orcl`.

Служебная программа проверит, доступен ли прослушиватель для данного хоста, порта и протокола. При этом не проверяется, обработает ли прослушиватель имя службы. Служебная программа `tnsping` также выявляет расположение файлов конфигурации. Это удобно для системы с несколькими каталогами `ORACLE_HOME`.

Сеансы пользователей: выделенный серверный процесс

Сеансы пользователей



© Oracle, 2009. Все права защищены.

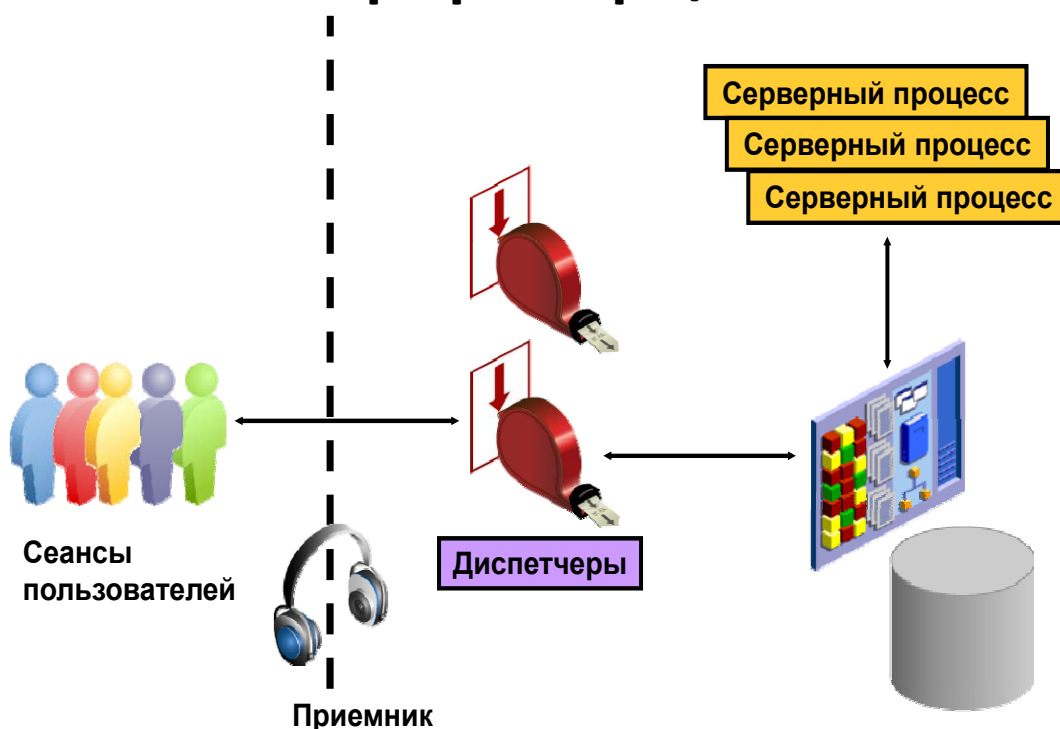
Сеансы пользователей: выделенный серверный процесс

При использовании выделенных серверных процессов соотношение пользовательских и серверных процессов равно один к одному. Каждый серверный процесс использует системные ресурсы, включая процессорное время и память.

В системах с интенсивной нагрузкой выделение ресурсов памяти и процессора выделенным серверным процессам может оказаться недопустимым и негативно отразиться на масштабируемости таких систем. Если требования к ресурсам, предъявляемые архитектурой выделенных серверных процессов, негативно влияют на систему, можно выполнить следующие действия:

- увеличить объем системных ресурсов, увеличив объем памяти и мощность ЦП;
- использовать архитектуру разделяемых серверных процессов Oracle.

Сеансы пользователей: разделяемые серверные процессы



ORACLE

© Oracle, 2009. Все права защищены.

Сеансы пользователей: разделяемые серверные процессы

Для каждой службы в архитектуре разделяемого сервера используется как минимум один процесс диспетчера (обычно больше). При поступлении запроса соединения прослушиватель не порождает выделенный серверный процесс. Вместо этого прослушиватель хранит список доступных диспетчеров для каждого имени службы, а также информацию о загрузке соединения (количество одновременных соединений) для диспетчера.

Запрос соединения перенаправляется к наименее загруженному диспетчеру, обслуживающему службу с запрошенным именем. В течение сеанса пользователь поддерживает соединение с одним и тем же диспетчером.

В отличие от выделенных серверных процессов, один диспетчер может обслуживать сотни сеансов пользователей.

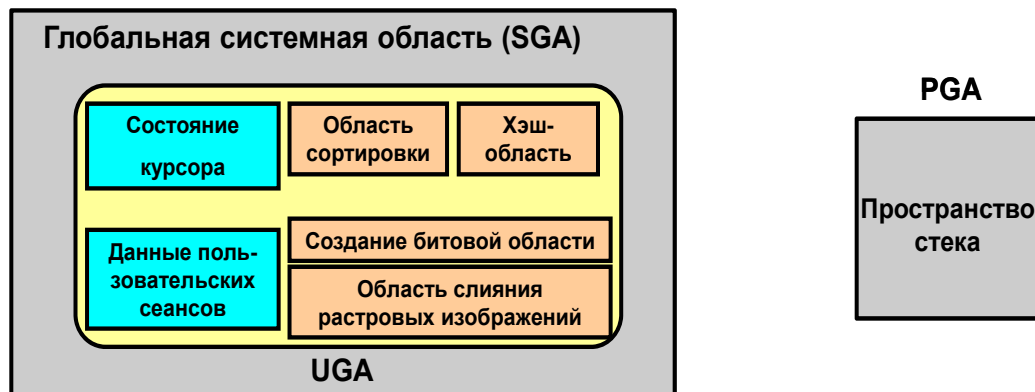
На самом деле диспетчеры не занимаются обработкой запросов пользователей. Они направляют запросы пользователей в общую очередь, которая размещена в области SGA, выделенной для разделяемого пула.

Процессы разделяемого сервера выполняют большую часть операций выделенных серверных процессов. Они выбирают запросы из очереди и обрабатывают их.

Поскольку запросы одного пользовательского процесса могут обрабатывать несколько процессов разделяемого сервера, большинство структур хранения, которые обычно находятся в PGA, следует разместить в области совместно используемой памяти (по умолчанию это разделяемый пул). Тем не менее, если настроен большой пул или параметр SGA_TARGET настроен для автоматического управления памятью, такие структуры хранения размещаются в области SGA, выделенной для большого пула.

Области SGA и PGA

Разделяемый сервер Oracle: данные пользовательских сеансов хранятся в SGA.



При выборе размера SGA необходимо учитывать требования к памяти для разделяемых серверов.

ORACLE

© Oracle, 2009. Все права защищены.

Области SGA и PGA

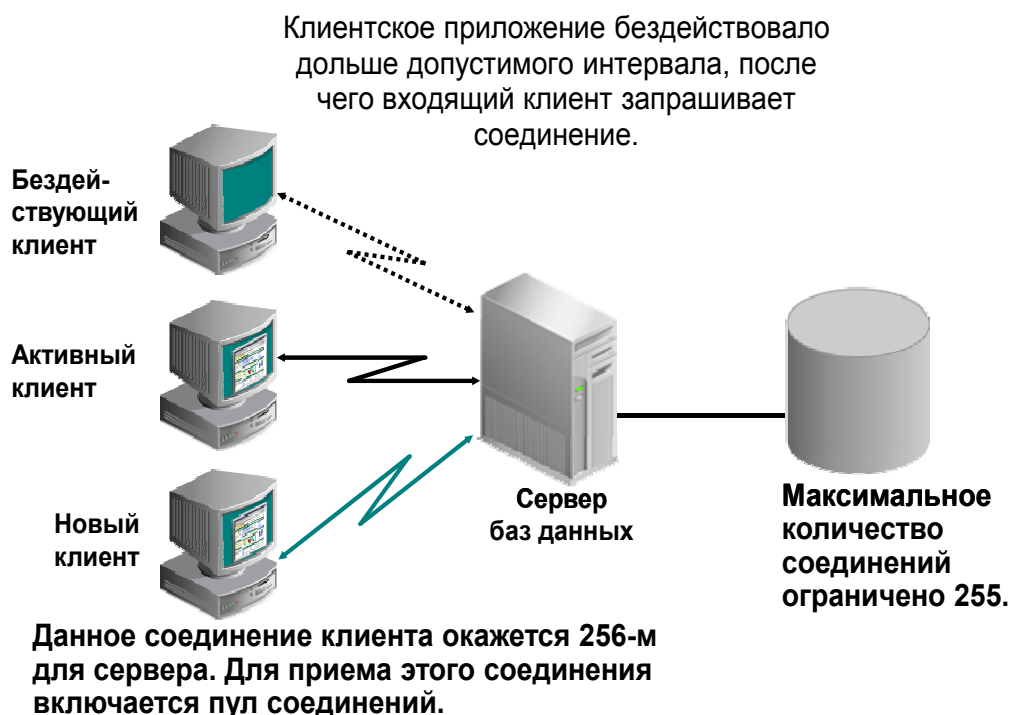
Содержимое областей SGA и PGA зависит от того, какого типа серверы используются — разделяемые или выделенные:

- текст и разобранные формы всех SQL-операторов хранятся в SGA;
- состояние курсора содержит значения из памяти во время выполнения SQL-оператора, например извлеченные строки;
- к данным пользовательского сеанса относятся сведения о безопасности и использовании ресурсов;
- область стека содержит локальные переменные процесса.

Техническое примечание

Изменения SGA и PGA прозрачны для пользователя. Однако, если поддерживается несколько пользователей, необходимо увеличить значение параметра инициализации `LARGE_POOL_SIZE`. Любой процесс разделяемого сервера должен иметь доступ к областям данных всех сеансов, тогда любой сервер сможет обработать запросы любого из сеансов. В SGA выделяется область для данных каждого сеанса. Чтобы ограничить размер области, выделяемой для сеанса, необходимо на странице «General» (Общие сведения) профиля пользователя в области «Database Services» (Службы базы данных) установить ограничение на ресурсы `PRIVATE_SGA`.

Разделяемый сервер: пул соединений



ORACLE

© Oracle, 2009. Все права защищены.

Разделяемый сервер: пул соединений

Организация пула соединений позволяет серверу БД завершать сеансы, превысившие время ожидания, и использовать освободившееся соединение для обработки активного сеанса. Бездействующий логический сеанс остается открытым, а физическое соединение автоматически восстанавливается, как только от этого сеанса поступает следующий запрос. Таким образом, Web-приложения могут одновременно обслуживать большее количество пользователей без изменения аппаратной конфигурации. Для настройки пула соединений используется разделяемый сервер.

В приведенном примере сервер БД Oracle поддерживает 255 соединений. Один из клиентов бездействовал дольше допустимого времени. При организации пула соединений это соединение может принимать входящее соединение клиента, по счету 256-е. Когда бездействующему клиенту понадобится выполнить какие-либо операции, его соединение будет восстановлено за счет соединения другого бездействующего на тот момент клиента.

Когда не стоит использовать разделяемый сервер

Определенные операции с БД не стоит выполнять с помощью разделяемых серверов:

- администрирование БД;
- **операции резервного копирования и восстановления;**
- пакетная обработка и операции с массовой загрузкой;
- операции с хранилищами данных.



Диспетчер



Выделенный
серверный процесс

ORACLE

© Oracle, 2009. Все права защищены.

Когда не стоит использовать разделяемый сервер

Архитектура разделяемых серверов Oracle позволяет эффективно использовать процессорное время и память, однако она подходит не для всех соединений. Ввиду использования общей очереди соединений и того факта, что очередь откликов диспетчера могут использовать несколько пользователей, разделяемые серверы не самым лучшим образом подходят для выполнения операций с большими наборами данных, например, для запросов хранилищ данных или пакетной обработки.

Сеансы резервного копирования и восстановления, использующие диспетчер восстановления Oracle (будет рассмотрен в следующих уроках), также оперируют большими наборами данных и требуют использования выделенных соединений.

Многие задачи администрирования не следует (или невозможно) выполнять с помощью соединений с разделяемым сервером. К ним относятся запуск и остановка экземпляров, создание табличных пространств и файлов данных, обслуживание индексов и таблиц, анализ статистики, а также многие другие типичные задачи администратора БД. Для всех сеансов администратора БД необходимо использовать выделенные серверы.

Настройка соединения между базами данных

- Передача данных или сообщений между двумя узлами требует настройки сети на обоих узлах.
- Необходимо выполнить настройку:
 - связности сети (например, TNSNAMES.ora);
 - ссылок БД;

```
CREATE DATABASE LINK <глоб_имя_удален_бд>  
CONNECT TO <пользователь> IDENTIFIED BY <пароль>  
USING '<строка_соед_для_удален_бд>';
```

ORACLE

© Oracle, 2009. Все права защищены.

Настройка соединения между базами данных

Ссылка БД является объектом схемы в БД, который позволяет получать доступ к объектам другой БД. Эта другая БД не обязательно должна быть БД Oracle. Однако для БД систем, отличных от Oracle, необходимо использовать гетерогенные службы Oracle.

Для создания частной ссылки на БД необходимо обладать системными полномочиями **CREATE DATABASE LINK**. Для создания общедоступной ссылки на БД необходимо обладать системными полномочиями **CREATE PUBLIC DATABASE LINK**. Также требуются системные полномочия **CREATE SESSION** на удаленной БД Oracle.

Когда приложение с помощью ссылки на БД обращается к удаленной базе данных, БД Oracle устанавливает соединение с базой данных на удаленной БД как при локальном запросе. Фраза **CONNECT TO**, которая используется при создании ссылки на БД, определяет, как будет устанавливаться соединение на удаленной БД. Можно создавать ссылки на БД с постоянным, текущим или подключенным пользователем. Ссылки с текущим пользователем доступны только при использовании функции дополнительной защиты Oracle. В примере на слайде показан синтаксис для создания ссылки на БД с постоянным пользователем.

Созданную ссылку на БД можно использовать для обращения к таблицам и представлениям другой базы данных. Чтобы указать в SQL-операторах, что обращение происходит к таблице или представлению другой БД, необходимо к его имени добавить **@dblink**. Можно выполнять запросы к таблицам и представлениям другой БД, а также применять к таблицам любые операторы **INSERT**, **UPDATE**, **DELETE** или **LOCK TABLE**.

Соединение с другой БД

```
REMOTE_ORCL =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)
      (HOST = host02.example.com)
      (PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = orcl.example.com)
    )
  )
)
```

tnsnames.ora

```
CONNECT hr/hr@orcl;
```

SQL*Plus

```
CREATE DATABASE LINK remote
CONNECT TO HR IDENTIFIED BY HR
USING 'REMOTE_ORCL';

SELECT * FROM employees@remote
```

ORACLE

© Oracle, 2009. Все права защищены.

Соединение с другой БД

На слайде показана запись из файла `tnsnames.ora`, необходимая для создания ссылки на БД.

В примере показана ссылка на БД с постоянным пользователем, которой присвоено имя `REMOTE`. Она подключается к пользователю `HR` с помощью строки соединения `REMOTE_ORCL`. Созданную ссылку на БД можно использовать для обращения к таблицам и представлениям другой базы данных.

Представление описывается следующим образом:

```
SQL> DESC DBA_DB_LINKS
```

Name	Null?	Type
OWNER	NOT NULL	VARCHAR2(30)
DB_LINK	NOT NULL	VARCHAR2(128)
USERNAME		VARCHAR2(30)
HOST		VARCHAR2(2000)
CREATED	NOT NULL	DATE

```
SQL> select owner, db_link, username from dba_db_links;
```

OWNER	DB_LINK	USERNAME
HR	REMOTE.EXAMPLE.COM	HR

Экспресс-тест

Какие файлы конфигурации используются для настройки прослушивателя?

1. listener.ora
2. listener.conf
3. tnsnames.ora
4. tnsnames.conf
5. sqlnet.ora
6. sqlnet.conf

ORACLE

© Oracle, 2009. Все права защищены.

Ответы: 1, 5

Экспресс-тест

При использовании архитектуры разделяемых серверных процессов PGA перемещается в SGA.

1. Да
2. Нет

ORACLE

© Oracle, 2009. Все права защищены.

Ответ: 2

Заключение

В ходе прохождения урока вы научились:

- Использовать Enterprise Manager для:
 - создания дополнительных прослушивателей
 - **создания псевдонимов служб Oracle Net Service**
 - настройки обработки отказов (failover) для времени соединения
 - управления прослушивателем Oracle Net
- Использовать `tnsping` для проверки связности сети **Oracle Net**
- Определять, когда следует использовать разделяемые серверы, а когда — выделенные серверы

ORACLE

© Oracle, 2009. Все права защищены.

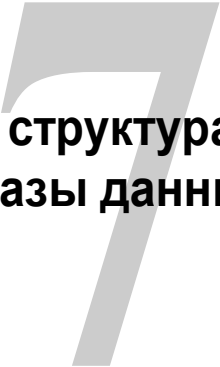
Обзор упражнения 6: работа с сетевыми компонентами Oracle

В данном упражнении рассматриваются следующие темы:

- настройка локального разрешения имен для соединения с другой БД;
- **создание второго прослушивателя для обработки отказов (failover) соединения.**

ORACLE

© Oracle, 2009. Все права защищены.



Управление структурами хранения базы данных

ORACLE®

© Oracle, 2009. Все права защищены.

Цели

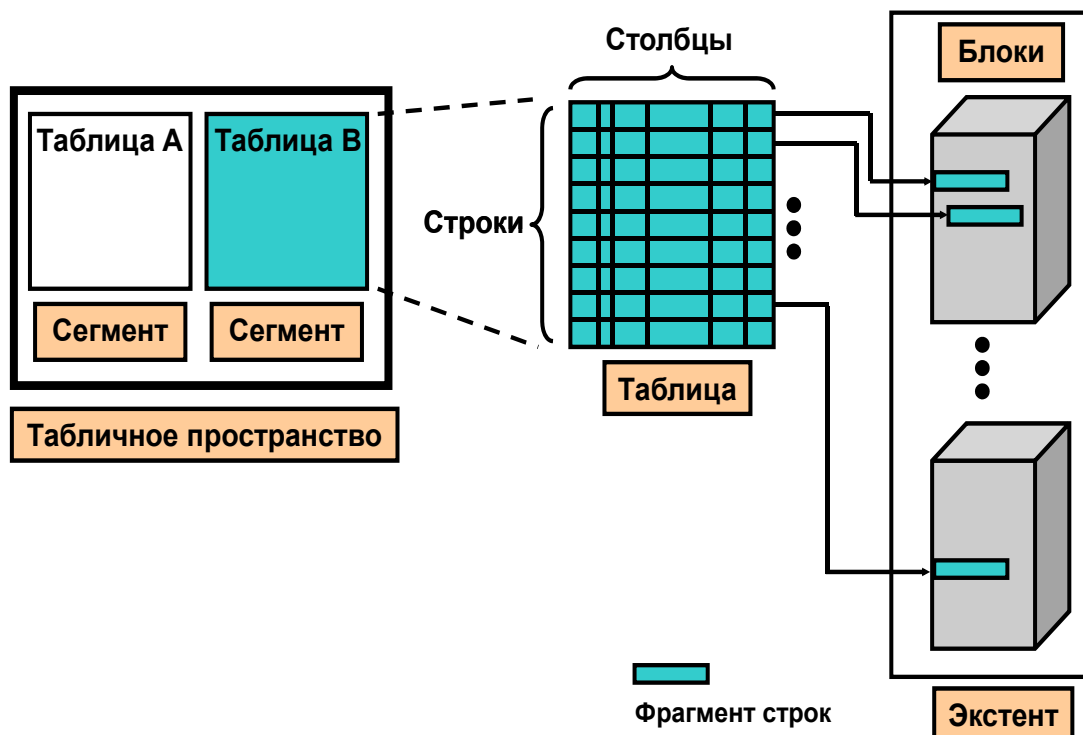
По завершении данного урока вы сможете:

- описывать хранение данных строк таблиц в блоках;
- создавать табличные пространства и управлять ими;
- получать сведения о табличных пространствах.

ORACLE

© Oracle, 2009. Все права защищены.

Как хранятся табличные данные



ORACLE

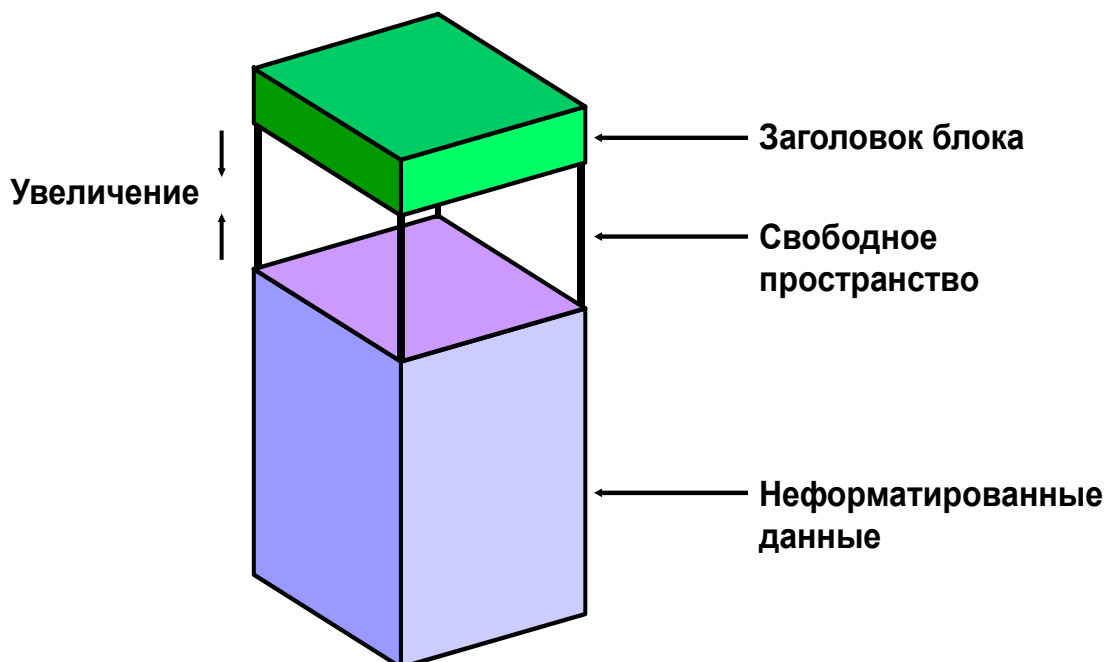
© Oracle, 2009. Все права защищены.

Как хранятся табличные данные

При создании таблицы создается сегмент для хранения ее данных. Табличное пространство содержит набор сегментов.

С логической точки зрения таблица содержит строки, заполненные значениями столбцов. Строка хранится в блоке базы данных в виде неформатированной строки. Она называется неформатированной строкой потому, что по некоторым причинам *фрагменты строки* могут храниться в разных местах. Это происходит по той причине, что вставленная строка может быть слишком большой, чтобы поместиться в одном блоке (строка, разбитая на цепочки), или же в результате операции обновления существующая строка перестает помещаться в доступное свободное пространство текущего блока (перенесенная строка). Фрагменты строки также используются, если таблица содержит более 255 столбцов. В этом случае фрагменты могут размещаться в одном и том же блоке (разбиение на цепочки между блоками) или в нескольких блоках.

Блок базы данных: содержимое



ORACLE

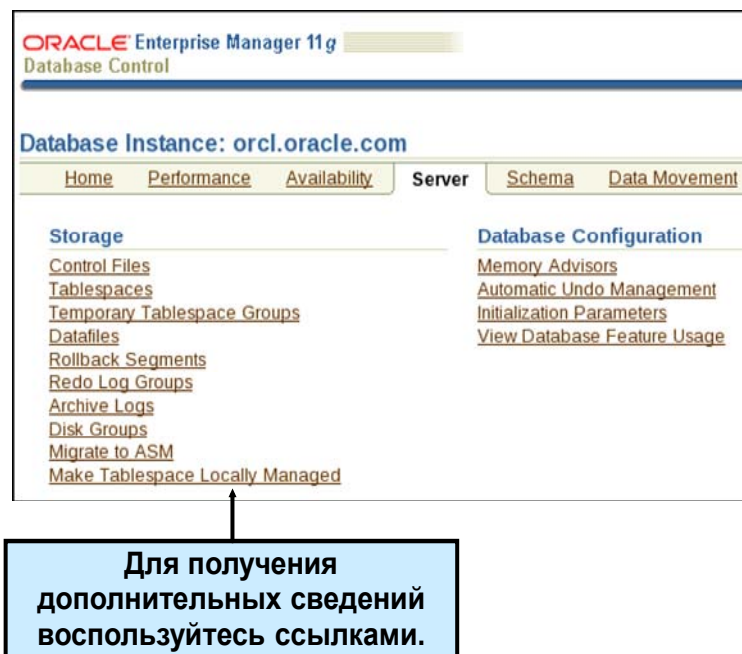
© Oracle, 2009. Все права защищены.

Блок базы данных: содержимое

- **Заголовок блока:** заголовок блока содержит сведения о типе сегмента (табличный или индексный), адресе блока данных, каталоге таблицы, каталоге строки и транзакционных слотах размером приблизительно по 23 байта каждый, которые используются при внесении изменений в строки блока. Заголовок блока заполняется сверху вниз.
- **Неформатированные данные:** это действительные данные для строк блока. Пространства неформатированных данных заполняются снизу вверх.
- **Свободное пространство:** свободное пространство внутри блока, за счет которого могут увеличиваться заголовок и пространство неформатированных данных. Свободное пространство заполняется неформатированными данными при вставке новых строк или при обновлении существующих столбцов большими значениями. События, которые могут увеличить размер заголовка:
 - для неформатированных каталогов требуется больше неформатированных записей;
 - требуется большее количество транзакционных слотов, чем было указано изначально.

Изначально свободное пространство является непрерывным. Однако в ходе выполнения операций удаления и обновления свободное пространство блока может стать фрагментированным. Свободное пространство блока при необходимости может быть объединено (дефрагментировано) сервером Oracle.

Структура хранилища



ORACLE

© Oracle, 2009. Все права защищены.

Структура хранилища

Логические структуры данных хранятся в физических файлах базы данных. Логические структуры базы данных можно легко просматривать с помощью Enterprise Manager (ЕМ). Дополнительные сведения о каждой структуре можно получить с помощью ссылок области «Storage» (Хранение) на странице сервера.

Создание нового табличного пространства

Create Tablespace

General Storage

Name: INVENTORY

Extent Management

☒ Locally Managed
☐ Dictionary Managed

Type

☒ Permanent
☐ Set as default permanent tablespace
☐ Encryption **Encryption Options**
☐ Temporary
☐ Set as default temporary tablespace
☐ Undo
Undo Retention Guarantee ☐ Yes ☒ No

Status

☒ Read Write
☐ Read Only
☐ Offline

Datafiles

☐ Use bigfile tablespace
Tablespace can have only one datafile with no practical size limit.

Add

Select Name	Directory	Size (MB)
No items found		

ORACLE

© Oracle, 2009. Все права защищены.

Создание нового табличного пространства

1. Перейдите на вкладку «Server» (Сервер), а затем выберите ссылку «Tablespaces» (Табличные пространства) под заголовком области «Storage» (Хранение).
2. Нажмите кнопку «Create» (Создать).

Примечание. Если требуется создать табличное пространство по образцу существующего, выберите существующее табличное пространство, а затем выберите команду «Create Like» (Создать подобный) из меню «Actions» (Действия). Нажмите кнопку «Go» (Перейти).

На экране отобразится страница создания табличного пространства.

3. Введите имя табличного пространства.
4. Под заголовком области «Extent Management» (Управление экстенентами) выберите пункт «Locally Managed» (Управляемый локально).
Эффективное управление экстенентами управляемых локально табличных пространств осуществляется сервером базы данных Oracle. Управление экстенентами управляемого словарем данных табличного пространства осуществляется более активно, и для этого необходим доступ к словарю данных. Функция управления словарем предоставляется только в целях обратной совместимости; Oracle не рекомендует их использовать.
5. Под заголовком области «Type» (Тип) выберите пункт «Permanent» (Постоянный).
Постоянные табличные пространства хранят постоянные объекты базы данных, созданные системой или пользователями.
6. Под заголовком области «Status» (Состояние) выберите «Read Write» (Чтение-запись).
Состояние чтения-записи означает, что пользователи могут читать и записывать табличное пространство после его создания. Такой режим выбирается по умолчанию.
7. В области «Datafiles» (Файлы данных) щелкните на элементе «Add» (Добавить), чтобы добавить файлы данных в табличное пространство.

Создание нового табличного пространства

Выберите подходящий тип хранилища

ORACLE

© Oracle, 2009. Все права защищены.

Создание нового табличного пространства (продолжение)

Табличное пространство должно состоять как минимум из одного файла. Выберите соответствующий тип хранилища в зависимости от среды. Табличные пространства типа BIGFILE используются в особенно больших базах данных, в которых ASM или другие диспетчеры логических томов поддерживают разнесенные или избыточные массивы жестких дисков (RAID) и динамически расширяемые логические тома.

8. На странице «Add Datafiles» (Добавить файлы данных) выберите требуемый тип хранилища и введите необходимую информацию. Для ASM выберите необходимую дисковую группу. Для хранилища файловой системы укажите имя файла и каталог для файла данных.
9. Введите необходимый размер файла.
10. В области «Storage» (Хранение) выберите пункт «Automatically extend datafile when full (AUTOEXTEND)» (Автоматически увеличить файл при заполнении (АВТОУВЕЛИЧЕНИЕ)), а затем укажите значение для поля «Increment» (Приращение). Это позволит файлу данных расширяться автоматически каждый раз, когда свободное пространство в нем будет заканчиваться. Размер файла ограничен физическим носителем, на котором он размещен. Для параметра «Maximum File Size» (Максимальный размер файла) оставьте значение «Unlimited» (Не ограничено) или введите максимальный размер. Нажмите кнопку «Continue» (Продолжить), чтобы вернуться на страницу создания табличного пространства.
12. Вернувшись на страницу создания табличного пространства, можно выбрать вкладку «Storage» (Хранилище), чтобы внести изменения в параметры хранилища для этого табличного пространства по мере необходимости. В большинстве случаев можно просто принять все настройки по умолчанию на странице «Storage» (Хранение). Для создания табличного пространства нажмите кнопку «OK».

Примечание. Описанные шаги показывают, как можно быстро создать табличное пространство в большинстве ситуаций. Возможно, потребуется изменить некоторые настройки в зависимости от потребностей.

Хранилище для табличных пространств

The screenshot shows the 'Storage' configuration window for an Oracle database. It includes sections for Extent Allocation, Segment Space Management, Compression Options, Enable logging, and Block information. The 'Automatic' options are selected for Extent Allocation, Segment Space Management, and Enable logging. The 'Disabled' option is selected for Compression. The Block Size is set to 8192 B.

ORACLE

© Oracle, 2009. Все права защищены.

Хранилище для табличных пространств

Выделение экстентов: экстенты в управляемом локально табличном пространстве могут выделяться одним из двух способов:

- **Автоматически:** этот способ также называют *автовыделением*. При его применении размерами экстентов табличного пространства управляет система. Этот способ управления не применим ко временным табличным пространствам.
- **Унифицировано:** табличные пространства должны использовать унифицированный размер экстентов, который вы укажете. Размером по умолчанию является 1 Мбайт. Размер всех экстентов временных табличных пространств унифицирован. Этот режим нельзя использовать для табличных пространств отмены операций.

Управление пространством сегментов: управление пространством сегментов в управляемом локально табличном пространстве можно осуществлять:

- **Автоматически:** база данных Oracle использует битовые образы для управления свободным пространством сегментов. Битовый образ описывает состояние каждого блока данных в сегменте в соответствии с объемом пространства блока, которое доступно для вставки строк. По мере изменения объема доступного пространства в блоке, его новое состояние будет отражаться в битовом образе. База данных Oracle автоматически управляет свободным пространством с помощью битовых образов. Такая форма управления пространством называется автоматическим управлением пространством сегментов (Automatic Segment Space Management или ASSM).

Хранилище для табличных пространств (продолжение)

- **Вручную:** позволяет указать, что вы хотите использовать списки свободных сегментов для управления свободным пространством в сегментах. Списки свободных сегментов представляют собой списки блоков данных, в которых имеется свободное пространство для вставки строк. Эта форма управления пространством в сегментах называется *управлением пространством сегментов вручную*, так как требуется вручную задавать и настраивать значения параметров хранения **PCTUSED**, **FREELISTS** и **FREELIST GROUPS** для созданных в табличном пространстве объектов схемы. Этот режим используется для обеспечения обратной совместимости. Рекомендуется использовать режим **ASSM**.

Параметры сжатия: сжатие сегмента данных отключено по умолчанию. Включение сжатия сегментов данных позволяет сокращать использование дискового пространства, уменьшать использование памяти в буферном кэше и ускорять выполнение запросов при выполнении операций чтения. Однако это выполняется за счет увеличения накладных расходов ЦП для загрузки данных и DML. Это особенно полезно в системах интерактивной аналитической обработки данных (OLAP), где осуществляются длительные операции только чтения, но данную возможность также можно использовать в системах интерактивной обработки транзакций (OLTP).

Более подробная информация об использовании предложений сжатия приведена в руководстве *Oracle Database Administrator's Guide* (Руководство администратора базы данных Oracle).

Протоколирование: предложение протоколирования задает значение протоколирования по умолчанию для любого сегмента, созданного в табличном пространстве. Сведения об изменении объектов табличного пространства вносятся в журнал повторов. Если протоколирование отключено, все операции прямой загрузки, выполненные с помощью SQL*Loader или команды **INSERT**, не будут записаны в журнал повторов, и поэтому в случае утери данных объекты восстановить будет невозможно. Если объект был создан при отключенном протоколировании, следует создать для него резервную копию, если может потребоваться выполнить для него операцию восстановления. Отказ от включения протоколирования может оказывать значительное влияние на возможность восстановления объектов в будущем. Используйте эту функцию с осторожностью. Для получения дополнительных сведений о протоколировании см. руководство *Oracle Database SQL Reference* (Справочное руководство по языку SQL Oracle Database).

Примечание. Если режим **FORCE LOGGING** включен для базы данных, он переопределяет настройку протоколирования табличного пространства. Базу данных можно перевести в режим **FORCE LOGGING** в момент создания базы данных или после создания базы данных с помощью команды **alter database force logging**.

Информация о блоке: в этой области отображается размер блока, который использовался при создании табличного пространства. Это значение отображается в режиме «только для чтения». Если были заданы параметры инициализации альтернативного размера блока (**DB_nK_CACHE_SIZE**), их значения также будут отображены.

Для получения дополнительных сведений о размерах блоков обратитесь к руководству *Oracle Database Administrator's Guide* (Руководство администратора базы данных Oracle).

Табличные пространства в предварительно настроенной базе данных

- SYSTEM
- SYSAUX
- TEMP
- UNDOTBS1
- USERS
- EXAMPLE (необязателен)

Tablesaces

Object Type: Tablespace

Search

Enter an object name to filter the data that is displayed in your results set.

Object Name Go

By default, the search returns all uppercase matches beginning with the string you entered. To run an exact or case-sensitive match, double quote the search string. You can use the wildcard symbol (%) in a double quoted string.

Selection Mode: Single Create

Edit View Delete Actions Add Datafile Go

Select	Name	Allocated Size(MB)	Space Used(MB)	Allocated Space Used(%)	Auto Extend	Allocated Free Space(MB)	Status	Datafiles	Type	Extent Management	Segment Management
☒	EXAMPLE	100.0	78.8	78.8	YES	21.2	✓	1	PERMANENT	LOCAL	AUTO
☐	SYSAUX	697.2	663.9	95.2	YES	33.3	✓	1	PERMANENT	LOCAL	AUTO
☐	SYSTEM	750.0	744.2	99.2	YES	5.8	✓	1	PERMANENT	LOCAL	MANUAL
☐	TEMP	27.0	0.0	0.0	YES	27.0	✓	1	TEMPORARY	LOCAL	MANUAL
☐	UNDOTBS1	100.0	16.1	16.1	YES	83.9	✓	1	UNDO	LOCAL	MANUAL
☐	USERS	5.0	4.1	82.5	YES	0.9	✓	1	PERMANENT	LOCAL	AUTO

Total Allocated Size (GB) 1.64
Total Used (GB) 1.47
Total Allocated Free Space (GB) 0.17

✓ Online ✗ Offline 🔄 Read Only

ORACLE

© Oracle, 2009. Все права защищены.

Табличные пространства в предварительно настроенной базе данных

В предварительно созданной для данного курса базе данных созданы такие табличные пространства:

- **SYSTEM:** табличное пространство SYSTEM используется сервером Oracle для управления базой данных. Оно содержит словарь данных и таблицы с данными администрирования для базы данных. Все эти объекты хранятся в схеме SYS, и доступ к ним имеет только пользователь SYS или администраторы с соответствующими полномочиями.
- **SYSAUX:** это вспомогательное табличное пространство для табличного пространства SYSTEM. Некоторые компоненты и продукты, которые использовали табличное пространство SYSTEM или собственные табличные пространства более ранних версий базы данных Oracle, теперь используют табличное пространство SYSAUX. Каждая база данных Oracle Database, начиная с версии 10g, должна содержать табличное пространство SYSAUX.

Табличные пространства в предварительно настроенной базе данных (продолжение)

В Enterprise Manager (ЕМ) вы можете просмотреть диаграмму содержимого данного табличного пространства. Для этого выберите пункт «Tablespaces» (Табличные пространства) на странице администрирования. Выберите SYSAUX и нажмите кнопку «Edit» (Редактировать). Перейдите на закладку «Occupants» (Присутствующие). После создания табличного пространства с помощью ЕМ можно будет наблюдать за расходом пространства каждым присутствующим компонентом в табличном пространстве SYSAUX. Если вы решите, что компонент занимает слишком много места в табличном пространстве SYSAUX или ожидаете, что такое произойдет, можно переместить его в другое табличное пространство, выбрав нужный компонент и нажав кнопку «Change Tablespace» (Изменить табличное пространство).

- **TEMP:** ваше временное табличное пространство, используемое при выполнении SQL-оператора, для которого требуется создать временные сегменты (например, операция большой сортировки или создание индекса). Так же как каждому пользователю назначается табличное пространство по умолчанию для хранения созданных объектов данных, ему назначается и временное табличное пространство. Рекомендуется определить для базы данных временное табличное пространство по умолчанию, которое будет назначаться практически всем новосозданным пользователям, если для них не было назначено другое временное табличное пространство. В предварительно настроенной базе данных табличное пространство TEMP определено как временное табличное пространство по умолчанию. Это означает, что если при создании формуляра пользователя ей не было назначено временное табличное пространство, база данных Oracle назначит ей временное табличное пространство по умолчанию.
- **UNDOTBS1:** это табличное пространство отмены операций, которое используется сервером базы данных для хранения информации об отмене операций. Если база данных использует режим автоматического управления отменой операций, в каждый момент времени возможно использование только одного табличного пространства отмены операций. Данное табличное пространство создается на этапе создания базы данных.
- **USERS:** данное табличное пространство используется для хранения пользовательских объектов и данных. Если при создании пользователя табличное пространство по умолчанию не указывается, табличное пространство USERS является табличным пространством по умолчанию для всех объектов, созданных данным пользователем. Для пользователей SYS и SYSTEM постоянным табличным пространством по умолчанию остается SYSTEM.
- **EXAMPLE:** это табличное пространство содержит схемы-образцы, которые можно установить при создании базы данных. Схемы-образцы используются в примерах и поддерживают большинство платформ. Документация Oracle и материалы курсов содержат примеры, в которых используются схемы-образцы.

Примечание. Чтобы упростить администрирование, рекомендуется выделить отдельное табличное пространство для индексов.

Изменение табличного пространства

Select	Name	Allocated Size(MB)	Space Used(MB)	Allocated Space Used(%)	Auto Extend	Allocated Free Space(MB)	Status	Datafiles	Type	Extent Management	Segment Management
☒	EXAMPLE	100.0	78.8	78.8	YES	21.2	✓	1	PERMANENT	LOCAL	AUTO
☐	SYSAU										

Edit Tablespace: EXAMPLE

Actions: Add Datafile Go Show SQL Revert Apply

General Storage Thresholds

Name: EXAMPLE

Bigfile tablespace: No

Extent Management

☒ Locally Managed
☐ Dictionary Managed

Type

☒ Permanent
☐ Set as default permanent tablespace
☐ Encryption Encryption Options
☐ Temporary
☐ Set as default temporary tablespace
☐ Undo

Status

☒ Read Write
☐ Read Only
☐ Offline
Offline Mode: Normal

Datafiles

Edit Remove Add

Select	Name	Directory	Size (MB)	Used (MB)
☒	example_265_688820635	+DATA/orcl/datafile/	100.00	78.81

ORACLE

© Oracle, 2009. Все права защищены.

Изменение табличного пространства

После создания табличного пространства его можно будет изменять согласно потребностям системных изменений.

Переименование: введите новое имя для табличного пространства и нажмите кнопку «Apply» (Применить).

Изменение состояния: табличное пространство может находиться в одном из трех состояний. Не все из них могут быть доступными, так как доступные состояния зависят от типа табличного пространства.

- **Чтение-запись:** табличное пространство находится в оперативном режиме и доступно как для чтения, так и для записи.
- **Только чтение:** табличное пространство находится в режиме доступа только для чтения. Если табличное пространство было переведено в данное состояние, текущие транзакции будут завершены (с фиксацией или откатом), однако другие операции языка манипулирования данными (DML) запрещены для всех объектов табличного пространства. Табличное пространство, находясь в режиме «только чтение», функционирует в оперативном режиме. Табличные пространства SYSTEM или SYSAUX нельзя перевести в режим «только чтение».

Примечание. Табличные пространства отмены и временные табличные пространства нельзя перевести в режим «только чтение».

Изменение табличного пространства (продолжение)

- **Автономное:** табличное пространство можно перевести из оперативного режима в автономный, вследствие чего эта часть базы данных станет временно недоступной для использования. Остальная часть базы данных будет оставаться открытой и доступной для пользователей. При переводе табличного пространства в автономный режим можно использовать следующие параметры:
 - **Обычное:** табличное пространство можно сделать автономным в нормальном режиме, если во всех его файлах данных отсутствуют ошибки. База данных Oracle проследит за тем, чтобы все данные были сохранены на диске, установив контрольную точку для всех файлов данных табличного пространства перед переводом их в автономный режим.
 - **Временно:** табличное пространство временно переводится в автономный режим даже при наличии ошибок в одном или нескольких файлах данных табличного пространства. База данных Oracle переводит файлы данных (которые еще в нем не находятся) в автономный режим и устанавливает при этом для них контрольную точку. Если автономные файлы отсутствуют, однако используется фраза Temporal, для перевода табличного пространства в интерактивный режим не потребуется выполнять восстановление носителя. Однако если один или несколько файлов данных табличного пространства находится в автономном режиме по причине ошибок записи, и вы при этом пытаетесь перевести табличное пространство временно в автономный режим, вам понадобится выполнить восстановление этого пространства для перевода его в оперативный режим.
 - **Немедленно:** табличное пространство можно перевести в автономный режим немедленно, без установки контрольной точки для файлов данных БД Oracle. При использовании параметра «Immediate» (Немедленно) потребуется выполнить восстановление носителей перед возвратом табличного пространства в оперативный режим. Если база данных работает в режиме NOARCHIVELOG, моментально перевести табличное пространство в автономный режим не удастся.
 - **Для восстановления:** параметр FOR RECOVER устарел и не используется. Данный синтаксис используется только для обеспечения обратной совместимости.

Примечание. Системные табличные пространства не следует переводить в автономный режим.

Изменение размера: существующее табличное пространство можно увеличить как путем добавления к нему файлов данных, так и путем изменения размера существующего файла данных.

- Чтобы добавить в табличное пространство новый файл данных, нажмите кнопку «Add» (Добавить). Затем введите сведения о файле данных на странице «Add Datafile» (Добавление файла данных).

Примечание. Добавление дополнительных файлов данных в табличные пространства BIGFILE невозможно.

- **Чтобы изменить размер существующего файла данных, выберите нужный файл в области «Datafiles»** (Файлы данных) на странице изменения табличного пространства и нажмите кнопку «Edit» (Редактировать). На странице изменения файла данных вы сможете изменить размер файла данных. Размер табличного пространства можно как увеличить, так и уменьшить. Однако размер файла данных не может быть указан меньшим, чем объем занятого пространства в файле. Если вы попытаетесь это сделать, будет отображено сообщение об ошибке:

ORA-03297: file contains used data beyond requested RESIZE value

Параметры хранения: нажмите кнопку «Storage» (Хранение), чтобы изменить параметры протоколирования для табличного пространства.

Пороговые значения: нажмите кнопку «Thresholds» (Пороговые значения), чтобы изменить порог предупреждения или критического уровня использования пространства в табличном пространстве. На выбор доступны три варианта:

- **Использование пороговых значений базы данных по умолчанию:** используются настройки по умолчанию, которые можно изменять.
- **Указать пороговые значения:** позволяет задать пороговые значения для определенного табличного пространства.
- **Отключить пороговые значения:** отключает предупреждения о достижении пороговых значений пространства в табличном пространстве.

Примечание. Использование дискового пространства проверяется по умолчанию только каждые 10 минут, поэтому для регистрации уведомления о достижении порогового значения может потребоваться несколько минут.

Действия над табличными пространствами

Selection Mode: Single

Buttons: Edit, View, Delete, Actions, Add Datafile, Go, Create

Select	Name	Allocated Size(MB)	Used(%)	Auto Extend	Allocated Free Space(MB)	Status	Datafiles	Type	Extent Management	Segment Management
☒	EXAMPLE	100.0	78.8	YES	21.2	✓	1	PERMANENT	LOCAL	AUTO
☐	SYSAUX	697.2	95.2	YES	3.3	✓	1	PERMANENT	LOCAL	AUTO
☐	SYSTEM	750.0	99.2	YES	5.8	✓	1	PERMANENT	LOCAL	MANUAL
☐	TEMP	27.0	0.0	YES	27.0	✓	1	TEMPORARY	LOCAL	MANUAL
☐	UNDOTBS1	100.0	17.1	YES	82.9	✓	1	UNDO	LOCAL	MANUAL
☐	USERS	5.0	82.5	YES	0.9	✓	1	PERMANENT	LOCAL	AUTO

Total Allocated Size (GB) 1.64
Total Used (GB) 1.47
Total Allocated Free Space (GB) 0.17

Legend: ✓ Online, ✗ Offline, 📖 Read Only

Show DDL

```
CREATE SMALLFILE TABLESPACE "EXAMPLE" DATAFILE '+DATA/orcl/datafile/example.265.688820635'  
SIZE 100M REUSE AUTOEXTEND ON NEXT 640K MAXSIZE 32767M NOLOGGING EXTENT MANAGEMENT LOCAL  
SEGMENT SPACE MANAGEMENT AUTO
```

ORACLE

© Oracle, 2009. Все права защищены.

Действия над табличными пространствами

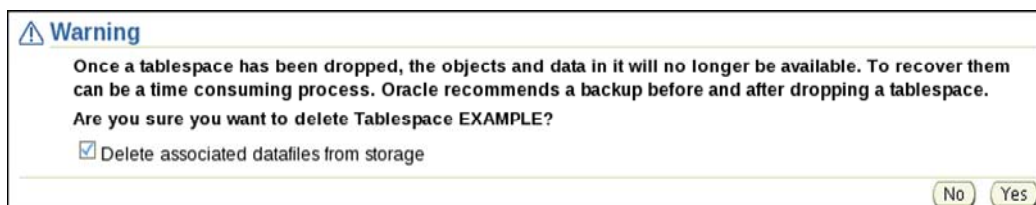
С помощью меню «Actions» (Действия) можно выполнять множество операций над табличными пространствами. Выберите табличное пространство, а затем укажите действие, которое необходимо выполнить.

- **Добавление файла данных:** добавление файла данных в табличное пространство, что позволяет его увеличить.
- **Создать подобный:** создание другого табличного пространства по образцу указанного.
- **Создать DDL:** создание оператора языка описания данных (DDL), который создает табличное пространство. В дальнейшем этот текст можно будет скопировать и вставить в текстовый файл, чтобы использовать его в документации или как сценарий.
- **Сделать управляемым локально:** преобразует табличное пространство в управляемое локально, если до этого оно управлялось словарем. Данное преобразование можно выполнить только в одну сторону. Преобразовать табличное пространство обратно в управляемое словарем будет невозможно. Для преобразования управляемого словарем табличного пространства можно воспользоваться пакетом PL/SQL DBMS_SPACE_ADMIN.TABLESPACE_MIGRATE_FROM_LOCAL.

Действия над табличными пространствами (продолжение)

- **Сделать доступным только для чтения:** блокирует все операции записи в табличное пространство. Текущие транзакции будут выполнены, однако новым DML-операторам и операциям записи будет запрещено изменять табличное пространство. Данный параметр будет доступен только в том случае, если табличное пространство не находится в режиме «только чтение».
- **Сделать доступным для записи:** разрешает DML-операторам и операциям записи изменять объекты табличного пространства. Данный параметр будет доступен только в том случае, если табличное пространство находится в режиме «только чтение».
- **Сделать интерактивным:** переводит табличное пространство из автономного режима в оперативный.
- **Реорганизовать:** запускает мастер реорганизации, который можно использовать для перемещения объектов в рамках табличного пространства для утилизации неиспользуемого пространства. Такие операции не следует выполнять, когда объекты базы данных используются активно.
- **Запустить помощник Segment Advisor:** запускает помощник Segment Advisor, который может использоваться для определения (на базе уровней фрагментации пространства в объекте) наличия в объекте пространства для утилизации. Рекомендации создаются для каждого сегмента табличного пространства.
- **Показать зависимости:** отображает объекты, от которых зависит данное табличное пространство, или объекты, которые от него зависят.
- **Показать содержимое табличного пространства:** отображает сведения обо всех сегментах табличного пространства, а также графическую карту всех экстенгов.
- **Перевести в автономный режим:** делает недоступным табличное пространство, которое в данный момент находится в оперативном режиме. Табличное пространство не удаляется, а просто становится недоступным.

Удаление табличных пространств



Select	Name	Allocated Size(MB)	Space Used(MB)	Allocated Space Used(%)	Auto Extend	Allocated Free Space(MB)	Status	Datafiles	Type	Extent Management	Segment Management
☒	EXAMPLE	100.0	78.8	78.8	YES	21.2	✓	1	PERMANENT	LOCAL	AUTO
☐	SYSAUX	697.2	663.9	95.2	YES	33.3	✓	1	PERMANENT	LOCAL	AUTO
☐	SYSTEM	750.0	744.2	99.2	YES	5.8	✓	1	PERMANENT	LOCAL	MANUAL
☐	TEMP	27.0	0.0	0.0	YES	27.0	✓	1	TEMPORARY	LOCAL	MANUAL
☐	UNDOTBS1	100.0	17.1	17.1	YES	82.9	✓	1	UNDO	LOCAL	MANUAL
☐	USERS	5.0	4.1	82.5	YES	0.9	✓	1	PERMANENT	LOCAL	AUTO

ORACLE

© Oracle, 2009. Все права защищены.

Удаление табличных пространств

Табличное пространство и его содержимое (сегменты, которые хранятся в табличном пространстве) можно удалить из базы данных, если оно и его содержимое больше не нужны. Для удаления табличного пространства необходимо обладать системным полномочием **DROP TABLESPACE**.

При удалении табличного пространства из управляющего файла соответствующей базы данных будут удалены указатели на файл. Если используются файлы OMF (Oracle-managed file), также будут удалены соответствующие файлы операционной системы. В противном случае можно указать серверу Oracle удалить файлы операционной системы (файлы данных), которые обслуживали удаленное табличное пространство. Если при удалении табличного пространства серверу Oracle не было указано удалить файлы данных, следует воспользоваться командами операционной системы, если понадобится удалить эти файлы.

Если табличное пространство содержит активные сегменты, его нельзя удалить. Например, если таблица табличного пространства используется в настоящий момент или табличное пространство содержит данные отмены операций, необходимые для отмены неподтвержденных транзакций, вы не сможете удалить табличное пространство. При удалении табличное пространство может находиться как в оперативном, так и в автономном режиме, однако рекомендуется перед удалением перевести табличное пространство в автономный режим.

Просмотр сведений о табличном пространстве

```
SELECT tablespace_name, status, contents, logging, extent_management,  
allocation_type, segment_space_management  
FROM dba_tablespaces
```

TABLESPACE_NAME	STATUS	CONTENTS	LOGGING	EXTENT_MANAGEMENT	ALLOCATION_TYPE	SEGMENT_SPACE_MANAGEMENT
SYSTEM	ONLINE	PERMANENT	LOGGING	LOCAL	SYSTEM	MANUAL
SYSAUX	ONLINE	PERMANENT	LOGGING	LOCAL	SYSTEM	AUTO
UNDOTBS1	ONLINE	UNDO	LOGGING	LOCAL	SYSTEM	MANUAL
TEMP	ONLINE	TEMPORARY	NOLOGGING	LOCAL	UNIFORM	MANUAL
USERS	ONLINE	PERMANENT	LOGGING	LOCAL	SYSTEM	AUTO
EXAMPLE	ONLINE	PERMANENT	NOLOGGING	LOCAL	SYSTEM	AUTO

```
SELECT ts#, name FROM v$tablespace
```

TS#	NAME
0	SYSTEM
1	SYSAUX
2	UNDOTBS1
4	USERS
3	TEMP
6	EXAMPLE

ORACLE

© Oracle, 2009. Все права защищены.

Просмотр сведений о табличном пространстве

Нажмите кнопку «View» (Просмотр), чтобы просмотреть сведения о выбранном табличном пространстве. На странице просмотра сведений о табличном пространстве вы можете нажать кнопку «Edit» (Редактировать), чтобы изменить табличное пространство.

Сведения о табличных пространствах и файлах данных можно просмотреть, опросив представленные ниже представления.

- **Сведения о табличных пространствах:**
 - DBA_TABLESPACES
 - V\$TABLESPACE
- **Сведения о файлах данных:**
 - DBA_DATA_FILES
 - V\$DATAFILE

Примечание. Представление V\$DBFILE отображает все файлы данных в базе данных. Данное представление сохранено для исторической совместимости. Вместо него рекомендуется использовать представление V\$DATAFILE.

- **Сведения о временных файлах:**
 - DBA_TEMP_FILES
 - V\$TEMPFILE

Просмотр содержимого табличного пространства

Show Tablespace Contents

Size (MB) 100.0 Used (MB) 78.8 Extent Mgmt LOCAL Auto Extend Yes
Block Size (KB) 8 Used (%) 78.8 Segment Mgmt AUTO Extents 882 [Return](#)

Segments

Search
Segment Name Type Minimum Size Minimum Extents [Go](#)

By default, the search returns all uppercase matches beginning with the string you entered. To run an exact or case-sensitive match, double quote the search string. You can use the wildcard symbol (%) in a double quoted string.

Segment Name	Type	Minimum Size (KB)	Minimum Extents
SH.CUSTOMERS			
PM.SYS_LOB0000			
PM.SYS_LOB0000			
SH.SUPPLEMENTA			
OE.PRODUCT_DES			
SH.SALES.SALES			
SH.SALES.SALES			
SH.SALES.SALES			
SH.CUSTOMERS_F			
SH.SALES.SALES			

Extent Map
Clicking the Highlight Extents button for a segment in the table will cause all extents that belong to that segment to be highlighted in the Extent Map. Clicking on a used extent in the Extent Map will select the segment to which that extent belongs in the segment table.

Legend: Header (pink), Used (blue), Free (green), Selected (yellow), Unmapped (black). Zoom: 75%.

[▶ Extent Map](#)

ORACLE

© Oracle, 2009. Все права защищены.

Просмотр содержимого табличного пространства

На основной странице табличных пространств или на странице определенного табличного пространства выберите «Show Tablespace Contents» (Показать содержимое табличного пространства) в раскрывающемся списке «Actions» (Действия) и щелкните «Go» (Перейти). На странице содержимого табличного пространства отображаются детализированные сведения о табличном пространстве, включая список его сегментов, тип и размер каждого сегмента, а также количество экстенгов в каждом сегменте. Таблицу можно отсортировать по любому из этих параметров, щелкнув на заголовке соответствующего столбца или же отфильтровав список путем ввода значений в области «Search» (Поиск). Для управляемого словарем табличного пространства будут отображаться дополнительные столбцы:

- Максимальное число экстенгов
- Далее
- Процент прироста

Чтобы отобразить список экстенгов, щелкните ссылку в столбце «Extents» (Экстенги).

Чтобы отобразить сведения об экстенгах в графическом виде, разверните секцию «Extent map» (Карта экстенгов) и перемещайте курсор над нужными экстенгами. Будут отображены следующие сведения:

- имя сегмента, которому принадлежит экстенг
- идентификатор экстенга
- идентификатор блока
- размер экстенга в блоках
- файл данных, в котором хранится экстенг

Файлы, управляемые Oracle (OMF)

Файловые операции следует указывать в терминах объектов базы данных, а не имен файлов.

Параметр	Описание
DB_CREATE_FILE_DEST	Определяет расположение по умолчанию системного каталога с файлами данных и временными файлами
DB_CREATE_ONLINE_LOG_DEST_n	Определяет расположение, в котором создаются файлы журнала повторов и управляющий файл
DB_RECOVERY_FILE_DEST	Расположение по умолчанию для области быстрого восстановления

Пример

```
SQL> ALTER SYSTEM SET DB_CREATE_FILE_DEST = '+DATA';  
SQL> CREATE TABLESPACE tbs_1;
```

ORACLE

© Oracle, 2009. Все права защищены.

Файлы, управляемые Oracle (OMF)

Файлы, управляемые Oracle, устраняют необходимость в прямом управлении файлами операционной системы в базе данных Oracle. Операции указываются в терминах объектов базы данных, а не имен файлов. Внутри базы данных используются стандартные интерфейсы файловой системы для создания и удаления файлов следующих структур базы данных:

- табличные пространства
- файлы журнала повторов
- управляющие файлы
- архивные журналы
- файлы отслеживания изменений в блоках
- журналы Flashback
- резервные копии RMAN

База данных может содержать как управляемые Oracle файлы, так и неуправляемые. Каталог файловой системы, указанный с помощью данных параметров, должен существовать. База данных автоматически его не создаст. Также каталог должен иметь все необходимые разрешения, чтобы база данных могла создавать в нем файлы.

Файлы, управляемые Oracle (OMF) (продолжение)

В примере иллюстрируется, что после настройки параметра DB_CREATE_FILE_DEST оператор DATAFILE может быть опущен в конструкции CREATE TABLESPACE. Файл данных будет создан в расположении, указанном параметром DB_CREATE_FILE_DEST. В примере при создании табличного пространства всем параметрам присваиваются значения по умолчанию.

Файлы, управляемые Oracle, обладают определенным форматом именования. Например, в системах на основе Linux и Unix используется следующий формат:

```
<destination_prefix>/o1_mf_%t_%u_.dbf
```

Не переименовывайте файл, управляемый Oracle. База данных идентифицирует файл, **управляемый Oracle, на основе имени. Если вы переименовываете файл, база данных не** может больше распознавать файл как управляемый Oracle и не осуществляет соответствующее управление этим файлом.

В следующем примере задается местоположение по умолчанию для созданных файлов данных (/u01/oradata), а затем создается табличное пространство tbs_1 с файлом данных в этом местоположении.

```
SQL> ALTER SYSTEM SET DB_CREATE_FILE_DEST = '/u01/oradata';  
SQL> CREATE TABLESPACE tbs_1;
```

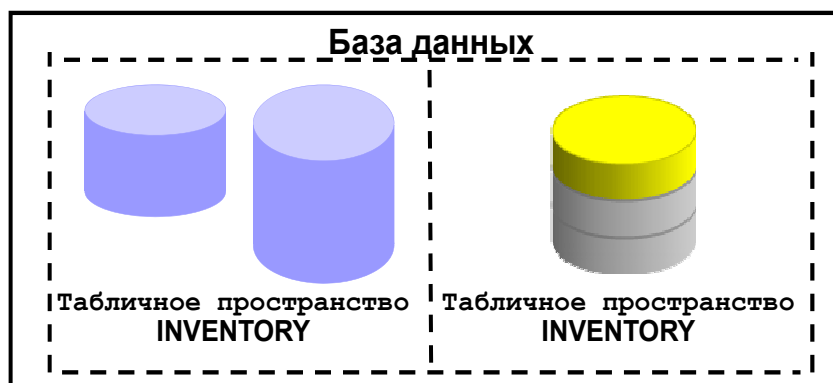
По умолчанию файлы данных, управляемые Oracle, включая файлы для табличных пространств SYSTEM и SYSAUX, имеют размер 100 Мбайт и могут увеличиваться в размере.

Примечание. По умолчанию ASM использует файлы OMF, но можно указать псевдоним для файлов данных ASM в момент создания табличного пространства или при добавлении файла данных ASM в существующее табличное пространство; в этом случае данный файл не будет являться файлом OMF.

Увеличение базы данных

Базу данных можно увеличить следующими способами:

- создать новое табличное пространство;
- добавить файл данных в существующее табличное пространство;
- увеличить размер файла данных;
- разрешить динамический рост файла данных.



ORACLE

© Oracle, 2009. Все права защищены.

Увеличение базы данных

Данные операции можно выполнять как с помощью Enterprise Manager, так и с помощью операторов SQL. Размер базы данных можно описать как сумму всех его табличных пространств.

Экспресс-тест

База данных может содержать как управляемые Oracle файлы, так и неуправляемые.

1. Да
2. Нет

ORACLE

© Oracle, 2009. Все права защищены.

Ответ: 1

Экспресс-тест

Табличные пространства типа BIGFILE должны содержать 1 файл размером по крайней мере 100 Мбайт.

1. Да
2. Нет

ORACLE

© Oracle, 2009. Все права защищены.

Ответ: 2

Заключение

В ходе прохождения урока вы научились:

- описывать хранение данных строк таблиц в блоках;
- создавать табличные пространства и управлять ими;
- получать сведения о табличных пространствах.

ORACLE

© Oracle, 2009. Все права защищены.

Обзор упражнения 7: управление структурами хранения базы данных

В данном упражнении рассматриваются следующие темы:

- создание табличных пространств;
- сбор сведений о табличных пространствах.

ORACLE

© Oracle, 2009. Все права защищены.

8

Администрирование защиты пользователей

ORACLE®

© Oracle, 2009. Все права защищены.

Цели

По завершении данного урока вы сможете:

- создавать и управлять формулярами пользователей БД для:
 - аутентификации пользователей
 - присвоения областей хранения по умолчанию (табличных пространств)
- предоставлять и отменять полномочия
- создавать роли и управлять ими
- создавать профили и управлять ими:
 - использовать возможности стандартной защиты паролем
 - контролировать использование ресурсов пользователями

ORACLE

© Oracle, 2009. Все права защищены.

Цели

Для понимания задач данного урока будут рассмотрены следующие понятия, связанные с администрированием пользователей БД:

- *Формуляр пользователя БД* — это способ организации принадлежности и доступа к объектам БД.
- *Пароль* необходим для аутентификации в БД Oracle.
- *Полномочие* — это право на выполнение определенного типа SQL-оператора или на доступ к объекту пользователя.
- *Роль* — это именованная группа связанных полномочий, которая предоставляется пользователям или другим ролям.
- *Профили* представляют собой именованные наборы ограничений на использование ресурсов БД и экземпляра и служат для управления состоянием формуляра и правилами управления паролем.
- *Квота* — это допустимый объем пространства в заданном табличном пространстве. Это один из способов управлять использованием ресурсов пользователями.

Формуляры пользователей БД

У каждого формуляра пользователя БД есть:

- уникальное имя пользователя
- метод аутентификации
- табличное пространство по умолчанию
- временное табличное пространство
- профиль пользователя
- исходная группа потребителей
- **состояние формуляра**

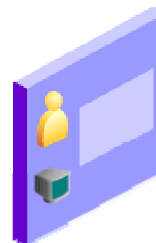


Схема:

- Это набор объектов БД, принадлежащих пользователю БД.
- Имеет то же имя, что и формуляр пользователя.

ORACLE

© Oracle, 2009. Все права защищены.

Формуляры пользователей БД

Для доступа к БД пользователю необходимо указать соответствующий формуляр пользователя БД и успешно пройти аутентификацию для данного формуляра. У каждого пользователя БД есть уникальный формуляр БД.

Oracle рекомендует руководствоваться этим правилом, чтобы избежать потенциальных брешей в безопасности и предоставить подробные данные для определенных операций аудита. Тем не менее, иногда пользователи могут иметь общий формуляр БД. В этих редких случаях безопасность БД необходимо обеспечить средствами ОС и приложений. У каждого формуляра пользователя есть:

- **Уникальное имя пользователя:** размер имени пользователя не может превышать 30 байт, имя не может содержать специальные символы и должно начинаться с буквы.
- **Метод аутентификации:** наиболее распространенным методом аутентификации является использование пароля, однако в Oracle Database 11g поддерживаются пароли, глобальная аутентификация и методы внешней аутентификации (включая биометрическую аутентификацию, аутентификацию с использованием сертификатов или маркеров).

Формуляры пользователей БД (продолжение)

- **Табличное пространство по умолчанию:** в этой области пользователь создает объекты, если не было указано другое табличное пространство. Обратите внимание, что определение табличного пространства по умолчанию вовсе не подразумевает, что у пользователя есть *полномочия* на создание объектов в этом пространстве или же что для пользователя задана *квота* на пространство для создания объектов в табличном пространстве по умолчанию. И то, и другое предоставляется отдельно.
- **Временное табличное пространство:** это область для хранения временных объектов, например сортировок и временных таблиц, которые создаются экземпляром от имени пользователя. Для временных табличных пространств квота не задается.
- **Профиль пользователя:** набор ресурсов и ограничений с помощью паролей, присвоенных пользователю.
- **Исходная группа потребителей:** используется диспетчером ресурсов.
- **Состояние формуляра:** пользователям доступны только «открытые» формуляры. Параметр `account_status` может иметь различные значения, сочетающие «locked» (заблокированный) и «expired» (просроченный).

Схемы: *Схема* — это набор объектов БД, принадлежащих пользователю БД. Объекты схемы представляют собой логические структуры, которые напрямую ссылаются на данные БД. Объектами схемы являются такие структуры, как таблицы, представления, последовательности, хранимые процедуры, синонимы, индексы, кластеры и каналы связи с базой данных. В общем случае объектами схемы являются все объекты, созданные приложением в базе данных.

Примечание. Пользователем БД не обязательно является физическое лицо. Довольно часто создается пользователь, являющийся владельцем объектов БД определенного приложения, например HR. Пользователем БД может выступать устройство или приложение. Также пользователь может создаваться для группирования объектов БД в целях безопасности. Для пользователя БД не требуется личная идентификационная информация.

Предопределенные формуляры администратора

- Формуляр SYS:
 - получает роль администратора БД, а также несколько других ролей.
 - обладает всеми полномочиями с параметром ADMIN OPTION
 - **необходим для запуска, остановки и выполнения некоторых служебных команд**
 - является владельцем словаря данных и репозитория автоматической рабочей нагрузки (AWR)
- Формуляру SYSTEM предоставляются роли DBA, MGMT_USER и AQ_ADMINISTRATOR_ROLE.
- Формуляру DBSNMP предоставляется роль OEM_MONITOR.
- Формуляру SYSMAN предоставляются роли MGMT_USER, RESOURCE и SELECT_CATALOG_ROLE.
- Эти формуляры не используются для стандартных операций.

ORACLE

© Oracle, 2009. Все права защищены.

Предопределенные формуляры администратора

Формулярам SYS и SYSTEM по умолчанию предоставляется роль администратора БД. Кроме того, формуляр SYS обладает всеми полномочиями с параметром ADMIN OPTION и является владельцем словаря данных. Для подключения к формуляру SYS необходимо использовать фразу AS SYSDBA для экземпляра БД и фразу AS SYSASM для экземпляра автоматического управления хранением (ASM). Пользователь с полномочиями SYSDBA может подключиться к формуляру SYS, воспользовавшись фразой AS SYSDBA. Только «уполномоченным» пользователям с полномочиями SYSDBA, SYSOPER или SYSASM разрешается запускать и останавливать экземпляры. Формуляр SYSTEM не обладает полномочием SYSDBA. Формуляру SYSTEM также предоставляются роли AQ_ADMINISTRATOR_ROLE и MGMT_USER. Формуляры SYS и SYSTEM являются **обязательными для БД. Они не могут быть удалены.**

Формуляр DBSNMP используется агентом управления Enterprise Manager для мониторинга базы данных и управления ей. Формуляр SYSMAN используется для выполнения задач администрирования Oracle Enterprise Manager. Ни формуляр DBSNMP, ни формуляр SYSMAN не обладает полномочием SYSDBA.

Практические рекомендации: согласно принципу меньших полномочий, эти формуляры не используются для стандартных операций. У пользователей, которым необходимы полномочия администратора БД, есть отдельные формуляры с необходимыми полномочиями. К примеру, Джим использует формуляр с незначительными полномочиями jim, а также формуляр с большими полномочиями jim_dba. Этот метод воплощает принцип меньших полномочий, устраняет необходимость в общих формулярах, а также позволяет проводить аудит отдельных действий.

Создание пользователя

Database Instance: orcl.oracle.com > Users > Logged in As SYS

Create User

General Roles System Privileges Object Privileges Quotas Consumer Group Privileges Proxy Users

* Name mydba

Profile DEFAULT

Authentication Password

* Enter Password *****

* Confirm Password *****

For Password choice, the role is authorized via password.

☐ Expire Password now

Default Tablespace USERS

Temporary Tablespace TEMP

Status ☐ Locked ☒ Unlocked

Show SQL Cancel OK

Show SQL

Return

```
CREATE USER "MYDBA" PROFILE "DEFAULT" IDENTIFIED BY "*****" DEFAULT
TABLESPACE "USERS" TEMPORARY TABLESPACE "TEMP" ACCOUNT UNLOCK
GRANT "CONNECT" TO "MYDBA"
```

Выберите последовательно пункты «Server» (Сервер) > «Users» (Пользователи), а затем нажмите кнопку «Create» (Создать).

ORACLE

© Oracle, 2009. Все права защищены.

Создание пользователя

На странице «Users» (Пользователи) утилиты Enterprise Manager можно управлять пользователями БД, которым разрешен доступ к текущей БД. С помощью этой страницы можно создать, удалить или изменить настройки пользователя.

Чтобы создать пользователя БД, выполните следующие действия:

1. В утилите Enterprise Manager Database Control щелкните вкладку «Server» (Сервер), затем в разделе «Security» (Безопасность) выберите пункт «Users» (Пользователи).
2. Нажмите кнопку «Create» (Создать).

Введите необходимую информацию. Обязательные пункты, например «Name» (Имя), помечены символом «звездочка» (*). Указанное имя должно соответствовать тем же правилам, которые используются для создания объектов базы данных. На следующих страницах этого урока будет подробнее описана процедура аутентификации. Профили будут рассмотрены позже в этом уроке.

Назначьте табличное пространство по умолчанию, а также временное табличное пространство для каждого из пользователей. Если при создании объекта пользователь не укажет табличное пространство, объект будет создан в табличном пространстве, заданном по умолчанию для данного пользователя. Это позволит контролировать размещение объектов каждого из пользователей. Если не указывать табличное пространство по умолчанию, будет использоваться системное постоянное табличное пространство. Аналогичным является подход и для временных табличных пространств. Если не указать табличное пространство, будет использоваться системное временное табличное пространство.

Примечание. Нажмите кнопку «Show SQL» (Показать SQL), чтобы просмотреть поддерживаемый синтаксис SQL. Полный синтаксис SQL для создания пользователей см. в руководстве *Oracle® Database SQL Language Reference* (Справочное руководство по языку SQL Oracle® Database).

Аутентификация пользователей

- Пароль
- Внешняя
- Глобальная

The screenshot shows the 'Edit User: HR' window in Oracle Database 11g. The 'Authentication' dropdown menu is open, showing 'Password', 'External', and 'Global' options. The 'Password' option is selected. The interface includes tabs for 'General', 'Roles', 'System Privileges', 'Object Privileges', 'Quotas', 'Consumer Group Privileges', and 'Proxy Users'. It also shows fields for 'Name' (HR), 'Profile' (DEFAULT), 'Default Tablespace' (USERS), 'Temporary Tablespace' (TEMP), and 'Status' (Locked/Unlocked). A blue 3D figure of a person in a uniform is visible on the right side of the window.

ORACLE

© Oracle, 2009. Все права защищены.

Аутентификация пользователей

Аутентификация означает проверку подлинности кого-либо или чего-либо (пользователя, устройства или другого объекта) при обращении к данным, ресурсам или приложениям. При этом проверяется подлинность установленного отношения доверия для дальнейшего взаимодействия. Использование аутентификации также позволяет вести отчетность, так как делает возможным связать доступ и действия с определенными идентификаторами. После аутентификации выполняется авторизация. Во время авторизации определяется уровень доступа и операций, разрешенных данному объекту.

При создании пользователя выбирается метод аутентификации, который в дальнейшем можно будет изменить.

Пароль: представляет собой один из способов аутентификации в БД Oracle. Каждый пользователь создается с собственным паролем, который необходимо ввести при установлении соединения. При настройке пароля можно сразу же сделать его срок действия истекшим. Тогда пользователь будет вынужден изменить пароль при первом же входе в систему. При этом следует убедиться, что у пользователей имеется возможность менять свой пароль. В некоторых приложениях эта функция отсутствует. Все пароли, создаваемые в Oracle Database 11g, по умолчанию зависят от регистра. Также эти пароли могут содержать многобайтовые символы, при этом размер пароля ограничен 30 байтами. Любой пароль, созданный в базе данных, обновленной до версии Oracle Database 11g, не зависит от регистра до первого изменения.

Аутентификация пользователей (продолжение)

При сетевом соединении (клиент-сервер или сервер-сервер) пароли всегда автоматически и прозрачно шифруются при помощи алгоритма улучшенного стандарта шифрования (AES), и только после этого рассылаются по сети.

Внешняя: данная аутентификация выполняется за пределами БД (средствами ОС, Kerberos или Radius). Для использования Kerberos или Radius необходима функция дополнительной защиты. Пользователи могут подключаться к базе данных Oracle без указания имени пользователя и пароля. Функция дополнительной защиты (представляющая собой строгую аутентификацию) позволяет идентифицировать пользователей с помощью биометрических устройств, сертификатов x509 или маркерных устройств. При использовании внешней аутентификации ограничение доступа к формулярам базы данных зависит от базовой операционной системы, а также сетевой или внешней службы аутентификации. При таком входе в систему пароль для БД не используется. Если ОС или сетевая служба позволяет, его можно использовать для аутентификации пользователей. Если используется аутентификация средствами ОС, настройте параметр инициализации `OS_AUTHENT_PREFIX` и используйте его значение как префикс для имен пользователей Oracle. Параметр `OS_AUTHENT_PREFIX` определяет префикс, который БД Oracle добавляет к именам формуляров операционной системы каждого из пользователей. По умолчанию параметр имеет значение `OPS$` в целях обратной совместимости с предыдущими версиями ПО Oracle. При запросе пользователя на подключение база данных Oracle сравнивает имя пользователя с префиксом и имена пользователей Oracle в базе данных. Например, предположим, что параметр `OS_AUTHENT_PREFIX` задан таким образом:

`OS_AUTHENT_PREFIX=OPS$`

Если пользователю с формуляром ОС `tsmith` необходимо подключиться к БД Oracle и пройти аутентификацию ОС, БД Oracle проверит, существует ли соответствующий пользователь базы данных `OPS$tsmith`. Если он существует, подключение будет разрешено. Любые ссылки на пользователя, который проходит аутентификацию средствами ОС, должны содержать префикс, как в примере — `OPS$tsmith`.

Примечание. Текстовое значение параметра инициализации `OS_AUTHENT_PREFIX` для некоторых ОС зависит от регистра. Чтобы получить более подробную информацию о данном параметре инициализации, обратитесь к документации Oracle для используемой операционной системы.

Глобальный: при использовании функции дополнительной защиты Oracle и глобальной аутентификации можно идентифицировать пользователей с помощью Oracle Internet Directory.

Подробнее дополнительные методы аутентификации рассматриваются в курсе *Безопасность БД Oracle*.

Аутентификация администраторов

Безопасность ОС.

- администратор должен владеть полномочиями ОС для создания и удаления файлов;
- обычные пользователи БД не должны обладать полномочиями ОС для создания или удаления файлов базы данных.

Безопасность администратора.

- Для подключений SYSDBA, SYSOPER и SYSASM:
 - имя пользователя DBA контролируется с помощью файла паролей и строгих методов аутентификации;
 - имя формуляра ОС контролируется с помощью аутентификации ОС;
 - для уполномоченных пользователей аутентификация средствами ОС является более приоритетной, чем аутентификация с помощью файла паролей;
 - в файлах паролей используются пароли, чувствительные к регистру.

ORACLE

© Oracle, 2009. Все права защищены.

Аутентификация администраторов

Безопасность ОС: в ОС UNIX и Linux администратор БД по умолчанию входит в группу операционной системы `oinstall`, которой предоставлены полномочия на создание и удаление файлов БД.

Безопасность администратора: подключения пользователей с полномочиями SYSDBA, SYSOPER и SYSASM проходят авторизацию только после проверки файла паролей или же полномочий и разрешений уровня операционной системы. Если используется аутентификация средствами ОС, для доступа к базе данных *не* используется имя пользователя и пароль. Такой метод аутентификации используется при отсутствии файла паролей, если в этом файле отсутствуют введенные имя пользователя и пароль или если имя и пароль не были введены. Пароли, созданные в файле паролей Oracle Database 11g, по умолчанию зависят от регистра.

Тем не менее, если аутентификация пройдена с помощью файла паролей, для соединения регистрируется имя пользователя. Если аутентификация пройдена средствами ОС, устанавливается соединение `CONNECT /`, для которого имя пользователя не сохраняется.

Примечание. В особенности если пользователь является участником группы OSDBA или OSOPER операционной системы и устанавливает соединение от имени SYSDBA или SYSOPER, то соединение будет установлено с применением соответствующих административных полномочий вне зависимости от указанного имени пользователя и пароля. Для SYSASM необходимо указать какое-либо имя пользователя или пароль (например, `sqlplus / as SYSASM`).

В Oracle Database 11g уполномоченные пользователи могут использовать методы строгой аутентификации: Kerberos, SSL или аутентификацию с помощью каталога при наличии лицензии на функцию дополнительной защиты.

Разблокирование формуляра пользователя и сброс пароля

The screenshot shows the Oracle 'Users' page. At the top, there is a search bar with the text 'Enter an object name to filter the data that is displayed in your results set.' Below it, a 'Go' button is highlighted with a red box. The 'Selection Mode' is set to 'Single'. The 'Actions' menu is open, showing options like 'Create Like', 'Expire Password', 'Generate DDL', 'Lock User', and 'Unlock User'. The 'Unlock User' option is highlighted with a red box. A red arrow points from the 'Unlock User' option to the 'APEX_PUBLIC_USER' row in the table below. The table has columns: 'Select', 'UserName', 'Account Status', 'Expiration Date', 'Default Tablespace', 'Temporary Tablespace', 'Profile', and 'Created'. The 'APEX_PUBLIC_USER' row shows 'EXPIRED & LOCKED' status and 'Aug 4, 2008 7:10:51 PM MDT' expiration date. A yellow box with a red border contains the text: 'Выберите пользователя и выберите «Unlock User» (Разблокировать пользователя), а затем щелкните «Go» (Перейти)'. A yellow padlock icon is also present.

Выберите пользователя и выберите «Unlock User» (Разблокировать пользователя), а затем щелкните «Go» (Перейти).

ORACLE

© Oracle, 2009. Все права защищены.

Разблокирование формуляра пользователя и сброс пароля

Во время установки ПО и создания базы данных можно разблокировать и сбросить настройки многих стандартных формуляров пользователей Oracle. Если не выполнить разблокирование на этом этапе, его можно будет осуществить на странице «Users» (Пользователи), выбрав кнопку «Unlock User» (Разблокировать пользователя) в списке «Actions» (Действия) и щелкнув «Go» (Перейти). Это не приведет к какому-либо изменению пароля. Если на момент разблокирования пользователя срок действия пароля истек, он остается просроченным до тех пор, пока не будет выполнено редактирование пользователя и изменение пароля.

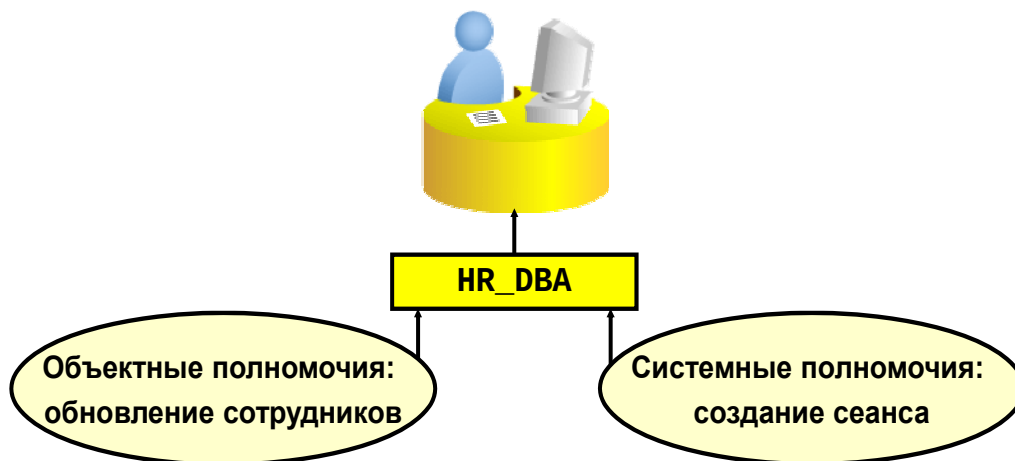
Чтобы разблокировать пользователя и сбросить пароль, выполните следующие действия на странице «Edit Users» (Редактирование пользователей):

1. Введите новый пароль в полях «Enter Password» (Введите пароль) и «Confirm Password» (Подтвердите пароль).
2. Установите флажок «Unlocked» (Разблокировано).
3. Чтобы сбросить настройки пароля и разблокировать формуляр пользователя, нажмите кнопку «Apply» (Применить).

Полномочия

Существует два типа полномочий пользователя:

- Система: позволяют пользователям выполнять определенные действия с базой данных
- Объект: позволяют пользователям получать доступ к определенным объектам и манипулировать ими



ORACLE

© Oracle, 2009. Все права защищены.

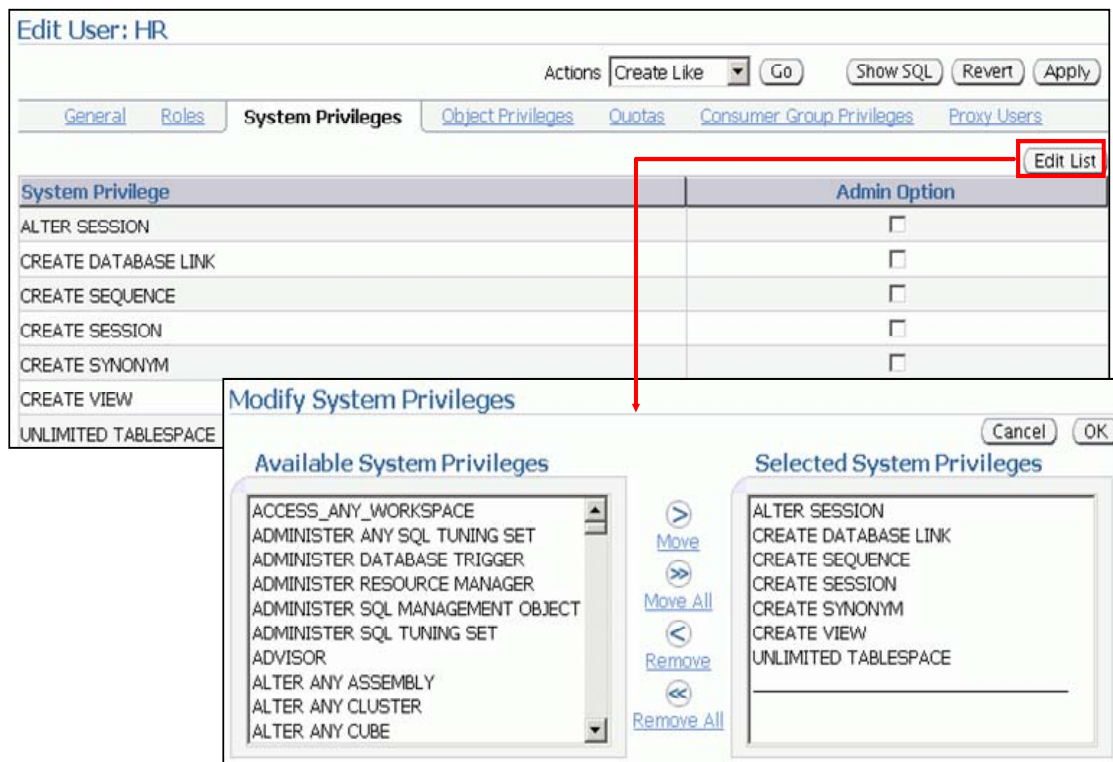
Полномочия

Полномочие — это право на выполнение определенного типа SQL-оператора или на доступ к объекту пользователя. С помощью БД Oracle можно контролировать списки операций в базе данных, которые разрешены и запрещены пользователю.

Полномочия делятся на две категории:

- **Системные полномочия:** каждое системное полномочие позволяет пользователю выполнять определенную операцию в базе данных или класс операций в базе данных. К примеру, полномочие на создание табличных пространств является системным. Системные полномочия предоставляются администратором или пользователем, который владеет явно заданным разрешением на управление полномочиями. Существует более 170 отдельных системных полномочий. Многие из них содержат фразу ANY.
- **Объектные полномочия:** объектное полномочие предоставляет пользователю возможность выполнять определенную операцию над определенным объектом, например таблицей, представлением, последовательностью, процедурой, функцией или пакетом. Без специального разрешения пользователю доступны только собственные объекты. Объектные полномочия может предоставить владелец объекта, администратор или пользователь, которому явным образом задано разрешение на предоставление таких полномочий.

Системные полномочия



ORACLE

© Oracle, 2009. Все права защищены.

Системные полномочия

Чтобы предоставить системные полномочия, на странице «Edit User» (Редактирование пользователя) перейдите на вкладку «Systems Privileges» (Системные полномочия). Выберите необходимые полномочия из списка доступных полномочий, после чего переместите их в список «Selected System Privileges» (Выбранные системные полномочия), щелкнув стрелку «Move» (Переместить).

Предоставление полномочий, которые содержат фразу ANY, означает выход за пределы схемы. Например, обладая полномочиями CREATE TABLE, можно создать таблицу, однако — только в собственной схеме. А полномочия SELECT ANY TABLE позволяют выбирать среди таблиц других пользователей. Пользователь SYS и пользователи, владеющие ролью DBA, обладают всеми полномочиями ANY. Поэтому им доступны любые операции с любыми данными. Область действия системных полномочий ANY можно контролировать с помощью функции Oracle Database Vault.

Если установить флажок «Admin Option» (Параметр администратора), пользователь сможет управлять полномочиями и предоставлять другим пользователям системные полномочия.

Синтаксис SQL для предоставления системных полномочий:

```
GRANT <system_privilege> TO <grantee clause> [WITH ADMIN OPTION]
```

Прежде чем предоставить системные разрешения, следует внимательно обдумать требования безопасности. Некоторые системные полномочия могут предоставляться только администраторам:

- **RESTRICTED SESSION:** это полномочие позволяет входить в систему, даже если БД была открыта в ограниченном режиме.

Системные полномочия (продолжение)

- **SYSDBA и SYSOPER:** эти полномочия позволяют остановить, запустить, восстановить, а также выполнить любые другие административные задачи в базе данных. **SYSOPER** позволяет пользователю выполнять простые оперативные задачи, однако без возможности просмотра пользовательских данных. В него входят следующие системные полномочия:
 - **STARTUP и SHUTDOWN**
 - **CREATE SPFILE**
 - **ALTER DATABASE OPEN/MOUNT/BACKUP**
 - **ALTER DATABASE ARCHIVELOG**
 - **ALTER DATABASE RECOVER** (Только полное восстановление. Для выполнения любой разновидности неполного восстановления, например **UNTIL TIME | CHANGE | CANCEL | CONTROLFILE**, требуется подключение от имени **SYSDBA**.)
 - **RESTRICTED SESSION**

Системное полномочие **SYSDBA** дополнительно дает разрешение на неполное восстановление и удаление базы данных. Фактически, системное полномочие **SYSDBA** позволяет пользователю установить соединение от имени пользователя **SYS**.

- **SYSASM:** это полномочие позволяет запустить, остановить экземпляр **ASM** и управлять им.
- **DROP ANY object:** Полномочие **DROP ANY** позволяет удалять объекты других пользователей схемы.
- **CREATE, MANAGE, DROP и ALTER TABLESPACE:** эти полномочия позволяют управлять табличными пространствами, включая создание, удаление и изменение атрибутов табличных пространств.
- **CREATE LIBRARY:** в БД **Oracle** имеется возможность создавать и вызывать внешний код (например, библиотеку **C**) с помощью **PL/SQL**. Такой библиотеке должно быть присвоено имя объектом **LIBRARY** в базе данных. Полномочие **CREATE LIBRARY** позволяет создавать произвольные библиотеки кода, исполняемые с помощью **PL/SQL**.
- **CREATE ANY DIRECTORY:** в качестве меры безопасности каталог ОС, в котором хранится код, необходимо связать с виртуальным объектом каталога **Oracle**. Владелец полномочия **CREATE ANY DIRECTORY** потенциально может вызвать небезопасные объекты кода.

Полномочие **CREATE ANY DIRECTORY** позволяет создать объект каталога (с доступом на чтение и запись) для любого каталога, доступного владельцу ПО **Oracle**. Это означает, что пользователь с таким полномочием имеет доступ к внешним процедурам этих каталогов. Он может попытаться прочитать или записать любой файл базы данных — файлы данных, журнал восстановления или журнал аудита. В организации обязательно должна быть реализована стратегия безопасности, предотвращающая злоупотребление административными полномочиями, как в приведенном случае.

- **GRANT ANY OBJECT PRIVILEGE:** это полномочие позволяет предоставлять разрешения для объектов, которыми его обладатель не владеет.
- **ALTER DATABASE и ALTER SYSTEM:** это полномочия очень высокого уровня. Они позволяют изменять базу данных и экземпляр **Oracle** (например, можно переименовать файл данных или очистить кэш буфера).

Объектные полномочия

Чтобы предоставить объектные полномочия:

- выберите тип объекта;
- выберите объекты;
- выберите полномочия.

ORACLE

© Oracle, 2009. Все права защищены.

Объектные полномочия

Чтобы предоставить объектные полномочия, на странице «Edit User» (Редактирование пользователя) перейдите на вкладку «Object Privileges» (Объектные полномочия). Выберите тип объекта, на доступ к которому предоставляются полномочия, после чего нажмите кнопку «Add» (Добавить). Выберите объекты из списка или введите их имена следующим образом: `<имя_пользователя.имя_объекта>`.

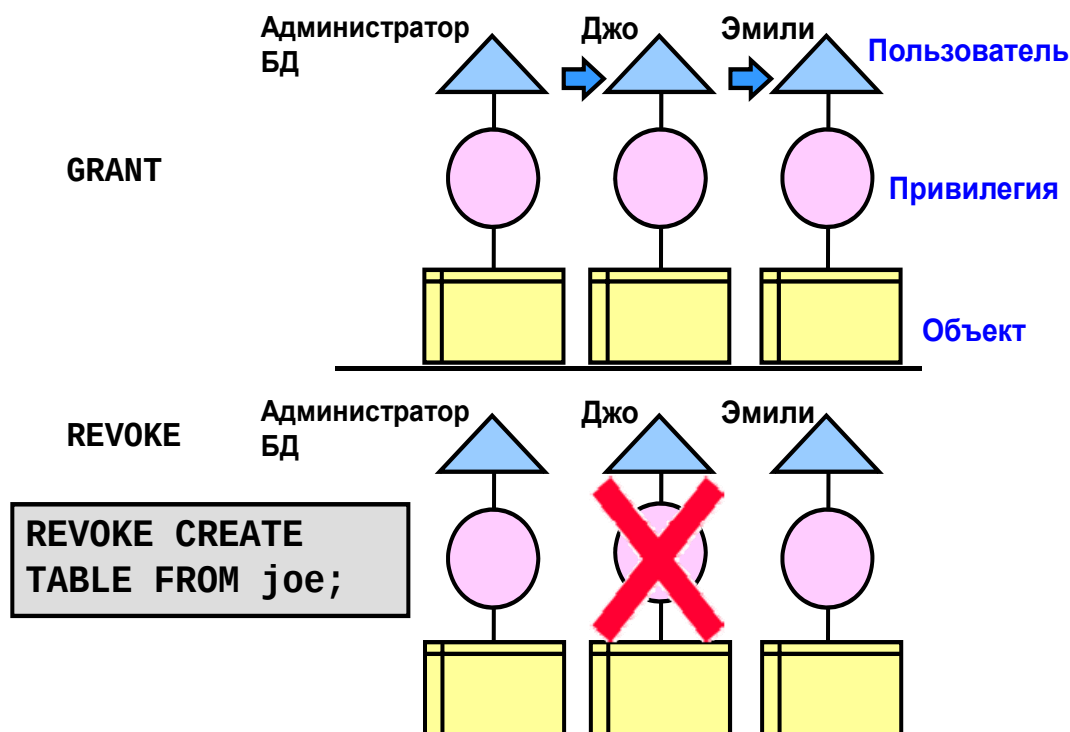
После этого выберите соответствующие полномочия из списка «Available Privileges» (Доступные полномочия) и нажмите кнопку «Move» (Переместить). Выбрав необходимые полномочия, нажмите кнопку «OK».

На странице «Edit User» (Редактирование пользователя) установите флажок «Grant» (Предоставление), если данному пользователю разрешено предоставлять другим пользователям те же права доступа.

Синтаксис SQL для предоставления объектных полномочий:

```
GRANT <object_privilege> ON <object> TO <grantee clause>
[WITH GRANT OPTION]
```

Аннулирование системных полномочий с параметром ADMIN OPTION



© Oracle, 2009. Все права защищены.

Аннулирование системных полномочий с параметром ADMIN OPTION

Если системные полномочия были предоставлены с помощью команды **GRANT**, их можно аннулировать с помощью SQL-оператора **REVOKE**. Пользователи, которым системные полномочия были предоставлены с параметром **ADMIN OPTION**, могут аннулировать полномочия других пользователей базы данных. Аннулировать полномочия может не только пользователь, предоставивший их.

Также при аннулировании системных полномочий отсутствует каскадный эффект за исключением полномочий с параметром **ADMIN OPTION**.

Синтаксис SQL для аннулирования системных полномочий:

```
REVOKE <system_privilege> FROM <grantee clause>
```

На слайде отображена следующая ситуация.

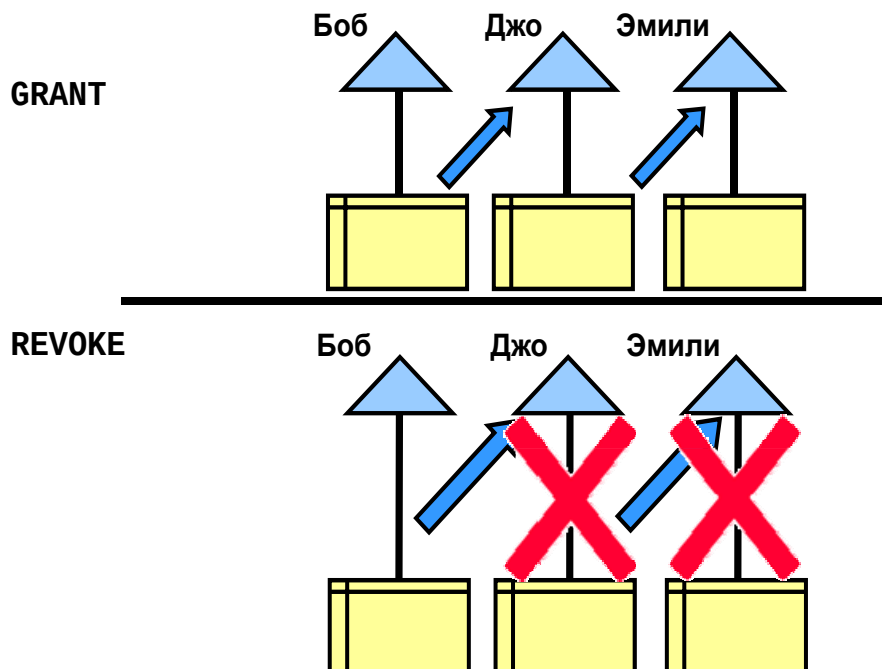
Сценарий

1. Администратор БД предоставляет пользователю Джо системные полномочия **CREATE TABLE** с параметром **ADMIN OPTION**.
2. Джо создает таблицу.
3. Джо предоставляет системные полномочия **CREATE TABLE** пользователю Эмили.
4. Эмили создает таблицу.
5. Администратор БД аннулирует системные полномочия **CREATE TABLE** для Джо.

Результат

Таблица Джо остается, но он больше не может создавать таблицы. Таблица Эмили остается, и она по-прежнему обладает системными полномочиями **CREATE TABLE**.

Аннулирование системных полномочий с параметром GRANT OPTION



ORACLE

© Oracle, 2009. Все права защищены.

Аннулирование объектных полномочий с параметром GRANT OPTION

Каскадный эффект можно наблюдать при аннулировании системных полномочий, связанных с операциями на языке манипулирования данными (DML). Например, если пользователю предоставлено системное полномочие `SELECT ANY TABLE` и этот пользователь создал процедуры с использованием таблиц, следует перекомпилировать все процедуры, которые хранятся в схеме пользователя, прежде чем снова их использовать.

Также каскадный эффект имеет место при аннулировании объектных полномочий с параметром `GRANT OPTION`. Пользователь может аннулировать только предоставленные им самим полномочия. К примеру, Боб не может аннулировать объектное полномочие, которое Джо предоставил Эмили. Только получатель полномочий или пользователь с полномочием, которое называется `GRANT ANY OBJECT PRIVILEGE`, может аннулировать объектные полномочия.

Сценарий

1. Джо предоставлено объектное полномочие `SELECT` на `EMPLOYEES` с параметром `GRANT OPTION`.
2. Джо предоставляет полномочие `SELECT` на `EMPLOYEES` пользователю Эмили.
3. Полномочие `SELECT` для Джо было аннулировано. При этом также аннулируется полномочие Эмили.

Преимущества использования ролей

- Упрощенное управление полномочиями
- Динамическое управление полномочиями
- Выбор доступных полномочий



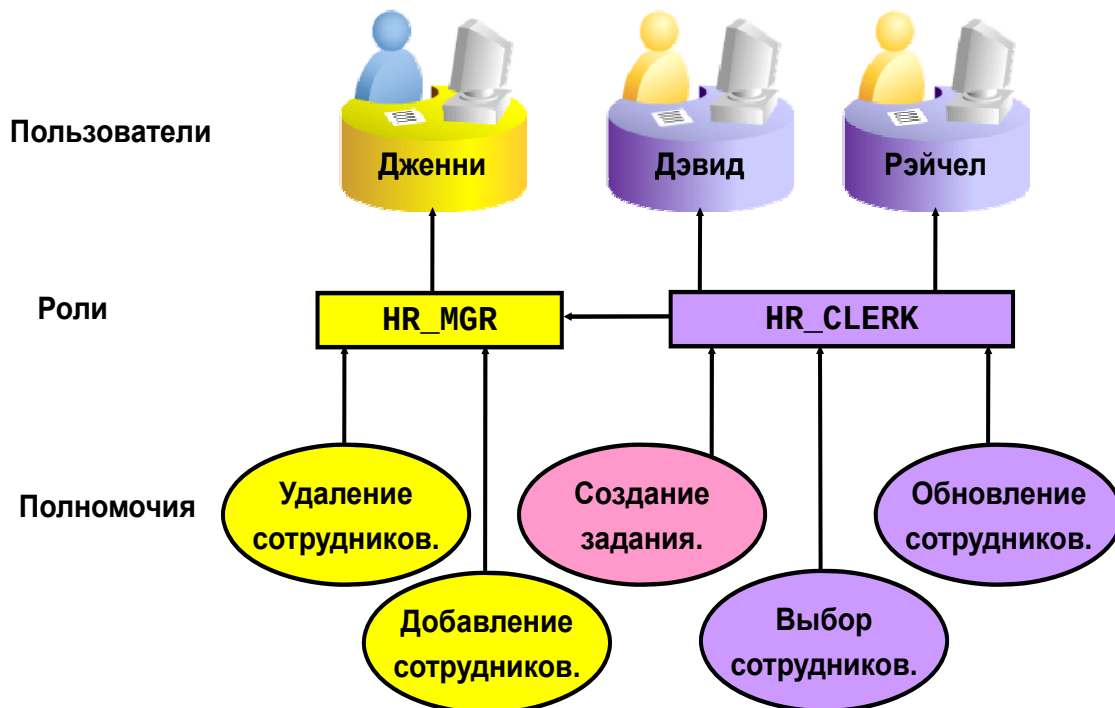
ORACLE

© Oracle, 2009. Все права защищены.

Преимущества использования ролей

- **Упрощенное управление полномочиями:** роли используются для упрощенного управления полномочиями. Вместо того чтобы предоставлять нескольким пользователям один и тот же набор полномочий, можно предоставить эти полномочия роли, а затем предоставить эту роль пользователям.
- **Динамическое управление полномочиями:** если полномочия, связанные с ролью, изменяются, все пользователи, которым предоставлена эта роль, немедленно автоматически получают обновленные полномочия.
- **Выбор доступных полномочий:** роли можно включать и отключать, чтобы временно активировать или деактивировать полномочия. Таким образом можно контролировать полномочия пользователя в той или иной ситуации.

Назначение полномочий ролям и назначение ролей пользователям



ORACLE

© Oracle, 2009. Все права защищены.

Назначение полномочий ролям и назначение ролей пользователям

В большинстве систем индивидуальное предоставление необходимых полномочий каждому пользователю занимает много времени и порождает множество ошибок. ПО Oracle обеспечивает простое и контролируемое управление полномочиями посредством ролей. Роли — это именованные группы связанных полномочий, которые предоставляются пользователям или другим ролям. Роли предназначены для упрощения администрирования полномочий в базе данных и, таким образом, для повышения безопасности.

Характеристики роли

- Полномочия можно предоставить роли или аннулировать их для нее, как если бы это был пользователь.
- Роли можно предоставить или аннулировать для пользователей или других ролей, **как системные полномочия.**
- Роль может состоять из системных и объектных полномочий.
- Роли можно включать и отключать для всех пользователей, которым эти роли предоставлены.
- Для включения роли может потребоваться ввод пароля.
- Роли не принадлежат никому и не хранятся ни в одной из схем.

В примере на слайде роли HR_CLERK предоставлены полномочия SELECT и UPDATE на таблицу employees и системное полномочие CREATE JOB. Роли HR_MGR предоставлены полномочия DELETE и INSERT на таблицу employees и роль HR_CLERK.

Менеджеру предоставлена роль HR_MGR и, таким образом, он может выбирать, удалять, добавлять и обновлять таблицу employees.

Предопределенные роли

Роль	Включаемые полномочия
CONNECT	CREATE SESSION
RESOURCE	CREATE CLUSTER, CREATE INDEXTYPE, CREATE OPERATOR, CREATE PROCEDURE, CREATE SEQUENCE, CREATE TABLE, CREATE TRIGGER, CREATE TYPE
SCHEDULER_ ADMIN	CREATE ANY JOB, CREATE EXTERNAL JOB, CREATE JOB, EXECUTE ANY CLASS, EXECUTE ANY PROGRAM, MANAGE SCHEDULER
Администратор БД	Большая часть системных полномочий; несколько других ролей. Следует предоставлять только администраторам.
SELECT_ CATALOG_ROLE	Системных полномочий нет; HS_ADMIN_ROLE и более 1700 объектных полномочий в словаре данных

ORACLE

© Oracle, 2009. Все права защищены.

Предопределенные роли

Существует несколько ролей, которые автоматически определяются в базах данных Oracle при запуске сценариев создания БД. Роль **CONNECT** автоматически предоставляется всем пользователям, созданным при помощи Enterprise Manager. В целях безопасности, начиная с версии 10.2.0 базы данных Oracle, роль **CONNECT** содержит только полномочия **CREATE SESSION**.

Примечание. Имейте ввиду, что предоставление роли **RESOURCE** также означает предоставление полномочия **UNLIMITED TABLESPACE**.

Функциональные роли

Прочие роли, позволяющие администрировать специальные функции, создаются после установки этих функций. К примеру, роль **XDBADMIN** содержит полномочия, необходимые для администрирования БД на расширяемом языке разметки (XML), если данный компонент установлен. Роль **AQ_ADMINISTRATOR_ROLE** обеспечивает полномочия для управления расширенными очередями. Роль **HS_ADMIN_ROLE** содержит полномочия, необходимые для управления гетерогенными службами.

Не следует изменять полномочия этих функциональных ролей без рекомендаций службы поддержки Oracle, так как это может привести к непреднамеренному отключению необходимых функций.

Создание роли

Выберите пункты «Server» (Сервер) > «Roles» (Роли).

Добавьте полномочия и роли с соответствующей вкладки.

По завершении щелкните «OK».

There is no authentication.

Select Object Type: Table Add

Select	Object Privilege	Schema	Object
☐	SELECT	OE	CUSTOMERS
☐	SELECT	OE	INVENTORIES
☐	SELECT	OE	ORDERS
☐	SELECT	OE	ORDER_ITEMS

ORACLE

© Oracle, 2009. Все права защищены.

Создание роли

Роль — это именованная группа связанных полномочий, которая предоставляется пользователям или другим ролям. С помощью ролей администратор БД управляет полномочиями.

Чтобы создать роль, выполните следующие действия:

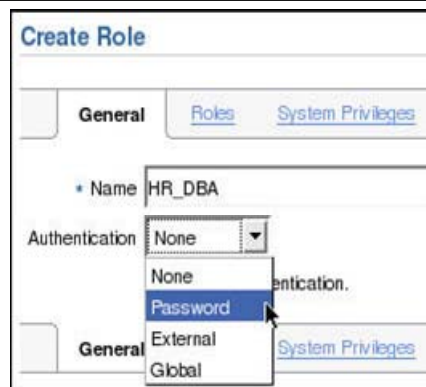
1. В утилите Enterprise Manager Database Control щелкните вкладку «Server» (Сервер), затем в разделе «Security» (Безопасность) выберите пункт «Roles» (Роли).
2. Нажмите кнопку «Create» (Создать).
3. Введите имя роли на вкладке «General» (Общие).
4. При необходимости добавьте системные и объектные полномочия, а также другие роли. В дальнейшем роль можно отредактировать, чтобы изменить эти параметры.
5. По завершении щелкните «OK».

Защита ролей

- Роли могут быть нестандартными и активироваться по мере необходимости.

```
SET ROLE vacationdba;
```

- Роли можно защитить с помощью аутентификации.



- Роли также можно защитить программными средствами.

```
CREATE ROLE secure_application_role  
IDENTIFIED USING <имя_процедуры_безопасности>;
```

ORACLE

© Oracle, 2009. Все права защищены.

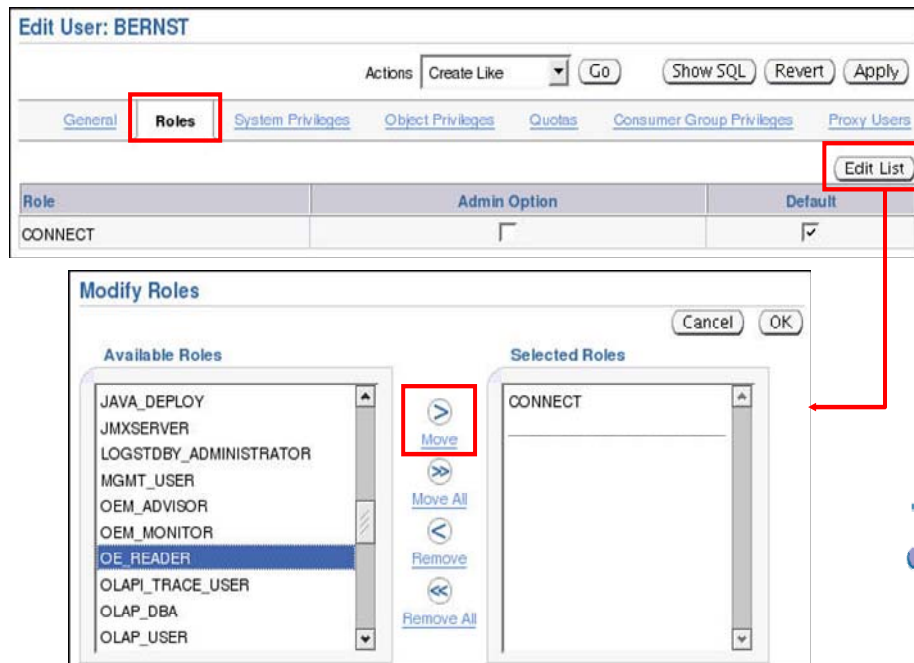
Защита ролей

Обычно роли включены по умолчанию, то есть если пользователю предоставляется роль, этот пользователь может использовать полномочия данной роли. Роли по умолчанию назначаются пользователю при соединении.

Существует возможность:

- сделать роль нестандартной. Для этого при предоставлении роли пользователю следует снять флажок DEFAULT. Теперь, прежде чем пользователь сможет использовать полномочия роли, ее нужно явным образом включить.
- настроить дополнительную аутентификацию для роли. По умолчанию аутентификация для роли не используется, однако можно установить дополнительную аутентификацию для роли при ее настройке.
- создать безопасные роли приложений, включаемые только посредством успешного выполнения процедуры PL/SQL. Эта процедура PL/SQL может проверять, например, сетевой адрес пользователя, программу, запущенную пользователем, время суток и другие данные, необходимые для обеспечения безопасности группы разрешений.
- легко администрировать роли с помощью функции Oracle Database Vault. Используются упрощенные защищенные роли приложений, а также дополнительные ограничения традиционных ролей.

Назначение ролей пользователям



ORACLE

© Oracle, 2009. Все права защищены.

Назначение ролей пользователям

Роли можно использовать для управления полномочиями в базе данных. Можно предоставить полномочия роли, после чего назначить роль пользователю. После этого пользователь может включить роль и использовать ее полномочия. Роль содержит все полномочия, предоставленные непосредственно ей, а также полномочия предоставленных ей других ролей.

По умолчанию Enterprise Manager автоматически предоставляет роль **CONNECT** новым пользователям. Таким образом, пользователи могут соединяться с базой данных и создавать объекты в собственных схемах.

Чтобы назначить пользователю роль, необходимо выполнить следующие действия:

1. В утилите Enterprise Manager Database Control щелкните вкладку «Server» (Сервер), затем в разделе «Security» (Безопасность) выберите пункт «Users» (Пользователи).
2. Выберите пользователя и нажмите кнопку «Edit» (Редактировать).
3. Щелкните вкладку «Roles» (Роли) и нажмите кнопку «Edit List» (Изменить список).
4. Из списка «Available Roles» (Доступные роли) выберите нужные роли и переместите их в список «Selected Roles» (Выбранные роли).
5. Завершив выбор ролей, нажмите кнопку «OK».

Экспресс-тест

Все пароли, создаваемые в Oracle Database 11g, по умолчанию не зависят от регистра.

1. Да
2. Нет



ORACLE

© Oracle, 2009. Все права защищены.

Ответ: 2

Экспресс-тест

Роль базы данных:

1. можно включать или выключать;
2. могут состоять из системных и объектных полномочий;
3. принадлежат создателю;
4. не могут быть защищены паролем

ORACLE

© Oracle, 2009. Все права защищены.

Ответы: 1, 2

Профили и пользователи

Пользователю одновременно назначается только один профиль.

Профили:

- управляют потреблением ресурсов;
- управляют состоянием формуляров и сроком действия паролей.

Create Profile	
General Password	
Name: LIMITED_USER	
Details	
CPU/Session (Sec./100)	1000
CPU/Call (Sec./100)	UNLIMITED
Connect Time (Minutes)	DEFAULT
Idle Time (Minutes)	60
Database Services	
Concurrent Sessions (Per User)	DEFAULT
Reads/Session (Blocks)	DEFAULT
Reads/Call (Blocks)	DEFAULT
Private SGA (KBytes)	DEFAULT
Composite Limit (Service Units)	DEFAULT

Примечание. Параметру `RESOURCE_LIMIT` необходимо присвоить значение `TRUE`, до того как профили смогут накладывать ограничения на использование ресурсов.

ORACLE

© Oracle, 2009. Все права защищены.

Профили и пользователи

Профили представляют собой именованные наборы ограничений на использование ресурсов БД и экземпляра. Также профили регулируют состояние формуляров и пароли пользователей (количество символов, срок действия и т. д.). Каждому пользователю назначается профиль, причем одновременно только один. Если при изменении профиля пользователь уже находился в системе, изменения вступят в силу со следующего сеанса работы пользователя.

Профиль `DEFAULT` является базовым для всех остальных профилей. Как показано на слайде, ограничения для профиля можно указать явным образом (процессор/сеанс), не ограничивать (процессор/вызов) или связать с настройками профиля `DEFAULT` (время соединения).

Профили не накладывают ограничения на использование ресурсов пользователями, если параметру инициализации `RESOURCE_LIMIT` не присвоено значение `TRUE`.

Если параметру `RESOURCE_LIMIT` присвоено стандартное значение `FALSE`, ограничения на использование ресурсов, заданные профилем, игнорируются. Настройки пароля для профиля всегда применяются принудительно.

С помощью профилей администратор может контролировать следующие системные ресурсы:

- **ЦП:** ресурсы процессора можно ограничивать на основе сеансов или вызовов. Ограничение процессор/сеанс, равное 1000, означает, что если какой-либо отдельный сеанс, использующий данный профиль, потребляет более 10 секунд процессорного времени (процессорное время указывается в сотых долях секунды), то данный сеанс вызывает ошибку и выполняет выход:
ORA-02392: exceeded session limit on CPU usage, you are being logged off

Профили и пользователи (продолжение)

Ограничение вызовов выполняет ту же роль, только вместо ограничения всего сеанса пользователя контролируется процессорное время, потребляемое отдельными командами. Если задано ограничение процессор/вызов и команда пользователя превышает его, ее выполнение прерывается. Пользователь получает сообщение об ошибке следующего вида:
ORA-02393: exceeded call limit on CPU usage

- **Сеть/память:** каждый сеанс БД потребляет ресурсы системной памяти и (если это не сеанс пользователя, являющегося локальным для сервера) сетевые ресурсы. Можно задать следующие параметры:
 - **Время соединения:** указывает интервал времени в минутах, в течение которого пользователь может быть соединен до того, как он будет автоматически отключен.
 - **Время простоя:** указывает интервал времени в минутах, в течение которого сеанс пользователя может бездействовать до того, как он будет автоматически отключен. Время простоя вычисляется только для серверного процесса. При этом не учитывается активность приложения. Ограничение времени `IDLE_TIME` не связано с продолжительными запросами и другими операциями.
 - **Параллельные сеансы:** определяет, сколько параллельных сеансов может быть создано с помощью формулы пользователя БД.
 - **Личная область SGA:** ограничивает размер области, выделяемой в глобальной системной области (SGA) для сортировки, слияния растровых изображений и т. д. Данное ограничение действует только при условии, если сеанс использует разделяемый сервер. (Подробная информация о разделяемых серверах приведена в уроке «Настройка сетевой среды Oracle».)
- **Дисковый ввод/вывод:** ограничивает объем данных, которые пользователь может считать за сеанс или за вызов. Показатели чтений/сеанс и чтений/вызов позволяют ограничить общее количество операций считывания из памяти и с диска. Таким образом, имеется возможность предотвратить перегрузку памяти и дисков при выполнении операторов с большим количеством операций ввода/вывода.

Кроме того, для профиля можно задать смешанные ограничения. Смешанные ограничения основываются на сбалансированных сочетаниях значений параметров процессор/сеанс, чтений/сеанс, времени соединения и частной области SGA. Смешанные ограничения подробнее рассмотрены в руководстве *Oracle Database Security Guide* (Руководство по безопасности базы данных Oracle).

Чтобы создать профиль, следует перейти на вкладку «Server» (Сервер), после чего под заголовком «Security» (Безопасность) щелкнуть пункт «Profiles» (Профили). Далее на странице «Profiles» (Профили) необходимо нажать кнопку «Create» (Создать).

Примечание. Диспетчер ресурсов позволяет управлять многими параметрами, которые определяются настройками профиля. Более подробная информация о диспетчере ресурсов приведена в руководстве *Oracle Database Administrator's Guide* (Руководство администратора базы данных Oracle).

Возможности защиты паролем



Примечание. Не следует использовать профили, приводящие к завершению срока действия паролей формуляров SYS, SYSMAN и DBSNMP и их блокированию.

ORACLE

© Oracle, 2009. Все права защищены.

Возможности защиты паролем

В Oracle защита паролем реализована посредством профилей пользователей. Профили обеспечивают многие стандартные функции безопасности.

Блокирование формуляров: обеспечивает автоматическое блокирование формуляров на заданный промежуток времени, если пользователю не удалось выполнить вход в систему за указанное количество попыток.

- **FAILED_LOGIN_ATTEMPTS:** задает количество неудачных попыток входа в систему перед блокированием формуляра.
- **PASSWORD_LOCK_TIME:** задает количество дней, в течение которых формуляр блокируется после определенного числа неудачных попыток входа в систему.

Время существования и срок действия пароля: позволяет определить время существования пароля пользователя, по истечении которого срок действия пароля завершается и пароль следует изменить.

- **PASSWORD_LIFE_TIME:** определяет время существования пароля в днях.
- **PASSWORD_GRACE_TIME:** задает длительность льготного периода (в днях), в течение которого можно изменить пароль после первого успешного входа в систему, осуществленного после истечения срока действия данного пароля.

Примечание. Истечение срока действия паролей и блокирование формуляров SYS, SYSMAN и DBSNMP приведет к неправильной работе Enterprise Manager. Предупреждение об истечении срока действия пароля должны перехватывать приложения, после чего проводить изменение пароля. В противном случае льготный период истечет, и пользователь будет заблокирован без очевидной для него причины.

Возможности защиты паролем (продолжение)

Хронология паролей: позволяет проверить, не используется ли новый пароль повторно в течение заданного периода времени или количества изменений пароля. Эта проверка реализуется с помощью одного из двух параметров:

- **PASSWORD_REUSE_TIME:** указывает, что пользователь не может повторно использовать пароль в течение заданного количества дней.
- **PASSWORD_REUSE_MAX:** задает количество изменений пароля, которые должны быть выполнены, прежде чем текущий пароль можно будет использовать вновь.

Помните, что значения параметров профиля устанавливаются профилем DEFAULT или же наследуются от него.

Если оба параметра хронологии паролей имеют значение UNLIMITED, БД Oracle будет игнорировать оба параметра. Пользователь сможет в любое время повторно использовать любой пароль, что является неудачным решением с точки зрения безопасности.

Если же оба параметра заданы, то повторное использование паролей разрешено, но только при выполнении обоих условий. Пользователю придется изменить пароль заданное число раз, также со времени использования нужного пароля должно будет пройти заданное количество дней.

Например, для профиля ALFRED значение параметра PASSWORD_REUSE_MAX было равно 10, а параметра PASSWORD_REUSE_TIME равно 30. Пользователь ALFRED не сможет повторно использовать пароль, пока не сбросит пароль 10 раз и пока не пройдет 30 дней с момента последнего использования этого пароля.

Если одному параметру присвоено числовое значение, а другому значение UNLIMITED, то пользователь никогда не сможет использовать пароли повторно.

Проверка сложности пароля: обеспечивает проверку сложности пароля на предмет соответствия определенным правилам. Необходимо проверить, достаточно ли сложен пароль, чтобы обеспечить защиту от злоумышленников, намеревающихся взломать систему путем подбора пароля.

Параметр PASSWORD_VERIFY_FUNCTION содержит имя функции PL/SQL, которая выполняет проверку сложности пароля перед его назначением. Функции проверки пароля должны принадлежать пользователю SYS и возвращать логическое значение (TRUE или FALSE). Шаблон функции проверки пароля содержится в сценарии utlpwdmg.sql, расположенном в таких каталогах:

- Для платформ Unix и Linux: \$ORACLE_HOME/rdbms/admin
- Для платформ Windows: %ORACLE_HOME%\rdbms\admin

Создание профиля с паролем

Create Profile

Show SQL Cancel OK

General Password

Password

Expire in (days) 90

Lock (days past expiration) 10

History

Number of passwords to keep 2

Number of days to keep for UNLIMITED

Complexity

Complexity function VERIFY_FUNCTION_11G

Failed Login

Number of failed login attempts to lock after 3

Number of days to lock for 5/1440

ORACLE

© Oracle, 2009. Все права защищены.

Создание профиля с паролем

Чтобы создать профиль с паролем, следует перейти на вкладку «Server» (Сервер), после чего под заголовком «Security» (Безопасность) щелкнуть пункт «Profiles» (Профили). Далее на странице «Profiles» (Профили) необходимо нажать кнопку «Create» (Создать). Чтобы настроить ограничения пароля, необходимо щелкнуть вкладку «Password» (Пароль).

Можно выбрать из списка стандартные значения настроек (используйте значок фонарика для обзора значений) или ввести нестандартные значения.

Все временные периоды задаются в днях, однако можно использовать дробные значения. В дне 1440 минут, таким образом 5/1440 дня равно 5 минутам.

Изменять существующие профили с паролями можно с помощью Enterprise Manager.

Если был запущен сценарий `utlpwdmg.sql`, пользователю доступны функции `VERIFY_FUNCTION` и `VERIFY_FUNCTION_11G`. Если пользователь создал собственную функцию проверки сложности, можно ввести ее имя. Имя функции отсутствует в списке «Select» (Выбор). Если функция вызовет ошибки выполнения, пользователь не сможет изменить пароль.

Удаление профиля с паролем

С помощью Enterprise Manager нельзя удалить профиль, используемый пользователями. Однако, если удалить профиль с параметром `CASCADE` (например, в `SQL*Plus`), то всем пользователям, использовавшим этот профиль, будет автоматически назначен профиль `DEFAULT`.

Стандартная функция проверки пароля: **VERIFY_FUNCTION_11G**

Функция VERIFY_FUNCTION_11G проверяет пароль на соблюдение следующих условий:

- не менее восьми символов
- **отличие от имени пользователя, имени пользователя с числом или имени пользователя, написанного наоборот**
- отличие от имени БД или имени БД с числом
- строка с как минимум одной буквой и одной цифрой
- отличие от предыдущего пароля хотя бы на три символа

Совет. Эту функцию можно использовать как шаблон для создания своей нестандартной функции проверки пароля.



ORACLE

© Oracle, 2009. Все права защищены.

Стандартная функция проверки пароля: VERIFY_FUNCTION_11G

Сервер Oracle использует две функции проверки сложности пароля: VERIFY_FUNCTION и VERIFY_FUNCTION_11g. Они создаются с помощью сценария `<oracle_home>/rdbms/admin/utlpwdmg.sql`. Функция VERIFY_FUNCTION предназначена для пользователей, предпочитающих функцию проверки пароля из предыдущих версий. Функцию проверки сложности пароля следует создать в схеме SYS. Также ее можно использовать в качестве шаблона для собственной функции проверки пароля.

При создании функции VERIFY_FUNCTION сценарий utlpwdmg также вносит изменения в профиль DEFAULT с помощью следующей команды ALTER PROFILE:

```
ALTER PROFILE default LIMIT
PASSWORD_LIFE_TIME 180
PASSWORD_GRACE_TIME 7
PASSWORD_REUSE_TIME UNLIMITED
PASSWORD_REUSE_MAX UNLIMITED
FAILED_LOGIN_ATTEMPTS 10
PASSWORD_LOCK_TIME 1
PASSWORD_VERIFY_FUNCTION verify_function_11g;
```

Помните, что всем новым пользователям по умолчанию присваивается профиль DEFAULT, если только не был указан другой профиль.

Назначение квот пользователям

Если пользователю не предоставлено системное полномочие **UNLIMITED TABLESPACE**, для создания объектов в табличном пространстве такому пользователю следует задать квоту.

Квота может иметь:

- значение в мегабайтах или килобайтах;
- неограниченное значение.

Tablespace	Quota	Value	Unit
EXAMPLE	Value	20	MBytes
INVENTORY	None	0	MBytes
SYSAUX	None	0	MBytes
SYSTEM	None	0	MBytes
TEMP	None	0	MBytes
UNDOTBS1	None	0	MBytes
USERS (Default)	Unlimited	0	MBytes

ORACLE

© Oracle, 2009. Все права защищены.

Назначение квот пользователям

Квота — это допустимый объем пространства в заданном табличном пространстве.

По умолчанию пользователь не обладает квотой ни в одном из табличных пространств.

Существуют три способа предоставления квоты пользователю в табличном пространстве.

- **Неограниченное значение:** пользователь может использовать все доступное место в табличном пространстве.
- **Значение:** количество килобайт или мегабайт, доступных пользователю. Это не гарантирует, что для пользователя будет зарезервировано пространство указанного размера. Данное значение может быть больше или меньше размера текущего свободного места в табличном пространстве.
- Системные полномочия **UNLIMITED TABLESPACE:** замещает все отдельные квоты, установленные для табличных пространств, и предоставляет пользователю неограниченные квоты для всех табличных пространств, включая **SYSTEM** and **SYSAUX**. Данные полномочия следует предоставлять осторожно.

Примечание. Имейте в виду, что предоставление роли **RESOURCE** также означает предоставление этого полномочия.

Назначение квот пользователям (продолжение)

Нельзя предоставлять пользователям квоту для табличных пространств **SYSTEM** или **SYSAUX**. Обычно только пользователям **SYS** и **SYSTEM** доступно создание объектов в табличном пространстве **SYSTEM** или **SYSAUX**.

Для временных табличных пространств, а также табличных пространств отмены квоты не используются. Для вставки, обновления и удаления данных в базе данных Oracle квота не требуется. Единственными пользователями, которым требуется квота, являются формуляры, владеющие объектами базы данных. Довольно часто при установке кода приложения установщик создает формуляры базы данных для владения объектами. Квота требуется только этим формулярам. Только пользователям базы данных можно предоставить право на использование этих объектов без квоты.

- Экземпляр Oracle проверяет квоту, когда пользователь создает или расширяет сегмент.
- Среди операций, присвоенных схеме пользователя, в квоте учитываются только те, которые используют место в табличном пространстве. Операции, в ходе которых пространство назначенного табличного пространства не используется, не влияют на квоту (например, создание представлений или использование временных табличных пространств).
- Квота восполняется после удаления объектов пользователя с помощью фразы **PURGE** или их удаления из корзины.

Применение принципа меньших полномочий

- Защита словаря данных:

```
07_DICTIONARY_ACCESSIBILITY=FALSE
```

- Аннулирование ненужных полномочий для PUBLIC.
- Использование списков управления доступом (ACL) для доступа к сети.
- Ограничение доступа пользователей к каталогам.
- Ограничение административных полномочий пользователей.
- Ограничение удаленной аутентификации для БД:

```
REMOTE_OS_AUTHENT=FALSE
```

ORACLE

© Oracle, 2009. Все права защищены.

Применение принципа меньших полномочий

Принцип меньших полномочий означает, что предоставляться должны только те полномочия, которые позволят эффективно выполнить задачу. Это уменьшает вероятность изменения или просмотра данных (случайно или преднамеренно), к которым пользователь не должен иметь доступа.

Защита словаря данных: по умолчанию параметр `07_DICTIONARY_ACCESSIBILITY` имеет значение `FALSE`. Это значение не стоит менять без веской на то причины, так как оно запрещает пользователям с системными полномочиями `ANY TABLE` доступ к базовым таблицам словаря данных. Также это значение позволяет пользователю `SYS` входить в систему только от имени `SYSDBA`.

Аннулирование ненужных полномочий для PUBLIC: приложение может использовать несколько пакетов для оптимальной работы, однако для обеспечения безопасности требуется правильная конфигурация. `PUBLIC` предоставляет полномочия на выполнение следующих пакетов: `UTL_SMTP`, `UTL_TCP`, `UTL_HTTP` и `UTL_FILE`. В Oracle Database 11g сетевой доступ контролируется списком управления доступом (ACL), с помощью настройки которого определенным пользователям можно разрешить доступ к определенным сетевым службам. По умолчанию сетевой доступ запрещен. Чтобы его разрешить, нужно создать ACL. Доступ к файлу через `UTL_FILE` контролируется на двух уровнях: на уровне ОС за счет разрешений на доступ к файлам и каталогам, и в базе данных за счет объектов `DIRECTORY`, разрешающих доступ к определенным каталогам файловой системы. Объект `DIRECTORY` может предоставляться пользователю только для чтения или для чтения и записи. Полномочия на выполнение прочих пакетов PL/SQL следует внимательно контролировать.

Применение принципа меньших полномочий (продолжение)

Более мощные пакеты, которые могут быть несанкционированно использованы, включают:

- **UTL_SMTP:** разрешает отправлять произвольные сообщения электронной почты, используя базу данных в качестве почтового сервера простого протокола пересылки почты (SMTP). Для управления доступом пользователей к компьютерам используется ACL.
- **UTL_TCP:** разрешает устанавливать исходящие сетевые соединения сервера БД с любой принимающей или ожидающей сетевой службой. Следовательно, между сервером БД и любой ожидающей сетевой службой могут пересылаться произвольные данные. Для управления доступом используется ACL.
- **UTL_HTTP:** позволяют серверу БД запрашивать данные через протокол HTTP. Предоставление данного пакета пользователю может привести к передаче данных через HTML-формы на злоумышленный Web-узел. Необходимо ограничить доступ с помощью ACL.
- **UTL_FILE:** при неправильной конфигурации позволяет доступ на уровне текста к любому файлу ОС хоста. При правильной настройке пакет ограничивает доступ пользователя к определенным каталогам.

Ограничение доступа к каталогам ОС: объект DIRECTORY в базе данных позволяет администратору БД отобразить каталоги в пути ОС и предоставить отдельным пользователям полномочия на доступ к этим каталогам.

Ограничение административных полномочий для пользователей: не стоит предоставлять пользователям больше полномочий, чем им необходимо. Если пользователь не является администратором, ему нельзя предоставлять роль DBA. Для реализации принципа меньших полномочий необходимо ограничить:

- предоставление системных и объектных полномочий;
- подключения к БД с полномочиями SYS, например SYSDBA и SYSOPER
- ; предоставление других полномочий уровня администратора, например DROP ANY TABLE

. Ограничение удаленной аутентификации для БД: по умолчанию параметр REMOTE_OS_AUTHENT имеет значение FALSE. Его не следует изменять, если только все клиенты не достаточно надежны для аутентификации пользователей. После появления Secure External Password Store (доступен в Oracle Database 10g выпуска 2) появилось мало оправданных причин разрешать удаленную аутентификацию ОС.

При удаленной аутентификации:

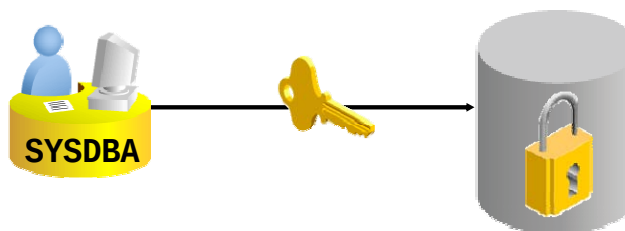
- пользователь БД проходит внешнюю аутентификацию;
- аутентификацию пользователя выполняет удаленная система;
- пользователь выполняет вход в БД без последующей аутентификации.

Примечание. В случае аннулирования полномочий всегда тщательно выполняйте тестирование приложений.

Защита формуляров с большими полномочиями

Для защиты формуляров с большими полномочиями используются:

- файл паролей, содержащий пароли, чувствительные к регистру;
- строгая аутентификация ролей администратора.



ORACLE

© Oracle, 2009. Все права защищены.

Настройка аутентификации администраторов БД

Пользователи с полномочиями SYSDBA, SYSOPER или SYSASM всегда должны проходить аутентификацию. При локальном соединении пользователь проходит только аутентификацию ОС, поскольку входит в группу с полномочиями ОС. При удаленном соединении для аутентификации пользователей с полномочиями используется файл паролей. Если настроено использование файла паролей, то он проверяется в первую очередь. В Oracle Database 11g эти пароли чувствительны к регистру. Oracle Database 11g поддерживает и другие методы, повышающие безопасность удаленной аутентификации пользователей с полномочиями и позволяющие централизованно администрировать этих пользователей.

Если база данных создана с помощью Database Configuration Assistant, пароли в файле паролей чувствительны к регистру. Если было выполнено обновление БД более ранней версии, необходимо проверить, что для удаленной аутентификации используется файл с паролями, которые чувствительны к регистру:

```
orapwd file=orapworcl entries=5 ignorecase=N
```

Если считается, что файл паролей слишком уязвим или обрабатывать много файлов паролей слишком обременительно, можно использовать строгую аутентификацию. Для использования строгих методов аутентификации требуется функция дополнительной защиты. Более подробная информация о строгой аутентификации представлена в руководстве *Oracle Database Advanced Security Administrator's Guide* (Руководство администратора по расширенным функциям обеспечения безопасности базы данных Oracle).

Экспресс-тест

Принципа меньших полномочий недостаточно для обеспечения надежности базы данных Oracle.

1. Да
2. Нет

ORACLE

© Oracle, 2009. Все права защищены.

Ответ: 1

Экспресс-тест

Если параметру RESOURCE_LIMIT присвоено стандартное значение FALSE, ограничения на использование ресурсов, заданные профилем, игнорируются.

1. Да
2. Нет

ORACLE

© Oracle, 2009. Все права защищены.

Ответ: 2

Заключение

В ходе прохождения урока вы научились:

- создавать и управлять формулярами пользователей БД для:
 - аутентификации пользователей
 - присвоения областей хранения по умолчанию (табличных пространств)
- предоставлять и отменять полномочия
- создавать роли и управлять ими
- создавать профили и управлять ими:
 - использовать возможности стандартной защиты паролем
 - контролировать использование ресурсов пользователями

ORACLE

© Oracle, 2009. Все права защищены.

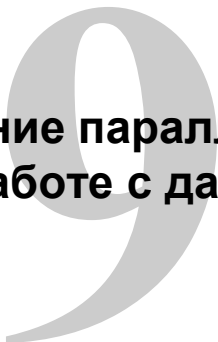
Обзор упражнения 8: администрирование пользователей

В данном упражнении рассматриваются следующие темы:

- Создание профиля для ограничения потребления ресурсов.
- **Создание двух ролей:**
 - HRCLERK
 - HRMANAGER
- Создание четырех новых пользователей:
 - одного менеджера и двух служащих;
 - одного пользователя схемы для следующего упражнения.

ORACLE

© Oracle, 2009. Все права защищены.



Управление параллелизмом при работе с данными

ORACLE

© Oracle, 2009. Все права защищены.

Цели

По завершении данного урока вы сможете:

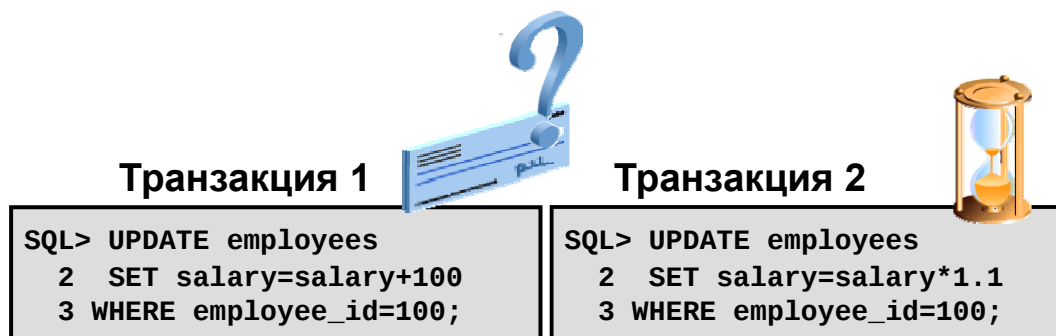
- описать механизм блокирования и метод управления параллелизмом Oracle при работе с данными;
- отслеживать и устранять конфликты блокировок.

ORACLE

© Oracle, 2009. Все права защищены.

Блокировки

- Не позволяют нескольким сеансам одновременно изменять одни и те же данные.
- Автоматически срабатывают на самом низком уровне, доступном для данного оператора.
- **Не расширяются до более высоких уровней.**



ORACLE

© Oracle, 2009. Все права защищены.

Блокировки

Перед тем как база данных позволит сеансу изменить данные, он должен заблокировать изменяемые данные. Блокировка дает сеансу монопольные права на работу с данными, благодаря чему другая транзакция не сможет изменить заблокированные данные до тех пор, пока блокировка не будет снята.

Транзакции могут блокировать отдельные строки, наборы строк или даже целые таблицы. База данных Oracle поддерживает как ручную, так и автоматическую блокировку. Автоматически созданные блокировки всегда действуют на самом нижнем уровне, чтобы минимизировать вероятность возникновения потенциальных конфликтов с другими транзакциями.

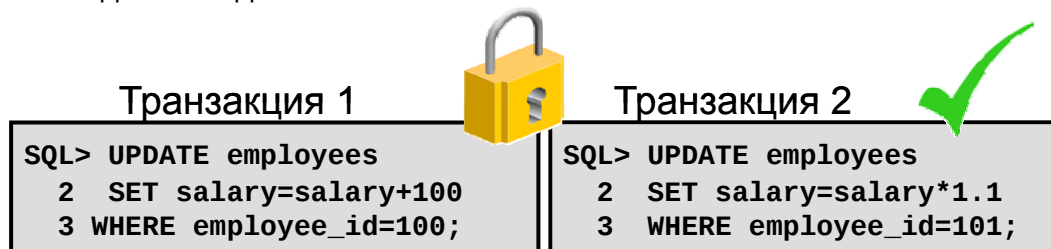
Примечание. Существуют различные типы блокировок, которые экземпляр Oracle использует для поддержания внутренней целостности. В этом курсе мы рассмотрим только блокировку, которая используется для защиты строк и таблиц.

Механизм блокирования

- Высокоуровневый параллелизм при работе с данными:
 - блокировки на уровне строки для операций вставки, обновления или удаления
 - для запросов блокировки не требуются
- Автоматическое управление очередью
- Блокировка сохраняется до завершения транзакции (с помощью оператора COMMIT или ROLLBACK)

Пример.

Предположим, что строки для столбцов employee_id 100 и 101 находятся в одном и том же блоке:



ORACLE

© Oracle, 2009. Все права защищены.

Механизм блокирования

Механизм блокирования разработан для обеспечения максимально возможной степени параллельного доступа к данным базы данных. Для транзакций, которые изменяют данные, используются блокировки на уровне строки, а не на уровне блока или таблицы. При изменениях объектов (например, перемещении таблицы) возникают блокировки на уровне объекта, при этом база данных или схема не блокируются.

При запросе данных блокировки не возникают. Кроме того, запрос будет выполнен успешно даже в том случае, если запрашиваемые данные уже заблокированы (будут отображены первоначальные значения, восстановленные из данных отмены операций и существовавшие до создания блокировки).

В ситуациях, когда несколько транзакций пытаются заблокировать один ресурс, блокировку создаст первая транзакция, которая попыталась обратиться к нему. Другие транзакции будут ожидать окончания первой транзакции. Механизм обслуживания очередей работает автоматически и не требует вмешательства администратора.

Все блокировки снимаются при завершении транзакций (то есть после выполнения оператора COMMIT или ROLLBACK). Если транзакцию выполнить не удалось, тот же фоновый процесс автоматически отменит все изменения, внесенные незаконченной транзакцией, и снимет все созданные ею блокировки.

Параллелизм при работе с данными

Время: 09:00:00	Транзакция 1	UPDATE hr.employees SET salary=salary+100 WHERE employee_id=100;
	Транзакция 2	UPDATE hr.employees SET salary=salary+100 WHERE employee_id=101;
	Транзакция 3	UPDATE hr.employees SET salary=salary+100 WHERE employee_id=102;
	...	...
	Транзакция x	UPDATE hr.employees SET salary=salary+100 WHERE employee_id=xxx;

ORACLE

© Oracle, 2009. Все права защищены.

Параллелизм при работе с данными

Механизм блокирования по умолчанию хорошо гранулирован и применяется на уровне строк. Разные транзакции могут обновлять различные строки одной таблицы, не влияя друг на друга.

По умолчанию блокирование осуществляется на уровне строки, однако база данных Oracle поддерживает ручные блокировки и на более высоких уровнях, если в них возникнет необходимость:

```
SQL> LOCK TABLE employees IN EXCLUSIVE MODE;  
Table(s) Locked.
```

Согласно вышеприведенному примеру, любая транзакция, которая попытается обновить строку в заблокированной таблице, будет вынуждена дожидаться завершения транзакции, создавшей блокировку. **EXCLUSIVE** является самым строгим режимом блокирования.

Существуют и другие режимы блокирования:

- **ROW SHARE:** делает возможным параллельный доступ к данным заблокированной таблицы и не дает сеансам заблокировать всю таблицу для монопольного доступа.
- **ROW EXCLUSIVE:** этот режим подобен режиму **ROW SHARE**, однако запрещает блокирование в режиме **SHARE**. Блокировки типа **ROW EXCLUSIVE** создаются автоматически при обновлении, вставке или удалении данных. Блокировки типа **ROW EXCLUSIVE** разрешают работу множеству процессов чтения и одному процессу записи.

Параллелизм при работе с данными (продолжение)

- **SHARE:** разрешает параллельные очереди, но запрещает обновления заблокированной таблицы. Блокировка **SHARE** необходима (и запрашивается автоматически) для создания индекса таблицы. Для оперативного создания индекса требуется блокировка **ROW SHARE**, которая используется при построении индекса.
Разделяемые блокировки позволяют работать с данными несколькими процессами чтения, а процессам записи запрещают. Разделяемые блокировки используются прозрачно при удалении или обновлении строк в родительской таблице, которая имеет дочернюю таблицу с ограничениями внешнего ключа для родительской таблицы.
- **SHARE ROW EXCLUSIVE:** используется для опроса всей таблицы и позволяет другим опрашивать строки таблицы, но при этом запрещает им блокировать таблицу в режиме **SHARE** или обновлять строки.
- **EXCLUSIVE:** разрешает опрашивать заблокированную таблицу и запрещает любые манипуляции над ней. Блокировка **EXCLUSIVE** необходима при удалении таблицы.

Подобно любому запросу на блокировку, операторы ручной блокировки ожидают, пока все сеансы, уже имеющие блокировки или создавшие запрос на блокировку, снимут свои блокировки. Команда **LOCK** может принимать особый аргумент, который управляет режимом ожидания **NOWAIT**.

NOWAIT немедленно возвращает контроль над указанной таблицей, если она уже была заблокирована другим сеансом:

```
SQL> LOCK TABLE hr.employees IN SHARE MODE NOWAIT;  
LOCK TABLE hr.employees IN SHARE MODE NOWAIT  
*
```

```
ERROR at line 1:
```

```
ORA-00054: resource busy and acquire with NOWAIT specified
```

Обычно не возникает необходимости в ручной блокировке объектов. Механизм автоматического блокирования обеспечивает параллельный доступ к данным для большинства приложений. Oracle рекомендует избегать применения ручных блокировок, особенно при разработке приложений. Неадекватное использование высоких уровней блокирования может негативно сказаться на производительности.

DML-блокировки

Транзакция 1

```
SQL> UPDATE employees
  2 SET salary=salary*1.1
  3 WHERE employee_id= 107;
1 row updated.
```

Транзакция 2

```
SQL> UPDATE employees
  2 SET salary=salary*1.1
  3 WHERE employee_id= 106;
1 row updated.
```

Каждая DML-транзакция должна получить две блокировки:

- блокировку EXCLUSIVE на уровне строки для обновляемой строки или строк
- блокировку ROW EXCLUSIVE (RX) на уровне таблицы (TM) для таблицы, содержащей строки

ORACLE

© Oracle, 2009. Все права защищены.

DML-блокировки

Каждая DML-транзакция использует две блокировки:

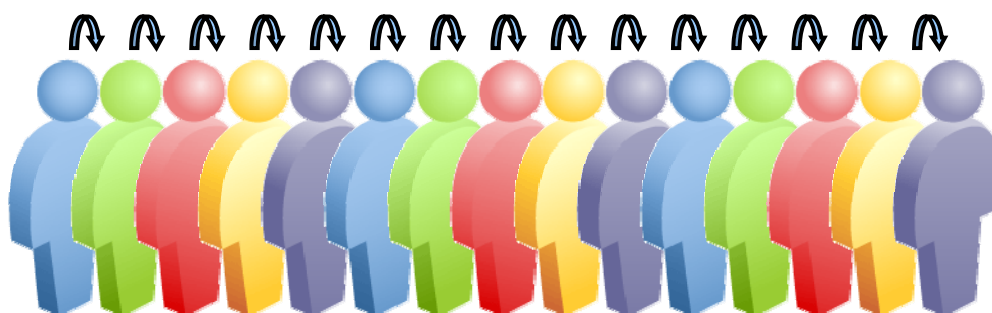
- блокировку EXCLUSIVE на уровне строки для обновляемой строки или строк
- блокировку ROW EXCLUSIVE (RX) на уровне таблицы (TM) для обновляемой таблицы. Это не даст другому сеансу заблокировать всю таблицу (возможно, для ее удаления или усечения) при внесении изменений. Этот режим также называется субмонопольной блокировкой на уровне таблицы (SX).

Блокировка ROW EXCLUSIVE на уровне таблицы не позволяет командам DDL изменять метаданные словаря при обработке неподтвержденной транзакции. Это позволяет поддерживать целостность словаря и согласованность данных на протяжении жизненного цикла транзакции.

Механизм обслуживания очереди

Механизм постановки в очередь отслеживает:

- сеансы, ожидающие блокировки
- запрашиваемый режим блокирования
- порядок, в котором сеансы запрашивали блокировки



ORACLE

© Oracle, 2009. Все права защищены.

Механизм обслуживания очереди

Очередь запросов на блокировки обслуживается автоматически. После завершения транзакции, создавшей блокировку, следующий в очереди сеанс получает право на создание блокировки.

Механизм обслуживания очереди отслеживает порядок, в котором поступали запросы на **создание олокировок**, а также запрашиваемые режимы олокировок.

Сеанс, который уже создал блокировку, может создать запрос на *преобразование блокировки*, не переходя в конец очереди. Например, сеанс создал блокировку SHARE на уровне таблицы. Сеанс может создать запрос на преобразование блокировки SHARE в блокировку типа EXCLUSIVE. Если никакая другая транзакция в таблице еще не имеет блокировок типа EXCLUSIVE или SHARE, то сеансу, который создал блокировку SHARE, будет предоставлено право на создание блокировки EXCLUSIVE вне очереди.

Примечание. Существуют две категории участников очереди: участники без разделяемого владения и участники с разделяемым владением, которые не поднимают уровень блокировки. Участники второй категории известны как *конверторы*. Они всегда имеют более высокий приоритет, чем остальные участники очереди, даже если они ожидают меньшее количество времени.

Конфликты блокировок

Транзакция 1	Время	Транзакция 2
UPDATE employees SET salary=salary+100 WHERE employee_id=100; 1 row updated.	9:00:00	UPDATE employees SET salary=salary+100 WHERE employee_id=101; 1 row updated.
UPDATE employees SET COMMISSION_PCT=2 WHERE employee_id=101; Сеанс ожидает постановки в очередь из-за конфликта блокировок.	9:00:05 	SELECT sum(salary) FROM employees; SUM(SALARY) ----- 692634
Сеанс все еще ожидает!	16:30:00	Множество операций выборки, вставки, обновления и удаления за последние 7,5 часов, но без каких-либо фиксаций или откатов!
1 row updated. Сеанс продолжается.	16:30:01	commit;

ORACLE

© Oracle, 2009. Все права защищены.

Конфликты блокировок

Конфликты блокировок обычно устраняются временем и механизмом обслуживания очереди. В некоторых редких случаях для устранения конфликта блокировок может потребоваться вмешательство администратора. В примере, представленном на втором слайде, вторая транзакция создает блокировку для одной строки в 9:00:00 и при этом не подтверждает изменения, из-за чего блокировка не снимается. Первая транзакция пытается обновить всю таблицу в 9:00:05, для чего необходимо создать блокировки для всех строк. Первая транзакция заблокирована второй транзакцией, пока та не подтвердила изменения в 16:30:01.

Пользователь, пытаясь осуществить первую транзакцию, почти наверняка свяжется с администратором, который должен выяснить причину конфликта и устранить ее.

Возможные причины конфликтов блокировок

- Неподтвержденные изменения
- Транзакции, которые обрабатываются длительное время
- Неадекватно высокие уровни блокирования



ORACLE

© Oracle, 2009. Все права защищены.

Возможные причины конфликтов блокировок

Наиболее распространенной причиной возникновения конфликта блокировок являются неподтвержденные изменения, однако существуют и другие причины:

- **Транзакции, которые обрабатываются длительное время:** множество приложений используют пакетную обработку для выполнения массовых обновлений. Такие операции обычно выполняются по расписанию в интервалы времени с низкой активностью пользователей или при ее отсутствии, но в некоторых случаях операция может не завершиться вовремя, или для ее завершения может потребоваться значительно больше времени, чем было рассчитано. Конфликты блокировок довольно часто возникают при одновременном выполнении операций пакетной обработки.
- **Неадекватно высокие уровни блокирования:** не все базы данных поддерживают низкоуровневое блокирование (Oracle получил поддержку этой функции в 6 версии, которая вышла в 1988 году). Некоторые базы данных все еще осуществляют блокировки на уровне страницы или таблицы. Разработчики приложения обычно используют при создании приложений множество различных баз данных с искусственно завышенными уровнями блокирования, из-за чего база данных Oracle вынуждена делать то же самое для обеспечения совместимости с менее гибкими системами баз данных. Разработчики, которые не имеют достаточного опыта работы с базами данных Oracle, также иногда используют без необходимости более высокие уровни блокирования, чем необходимо.

Выявление конфликтов блокировок

Выберите «Blocking Sessions» (Блокирующие сеансы) на странице «Performance» (Производительность).

Blocking Sessions											
Page Refreshed Aug 18, 2008 11:04:23 PM MDT Refresh											
View Session Kill Session											
Expand All Collapse All											
Select Username	Sessions Blocked	Session ID	Serial Number	SQL ID	Wait Class	Wait Event	P1 Value	P2 Value	P3 Value	Seconds In Wait	
Blocking Sessions											
BERNST	1	114	33091		Idle	SQL*Net message from client	1650815232	1	0	89	
SMAVRIS	0	124	46897	0tqktcvhr5fcf	Application	enq: TX - row lock contention	1415053318	65545	3085	69	

Щелкните ссылку «Идентификатор сеанса», чтобы просмотреть сведения о блокировках сеансов, включая текущий SQL-оператор.

ORACLE

© Oracle, 2009. Все права защищены.

Выявление конфликтов блокировок

Для выявления конфликтов блокировок воспользуйтесь страницей «Blocking Sessions» (Блокирующие сеансы) в Enterprise Manager. Конфликтующие запросы на создание блокировок будут отображены в иерархическом порядке со сведениями о сеансе, создавшем блокировку (вверху) и сеансах, которые создали запросы на блокировки (внизу).

для каждого сеанса, участвующего в конфликте, будет отображено имя пользователя, идентификатор сеанса и время ожидания в секундах. Перейдите к идентификатору сеанса, чтобы отобразить операторы SQL, которые выполняются или были вызваны сеансом.

Автоматический монитор диагностики базы данных (Automatic Database Diagnostic Monitor — ADDM) автоматически выявляет конфликты блокировок и дает рекомендации относительно неэффективных приемов блокирования.

Устранение конфликтов блокировок

Для разрешения конфликта блокировок:

- сеанс, создавший блокировку, должен подтвердить или отменить изменения
- следует прервать сеанс, создавший блокировку (при крайней необходимости)



ORACLE

© Oracle, 2009. Все права защищены.

Устранение конфликтов блокировок

Чтобы устранить конфликт блокировок, сеанс должен освободить созданную блокировку. Наилучший способ достичь этого — связаться с пользователем и попросить его завершить транзакцию.

В крайнем случае администратор может прервать сеанс, создавший блокировку, нажав кнопку «**Kill Session**» (**удалить сеанс**). Помните, что при таком завершении сеанса все изменения, внесенные данной транзакцией, будут потеряны (будет выполнен откат).

Пользователю завершенного сеанса нужно будет повторно подключиться к базе данных и повторить все операции, которые были выполнены после последней операции подтверждения.

Пользователи, сеанс которых был прерван, получают следующее сообщение об ошибке при попытке выполнить новый SQL-оператор:

```
ORA-03135: connection lost contact
```

Примечание. Снайпер сеансов PMON может автоматически прерывать сеансы по истечении заданного времени простоя. Также это можно делать с помощью профилей или диспетчера ресурсов.

Устранение конфликтов блокировок с помощью SQL

Чтобы выявить и прервать заблокированный сеанс, можно использовать операторы SQL.

1 SQL> select SID, SERIAL#, USERNAME
from V\$SESSION where SID in
(select BLOCKING_SESSION from V\$SESSION)

Результат:

SID	SERIAL#	USERNAME
144	8982	HR

2 SQL> alter system kill session '144,8982' immediate;

ORACLE

© Oracle, 2009. Все права защищены.

Устранение конфликтов блокировок с помощью SQL

Управление сеансами, как и большинство других заданий, можно осуществлять не только в Enterprise Manager, но и с помощью операторов SQL. Таблица V\$SESSION содержит сведения обо всех подключенных сеансах. Значение BLOCKING_SESSION — это идентификатор заблокированного сеанса. Если запросить SID и SERIAL# (где SID совпадает с идентификатором заблокированного сеанса), вы получите всю необходимую информацию для выполнения операции kill session (удаления сеанса).

Примечание. Для автоматического отключения сеансов, которые бездействуют и блокируют другие сеансы, можно использовать диспетчер ресурсов базы данных (Database Resource Manager).

Взаимные блокировки

Транзакция 1			Транзакция 2	
UPDATE employees SET salary = salary x 1.1 WHERE employee_id = 1000;	9:00		UPDATE employees SET manager = 1342 WHERE employee_id = 2000;	
UPDATE employees SET salary = salary x 1.1 WHERE employee_id = 2000;	9:15		UPDATE employees SET manager = 1342 WHERE employee_id = 1000;	
ORA-00060: Deadlock detected while waiting for resource	9:16			

ORACLE

© Oracle, 2009. Все права защищены.

Взаимные блокировки

Взаимная блокировка — это особый случай конфликта блокировок. Взаимные блокировки возникают при участии двух (или больше) сеансов, когда сеансы ожидают данные, взаимно заблокированные друг другом. Так как каждый из сеансов ожидает данные, заблокированные противоположной стороной, ни один из сеансов не может завершить транзакцию и устранить конфликт.

База данных Oracle автоматически выявляет взаимные блокировки и прерывает выполнение оператора с сообщением об ошибке. Реакцией на ошибку должна быть операция подтверждения или отмены изменений, которая освободит блокировку одного из сеансов, благодаря чему второй сеанс сможет завершить свою транзакцию.

В примере, приведенном в данном слайде, в ответ на ошибку о возникновении взаимной блокировки транзакция 1 должна выполнить операцию подтверждения или отмены изменений. В случае подтверждения потребуется еще раз выполнить второе обновление, чтобы успешно завершить транзакцию. В случае отката потребуется выполнить оба обновления, чтобы успешно завершить транзакцию.

Экспресс-тест

Механизм блокирования по умолчанию хорошо гранулирован и применяется на уровне строк.

1. Да
2. Нет

ORACLE

© Oracle, 2009. Все права защищены.

Ответ: 1

Экспресс-тест

При возникновении взаимной блокировки база данных Oracle автоматически:

1. Ожидает 300 секунд до завершения обоих сеансов
2. Завершает одну инструкцию с ошибкой в одном сеансе
3. Завершает инструкции с ошибкой в обоих сеансах
4. Не выполняет никаких действий по умолчанию и оставляет это АДБ

ORACLE

© Oracle, 2009. Все права защищены.

Ответ: 2

Заключение

В ходе прохождения урока вы научились:

- описывать механизм блокирования и метод управления параллелизмом Oracle при работе с данными;
- отслеживать и устранять конфликты блокировок.

ORACLE

© Oracle, 2009. Все права защищены.

Обзор упражнения 9: управление данными и параллельным доступом

В данном упражнении рассматриваются следующие темы:

- выявление конфликтов блокировок
- устранение конфликтов блокировок

ORACLE

© Oracle, 2009. Все права защищены.

10

Управление данными отмены операций

ORACLE®

© Oracle, 2009. Все права защищены.

Цели

По завершении данного урока вы сможете:

- объяснять DML и происхождение данных отмены операций;
- **отслеживать и администрировать данные отмены операций;**
- описывать различия между данными отмены операций и данными повторов;
- настраивать время хранения данных отмены операций;
- гарантировать актуальность данных отмены операций в течение заданного времени;
- использовать помощник Undo Advisor.

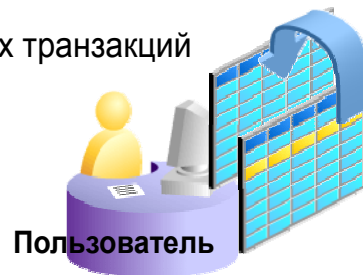
ORACLE

© Oracle, 2009. Все права защищены.

Данные отмены операций

Данные отмены операций:

- являются копией оригинальных (неизмененных) данных
- сохраняются для *каждой* транзакции, которая изменяет данные
- сохраняются как минимум до завершения транзакции
- используются для:
 - операций отката
 - обеспечения целостности данных при чтении
 - запросов в режиме Oracle Flashback, транзакций в режиме Oracle Flashback и таблиц Oracle Flashback
 - восстановления данных после сбойных транзакций



ORACLE

© Oracle, 2009. Все права защищены.

Данные отмены операций

База данных Oracle сохраняет исходные значения данных (данные отмены операций), когда процесс изменяет данные в базе данных. Данные сохраняются до внесения в них изменений. Сохранение данных отмены операций позволяет выполнить откат неподтвержденных данных. Данные отмены операций поддерживают целостность данных при чтении и запросы с моментальным откатом. Данные отмены операций могут использоваться для отмены (моментального отката) транзакций и таблиц.

Запросы с сохранением целостности данных при чтении возвращают результаты с сохранением целостности данных на момент обработки запроса. Для успешного выполнения запросов с сохранением целостности данных при чтении необходимо, чтобы исходные значения данных были сохранены в виде данных отмены операций. Если исходные данные недоступны, будет получена ошибка «Snapshot too old» (Снимок слишком старый) (ORA-01555). База данных Oracle может реконструировать данные для обеспечения целостности при чтении до тех пор, пока сохраняются данные отмены операций.

Данные отмены операций (продолжение)

Опросы с моментальным откатом обращаются к версии данных, которая была актуальна некоторое время назад. До тех пор, пока данные отмены операций существуют для запрошенного отрезка времени, можно будет успешно выполнять запросы с моментальным откатом. Транзакции в режиме Oracle Flashback используют данные отмены операций для создания компенсирующих транзакций, с помощью которых можно будет откатить транзакцию и зависимые от нее транзакции. Таблица Oracle Flashback позволит восстановить состояние таблицы на заданный интервал времени.

Данные отмены операций также используются для восстановления данных после сбойных транзакций. Сбойные транзакции возникают в момент непланового завершения сеанса пользователя (например, из-за сетевых ошибок или сбоя на клиентском компьютере), из-за чего пользователю не удастся подтвердить или откатить транзакцию. Сбойные транзакции также могут возникать при сбое экземпляра или в результате выполнения команды SHUTDOWN ABORT.

В случае возникновения сбойной транзакции база данных Oracle отменяет все изменения, которые внес пользователь, и восстанавливает исходные значения данных.

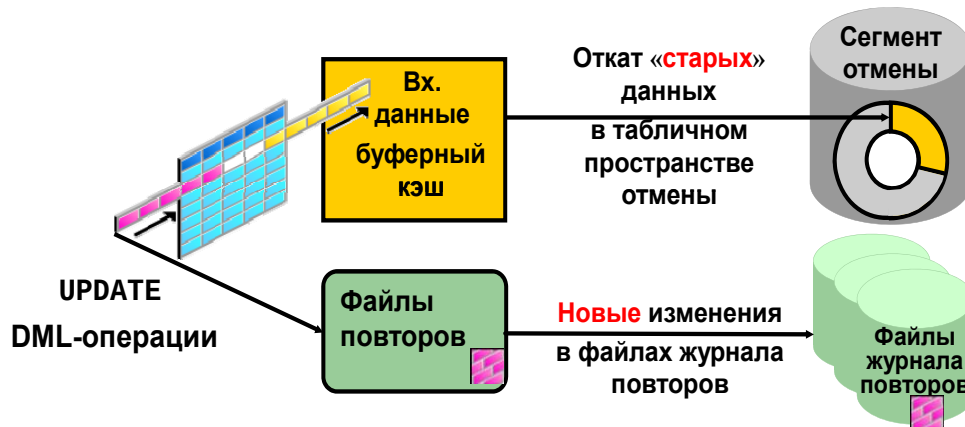
Данные отмены операций сохраняются для всех транзакций как минимум до завершения транзакции по одной из причин:

- Пользователь отменил транзакцию (откат транзакции).
- Пользователь завершил транзакцию (фиксация транзакции).
- Пользователь выполнил DDL-оператор, такой как CREATE, DROP, RENAME или ALTER. Если текущая транзакция содержит DML-операторы, база данных сначала подтвердит транзакцию, а затем выполнит и подтвердит операторы DDL в виде новой транзакции.
- Произошло неплановое завершение сеанса пользователя (откат транзакции).
- Произошло запланированное завершение сеанса пользователя (фиксация транзакции).

Объем хранимых данных отмены операций и время хранения зависят от уровня активности базы данных и ее конфигурации.

Примечание. Транзакции в режиме Oracle Flashback используют интерактивные журналы повторов для предотвращения выполнения соответствующей отмены SQL. Функция «Undo» (Отмена) используется только в качестве искусственного временного ограничения для определения времени начала предотвращения повтора целевой транзакции, если время начала транзакции не поддерживается вызовом транзакции в режиме Flashback.

Транзакции и данные отмены операций



- Каждой транзакции назначается только один сегмент отмены операций.
- Сегмент отмены операций одновременно может обслуживать несколько транзакций.

ORACLE

© Oracle, 2009. Все права защищены.

Транзакции и данные отмены операций

Перед началом выполнения транзакции ей присваивается сегмент отмены операций. На протяжении всего жизненного цикла транзакции, перед изменением данных их исходные значения копируются в сегмент отмены операций. Узнать о том, каким транзакциям назначены какие сегменты отмены операций, можно с помощью динамического представления производительности `V$TRANSACTION`.

Сегменты отмены операций являются специализированными сегментами, которые автоматически создаются экземпляром по мере необходимости для поддержки транзакций. Подобно остальным сегментам, сегменты отмены операций состоят из экстенгов, которые, в свою очередь, состоят из блоков данных. Размер сегментов отмены операций изменяется автоматически согласно текущим потребностям. Таким образом, сегменты отмены операций выступают циклическим буфером хранения для записанных им транзакций.

Транзакции будут заполнять экстенги сегментов отмены операций до своего завершения или пока не будет заполнено все доступное пространство. Если экстенг заполнен и при этом требуется дополнительное пространство, транзакция попытается использовать следующий экстенг сегмента. После заполнения всех экстенгов транзакция или вернется к первому сегменту и начнет его повторно заполнять, или создаст запрос на выделение нового экстенга в сегменте отмены операций.

Примечание. Параллельные операции DML и DDL могут привести к тому, что транзакция будет использовать более одного сегмента отмены операций. Дополнительные сведения о параллельном выполнении DML представлены в руководстве *Oracle Database Administrator's Guide* (Руководство администратора базы данных Oracle).

Хранение данных отмены операций

Данные отмены операций хранятся в сегментах отмены операций, которые, в свою очередь, хранятся в табличном пространстве отмены операций. Табличные пространства отмены операций:

- используются только для сегментов отмены операций
- имеют особые правила восстановления
- могут быть связаны только с одним экземпляром
- требуют, чтобы для каждого экземпляра в каждый момент времени было доступным для записи только одно из табличных пространств отмены операций

ORACLE

© Oracle, 2009. Все права защищены.

Хранение данных отмены операций

Сегменты отмены операций могут существовать только в особых табличных пространствах, которые называются *табличными пространствами отмены операций*. (В табличном сегменте отмены операций невозможно создать сегмент другого типа, например таблицу.)

DBCA автоматически создает файл SMALLFILE для табличного пространства отмены операций. Также для табличного пространства отмены операций можно создать BIGFILE. Однако в многотомной среде оперативной обработки транзакций (OLTP) с множеством коротких параллельных транзакций может возникнуть конкуренция за доступ к заголовку файла. Табличное пространство отмены операций, хранимое в нескольких файлах данных, может устранить данную проблему.

Несмотря на то, что база данных может хранить много табличных пространств отмены операций, только одно из них может быть назначено текущим табличным пространством отмены операций для экземпляра базы данных.

Сегменты отмены операций создаются автоматически и всегда принадлежат SYS. Так как сегменты отмены операций действуют как циклический буфер, каждый сегмент состоит как минимум из двух экстенгов. Максимальное количество экстенгов по умолчанию зависит от размера блока базы данных, но при этом оно достаточно велико (32765 для размера блока 8 Кбайт).

Табличные пространства отмены операций являются постоянными управляемыми локально табличными пространствами с автоматическим выделением экстенгов. Они автоматически управляются базой данных.

Так как данные отмены операций необходимы для восстановления данных после сбойных транзакций (например, возникших вследствие сбоя экземпляра), табличные пространства отмены операций могут быть восстановлены, только когда экземпляр работает в режиме MOUNT. Рекомендации по восстановлению табличных пространств отмены операций описаны в уроке «Выполнение восстановления базы данных».

Данные отмены операций и данные повторов

	Отмена	Повтор
Запись	Как отменить изменение	Как воссоздать изменение
Цели использования	Откат, целостность при чтении, моментальный откат	Повторное применение изменений базы данных
Место хранения	Сегменты отмены операций	Файлы журнала повторов
Защита от	Несогласованные операции чтения в многопользовательских системах	Потеря данных



ORACLE

© Oracle, 2009. Все права защищены.

Данные отмены операций и данные повторов

Данные отмены операций и данные повторов на первый взгляд похожи, однако они используются для разных целей. Данные отмены операций нужны для того, чтобы иметь возможность отменить внесенные изменения (обычно это требуется для сохранения целостности данных при чтении, а также в операциях отката). Данные повторов нужны для того, чтобы иметь возможность повторно применить изменения, если они будут утеряны по каким-либо причинам. Изменения в блоках данных отмены операций также сохраняются в журнале повторов.

Процесс фиксации сохраняет внесенные транзакцией изменения в файл журнала повторов, который является постоянным файлом на диске, в отличие от данных, хранимых в памяти. Кроме того, файл журнала повторов обычно мультиплексируется. Таким образом, на диске может храниться множество копий данных повторов. Так как изменения могут быть еще не записаны в файлы данных, в которых хранятся блоки таблицы, запись в постоянный журнал повторов является достаточной гарантией сохранения целостности базы данных.

Представим, что произошло отключение питания до того, как подтвержденные изменения были сохранены в файлах данных. Так как транзакция была подтверждена, данные не будут утеряны. При следующем запуске системы можно будет осуществить накат записей повторов для сохранения изменений, существовавших на момент отключения питания, в файлах данных.

Управление данными отмены операций

Автоматическое управление данными отмены операций:

- полностью автоматизированное управление данными отмены операций в выделенном табличном пространстве отмены операций
- осуществляется для всех сеансов
- самонастройка в табличных пространствах AUTOEXTEND для обработки запросов с длительным выполнением
- самонастройка в табличных пространствах с фиксированным размером для определения наилучшего времени хранения

Задачи администратора при выполнении операций моментальной отмены:

- настройка времени хранения данных отмены операций
- приведение табличного пространства отмены операций к типу с фиксированным размером
- недопущение ошибок пространства и ошибок типа «слишком старый снимок»

ORACLE

© Oracle, 2009. Все права защищены.

Управление данными отмены операций

База данных Oracle осуществляет *автоматическое управление данными отмены операций*. Для управления данными отмены и пространством в выделенном для всех сеансов табличном пространстве отмены операций используется полностью автоматизированный механизм. Система настраивается автоматически для определения наилучшего времени хранения данных отмены операций. Период хранения данных отмены операций для автоматически расширяющихся табличных пространств настраивается таким образом, чтобы он был немного большим, чем дольше всех выполняемый активный запрос. Для табличного пространства отмены операций фиксированного размера база данных динамически настраивает наиболее оптимальный период хранения данных отмены.

Автоматическое управление данными отмены операций по умолчанию включено в Oracle Database 11g (и более поздних версиях). Ручное управление данными отмены операций поддерживается для обеспечения обратной совместимости с Oracle8i и более ранними версиями, однако для него требуется вмешательство администратора. В ручном режиме управления данными отмены операций управление пространством осуществляется через сегменты откатов (а не табличное пространство отмены операций).

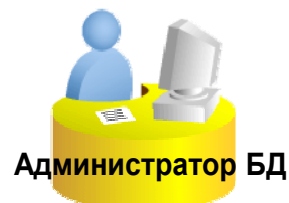
Примечание. Oracle настоятельно рекомендует использовать автоматическое управление данными отмены операций.

Несмотря на то, что база данных Oracle автоматически управляет пространством и данными отмены операций, в некоторых случаях может возникнуть необходимость в выполнении определенных задач, если в базе данных используются операции моментальной отмены. Администрирование данных отмены операций должно предотвратить ошибки, связанные с чрезмерным потреблением пространства, а также ошибки типа «слишком старый снимок».

Настройка времени хранения данных отмены операций

UNDO_RETENTION позволяет указать (в секундах) период хранения зафиксированных данных отмены операций. Этот параметр должен настраиваться только в следующих ситуациях:

- если для табличного пространства отмены операций включен параметр AUTOEXTEND
- если необходимо установить время хранения объектов типа LOB
- **если требуется гарантировать сохранение данных**



ORACLE

© Oracle, 2009. Все права защищены.

Настройка времени хранения данных отмены операций

Параметр инициализации UNDO_RETENTION позволяет указать (в секундах) нижнее пороговое значение периода хранения данных отмены операций. Минимальный период хранения данных отмены операций для автоматически расширяющихся табличных пространств отмены операций не должен быть меньше ожидаемого времени выполнения самой большой операции моментальной отмены. Для автоматически расширяющихся табличных пространств отмены операций система сохраняет данные отмены как минимум в течение указанного параметром интервала времени, а также автоматически настраивает период хранения данных отмены операций согласно потребностям запросов. Однако в некоторых случаях длительности автоматически настроенного периода времени может быть недостаточно для выполнения операций моментальной отмены.

Для табличных пространств отмены с фиксированным размером система автоматически настраивает наиболее эффективный период хранения данных отмены операций, основываясь на размере табличного пространства отмены и его статистических данных. Значение параметра UNDO_RETENTION игнорируется, если не задействована функция гарантированного сохранения. Для режима автоматического управления данными отмены операций параметр UNDO_RETENTION используется в трех случаях, которые представлены на слайде.

Во всех других случаях этот параметр будет проигнорирован.

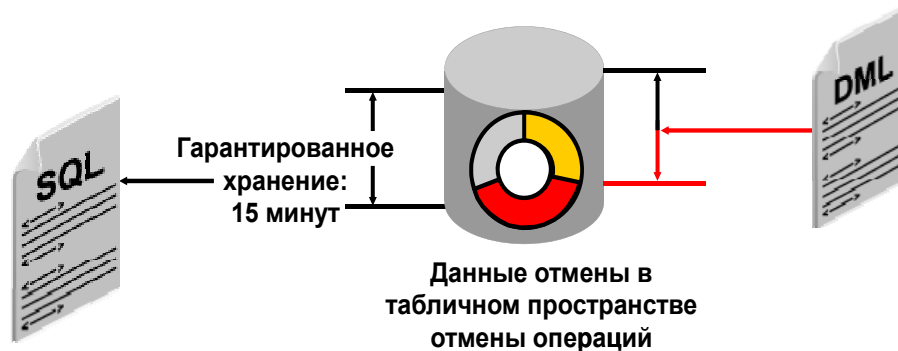
Настройка времени хранения данных отмены операций (продолжение)

Данные отмены операций делятся на три категории:

- **Незафиксированные данные отмены (активные):** обслуживают текущую транзакцию; необходимы для отката в случае сбоя транзакции. Незафиксированные данные отмены никогда не перезаписываются.
- **Зафиксированные данные отмены (неустаревшие):** не нужны для обслуживания текущей транзакции, однако хранятся определенный период времени как данные отмены операций. Также известны как «неустаревшие» данные отмены. Зафиксированные данные отмены операций сохраняются (когда это возможно) и не приводят к сбою транзакции из-за нехватки пространства.
- **Устаревшие данные отмены (устаревшие):** больше не требуются для обслуживания текущей транзакции. Устаревшие данные отмены операций перезаписываются тогда, когда активной транзакции требуется дополнительное пространство.

Гарантированное хранение данных отмены операций

```
SQL> ALTER TABLESPACE undotbs1 RETENTION GUARANTEE;
```



Операторы SELECT, которые обрабатываются не более 15 минут, всегда завершаются успешно.

Транзакция будет признана **сбойной**, если она создаст больше данных отмены, чем имеется пространства.

Примечание. В этом примере для оператора UNDO_RETENTION установлено значение 900 секунд (15 минут).

ORACLE

© Oracle, 2009. Все права защищены.

Гарантированное хранение данных отмены операций

По умолчанию данные отмены операций для подтвержденных и не устаревших транзакций будут перезаписаны в случае угрозы сбоя активной транзакции из-за нехватки пространства отмены операций.

Такое поведение можно изменить с помощью режима гарантированного сохранения. В этом режиме данные отмены операций сохраняются принудительно даже в том случае, если это может привести к сбою транзакции.

RETENTION GUARANTEE — атрибут табличного пространства, а не параметр инициализации. Атрибут можно изменить только с помощью SQL-операторов командной строки. Синтаксис изменения табличного пространства отмены операций для **использования гарантированного сохранения**:

```
SQL> ALTER TABLESPACE undotbs1 RETENTION GUARANTEE;
```

Чтобы вернуть табличное пространство отмены операций в обычный режим, воспользуйтесь командой:

```
SQL> ALTER TABLESPACE undotbs1 RETENTION NOGUARANTEE;
```

Понятие гарантированного сохранения применимо только к табличным пространствам отмены операций. Попытка перевести в такой режим табличное пространство другого типа приведет к ошибке:

```
SQL> ALTER TABLESPACE example RETENTION GUARANTEE;
```

```
ERROR at line 1:
```

```
ORA-30044: 'Retention' can only specified for undo tablespace
```

Приведение табличного пространства отмены операций к фиксированному размеру

Причины:

- поддержка операций моментальной отмены
- ограничение роста табличного пространства



Поток операций:

1. Поддерживайте рабочую нагрузку на обычном уровне.
2. Самонастраивающийся механизм обеспечивает минимально необходимый размер.
3. (Необязательно) Используйте помощник Undo Advisor, который вычисляет необходимый размер для будущего роста.
4. (Необязательно) Приведите табличное пространство отмены операций к фиксированному размеру.

ORACLE

© Oracle, 2009. Все права защищены.

Приведение табличного пространства отмены операций к фиксированному размеру

Существуют две причины приведения табличного пространства отмены операций к фиксированному размеру: поддержка операций моментального отката (когда планируется в дальнейшем использовать данные отмены операций) и предотвращение чрезмерного увеличения размера табличного пространства.

Если вы решите привести табличное пространство отмены операций к фиксированному размеру, установите достаточно большой размер, чтобы избежать следующих ошибок:

- сбой DML (недостаток пространства для отмены изменений для новых транзакций)
- ошибки типа «слишком старый снимок» (недостаток данных отмены операций для сохранения целостности данных для чтения)

Oracle рекомендует поддерживать постоянную рабочую нагрузку, что позволит табличному пространству отмены увеличиться до минимального необходимого размера. Автоматически собираемая статистика содержит данные о продолжительности выполнения самого длительного запроса и об объеме созданных данных отмены операций. Вычисление минимального размера для табличного пространства отмены операций на основе такой статистики рекомендуется для систем, в которых не применяются операции моментальной отмены, а также для систем, в которых не планируется в будущем использовать запросы с большим временем выполнения.

Можно воспользоваться помощником Undo Advisor для ввода желаемой длительности периода хранения данных отмены для операций моментального отката и запросов с большим временем выполнения.

Общие сведения о данных отмены операций

Database Instance: orcl.oracle.com >

Automatic Undo Management

In the General tab, you can view the current undo settings for your instance and use the Undo Advisor to analyze the undo tablespace requirements. This analysis can be performed based on the specified analysis period or the desired undo retention. The system activity for the specified time period can be viewed in the System Activity tab.

General System Activity

Undo Retention Settings		Undo Tablespace for this Instance	
Undo Retention (minutes)	15	Tablespace	UNDOTBS1 Change Tablespace
Retention Guarantee	No	Size (MB)	100
		Auto-Extensible	Yes

Текущий размер
табличного
пространства

ORACLE

© Oracle, 2009. Все права защищены.

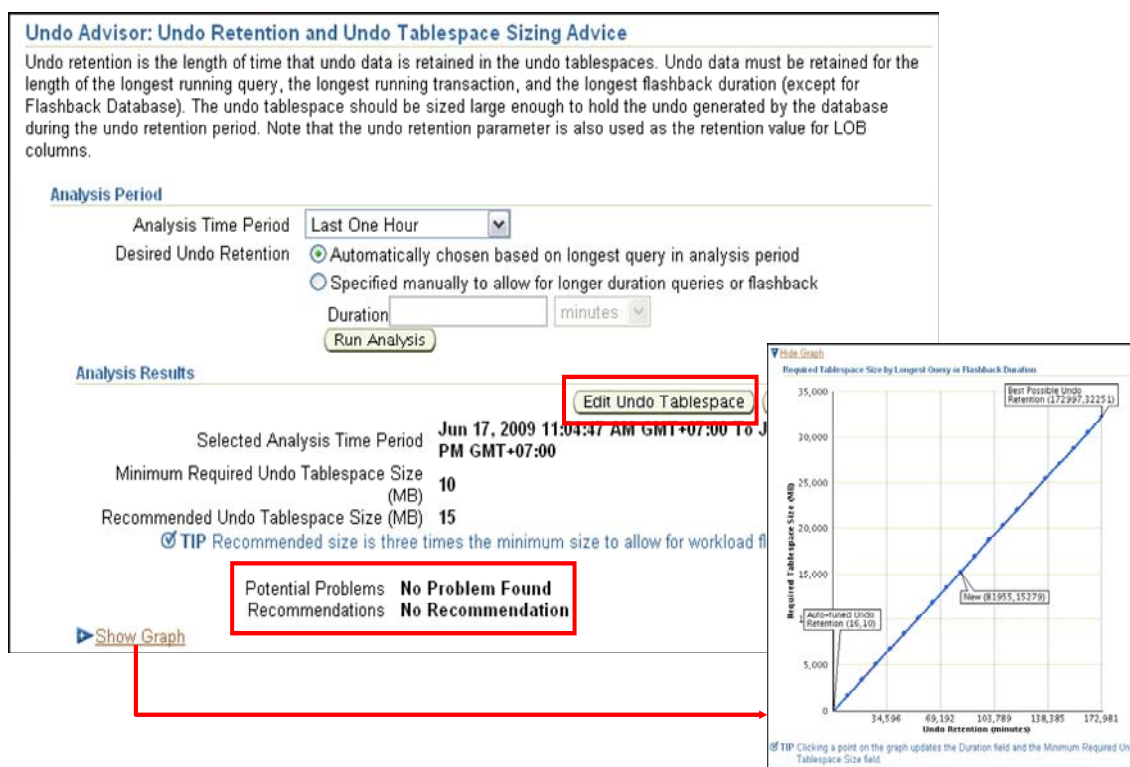
Общие сведения о данных отмены операций

В Enterprise Manager выберите «Server» (Сервер) > «Automatic Undo Management» (Автоматическое управление данными отмены операций).

Существуют две страницы: «General» (Общие) и «System Activity» (Системная активность).

В верхней части общей страницы отображаются параметры хранения данных отмены операций и информация о табличном пространстве отмены для данного экземпляра.

Использование помощника Undo Advisor



ORACLE

© Oracle, 2009. Все права защищены.

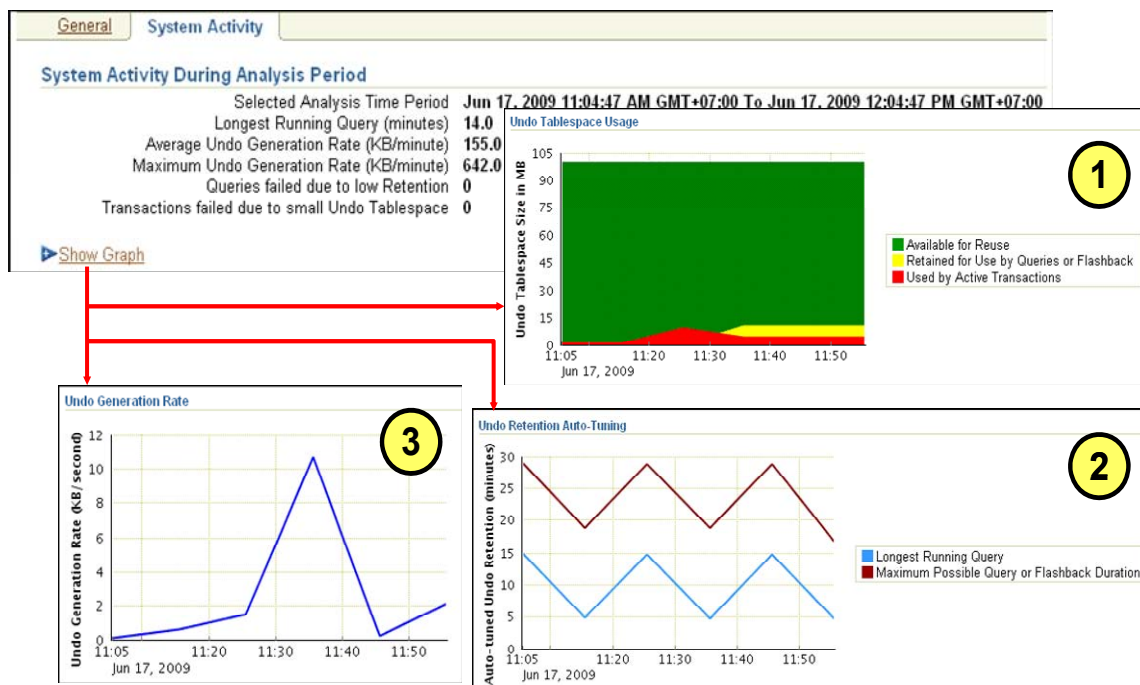
Использование помощника Undo Advisor

Средняя часть страницы «General Undo» (Общая отмена операций) предоставляет доступ к помощнику Undo Advisor. Он вычисляет размер табличного пространства отмены, который необходим для удовлетворения условиям гарантированного хранения данных отмены операций.

В аналитической области помощника отображается требуемый размер для поддержки периода гарантированного хранения данных. Можно выбрать точку на графике, чтобы увидеть размер табличного пространства, который необходим для поддержки выбранного периода.

Нажмите кнопку «Edit Undo Tablespace» (Редактировать табличное пространство отмены), а затем «Edit» (Редактировать) в области «Datafile» (Файл данных), чтобы привести табличное пространство отмены операций к фиксированному размеру.

Просмотр системной активности



ORACLE

© Oracle, 2009. Все права защищены.

Просмотр системной активности

В верхней части страницы отображается активность системы в указанный период времени.

Ниже представлены три графика:

1. **Использование табличного пространства отмены операций:** отображает размер табличного пространства (в Мбайт) относительно дней месяца.
2. **Автонастройка периода хранения данных отмены операций:** отображает настроенное время хранения данных отмены (в минутах) относительно дней месяца.
3. **Скорость создания данных отмены операций:** отображает скорость создания данных отмены операций (в Кбайт за секунду) относительно дней месяца.

Экспресс-тест

Чтобы обеспечить нахождение данных отмены, необходимых для сохранения целостности данных для чтения, всеми запросами, которые обрабатываются менее 15 минут, достаточно установить для параметра `UNDO_RETENTION` значение «15 минут».

1. Да
2. Нет

The Oracle logo is displayed in white text on a red rectangular background.

© Oracle, 2009. Все права защищены.

Ответ: 2

Экспресс-тест

Какое утверждение не относится к данным отмены?

1. Обеспечивают запись процедуры отмены изменения
2. Используются для отката, сохранения целостности при чтении и моментального отката
3. Хранятся только в памяти, не записываются на диск
4. Обеспечивают защиту от несогласованных операций чтения в многопользовательских системах

ORACLE

© Oracle, 2009. Все права защищены.

Ответ: 3

Заключение

В ходе прохождения урока вы научились:

- объяснять DML и происхождение данных отмены операций;
- **отслеживать и администрировать данные отмены операций;**
- описывать различия между данными отмены операций и данными повторов;
- настраивать время хранения данных отмены операций;
- гарантировать актуальность данных отмены операций в течение заданного времени;
- использовать помощник Undo Advisor.

ORACLE

© Oracle, 2009. Все права защищены.

Обзор упражнения 10: управление сегментами отмены операций

В данном упражнении рассматриваются следующие темы:

- просмотр системной активности;
- расчет размера табличного пространства отмены **для обеспечения 48-часового интервала хранения;**
- изменение табличного пространства отмены для обеспечения 48-часового интервала хранения.

ORACLE

© Oracle, 2009. Все права защищены.

11

Аудит базы данных Oracle

ORACLE

© Oracle, 2009. Все права защищены.

Цели

По завершении данного урока вы сможете:

- описывать обязанности АБД по обеспечению безопасности и проведению аудита;
- **включать стандартный аудит БД;**
- задавать параметры аудита;
- просматривать информацию об аудите;
- обслуживать журнал аудита.



ORACLE

© Oracle, 2009. Все права защищены.

Цели

Это первый урок о безопасности Oracle. Дополнительная информация по теме представлена в следующих документах:

- *Основные понятия базы данных Oracle*
- *Руководство администратора базы данных Oracle*
- *Руководство по безопасности базы данных Oracle*

Дополнительные занятия по теме представлены в следующих курсах:

- *База данных Oracle 11g: семинар по администрированию II*
- *База данных Oracle 11g: безопасность*

Разделение ответственности

- Необходимо обеспечить надежность пользователей с полномочиями администратора БД.
 - Злоупотребление доверием
 - **Защита доверенной должности с помощью журнала аудита**
- Ответственность администратора БД следует распределить.
- Формуляры не должны использоваться совместно.
- **Должности администратора БД и системного администратора совмещать нельзя.**
- Необходимо разделять зону ответственности оператора и администратора БД.

ORACLE

© Oracle, 2009. Все права защищены.

Разделение ответственности

Ниже приведены основные требования к разделению обязанностей.

Администраторы БД должны быть надежными: полномочия администратора БД ограничить сложно. Для выполнения обязанностей администратору требуются полномочия высокого уровня. Поскольку администратор БД — очень ответственная должность, претенденты должны пройти тщательный отбор. Даже если администратор надежен, необходимы отчеты по его действиям. Обдумайте следующие вопросы:

- **Злоупотребление доверием:** потенциально администратор БД имеет возможность несанкционированно использовать зашифрованные пароли из представления DBA_USERS.
- **Защита доверенной должности с помощью журнала аудита:** при тщательной реализации аудита и соблюдении руководящих принципов по журналу аудита можно выяснить, не нарушались ли процедуры и не был ли нанесен ущерб действиями какого-либо лица. Если злоумышленник попытается навести подозрение на надежного пользователя, хорошо спроектированные журналы аудита позволят предотвратить это.

Oracle Database Vault: можно использовать компонент Oracle Database Vault для ситуаций, когда разделение обязанностей приходится обеспечивать средствами базы данных или когда администратору БД недоступен просмотр данных в некоторых или во всех схемах базы данных.

Безопасность базы данных

Безопасная система обеспечивает конфиденциальность хранящихся в ней данных. Существует несколько аспектов безопасности:

- **ограничение доступа к данным и службам**
- аутентификация пользователей
- мониторинг подозрительных операций



ORACLE

© Oracle, 2009. Все права защищены.

Безопасность базы данных

Oracle Database 11g предлагает лучшую в отрасли структуру защищенной системы. Однако для того, чтобы эта структура была эффективной, администратор БД должен следовать установившейся практике и осуществлять непрерывный мониторинг активности базы данных.

Ограничение доступа к данным и службам

Всем пользователям не могут быть доступны все данные. В зависимости от содержимого базы данных, ограничение доступа обусловлено производственными требованиями, пожеланиями клиентов и (все в большей мере) правовыми ограничениями. Данные кредитных карт, здравоохранительные данные и т. п. требуют защиты от несанкционированного доступа. БД Oracle предоставляет чрезвычайно детализированные средства авторизации для ограничения доступа к базе данных. При ограничении доступа следует придерживаться принципа меньших полномочий.

Безопасность базы данных (продолжение)

Аутентификация пользователей

Для ужесточения контроля доступа к уязвимым данным системе необходимо предварительно опознать пользователя, пытающегося получить доступ к данным. При взломе аутентификации все остальные меры безопасности могут оказаться бессильны. Самым простым видом аутентификации пользователей является запрос данных, известных только пользователю, например, пароля. А если пароли соответствуют некоторым простым правилам, безопасность системы значительно повысится. При более строгих методах аутентификации пользователь должен предъявить нечто свое, например, маркер или сертификат инфраструктуры открытого ключа (PKI). При еще более строгой аутентификации идентификация пользователя требует ввода биометрических данных, например, отпечатков пальцев, изображения радужной оболочки глаза, структуры костей и т. п. В БД Oracle поддерживаются усложненные методы аутентификации (такие как идентификация по маркерам, биометрическим данным и сертификатам). Для этого служат функции дополнительной защиты. Неиспользуемые формуляры пользователей должны блокироваться во избежание несанкционированной аутентификации.

Мониторинг подозрительных действий

Иногда даже прошедшие авторизацию и аутентификацию пользователи могут выполнять несанкционированные действия в системе. Обнаружение необычных операций с базой данных (например, сотрудник неожиданно выполняет массовые запросы информации о кредитных картах, результатах исследований или другой уязвимой информации) может послужить первым шагом к выявлению хищения информации. БД Oracle предлагает богатый выбор средств аудита, позволяющих следить за действиями пользователей и выявлять подозрительные тенденции.

Мониторинг совместимости

Мониторинг или аудит должен быть неотъемлемой составляющей процедур обеспечения безопасности.

Следует рассмотреть следующие вопросы:

- **обязательный аудит;**
- стандартный аудит БД;
- аудит на основе значений;
- детализированный аудит (FGA);
- аудит SYSDBA (и SYSOPER).



ORACLE

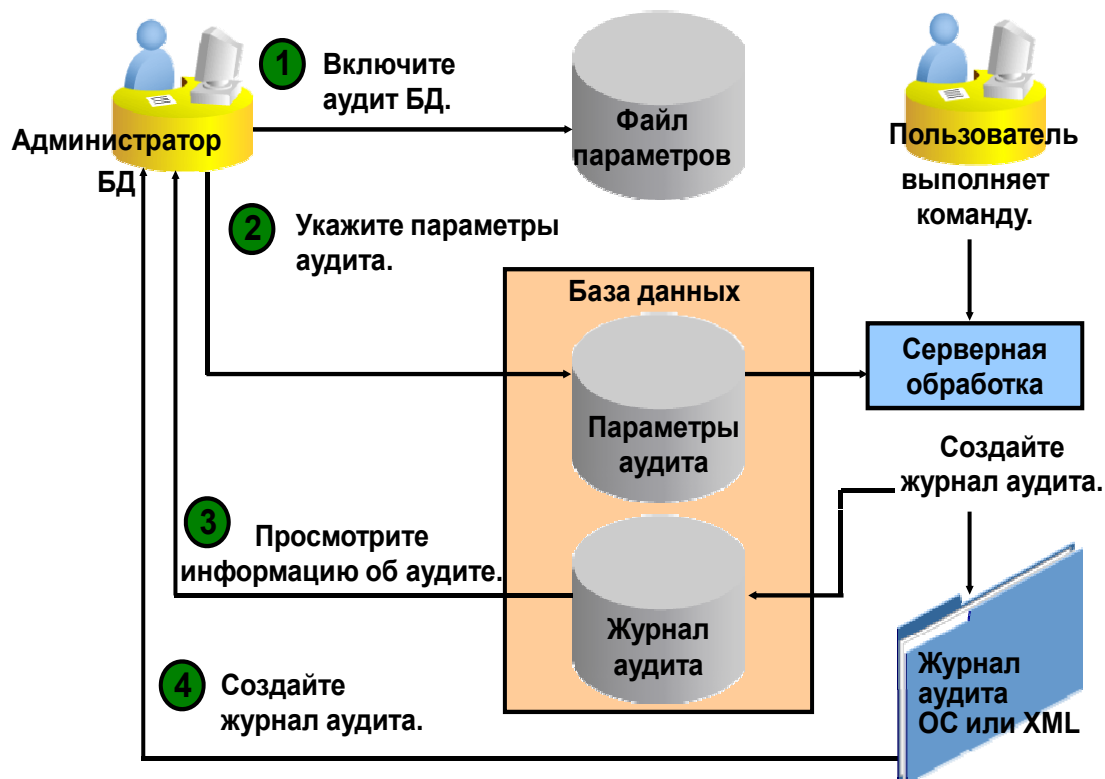
© Oracle, 2009. Все права защищены.

Мониторинг совместимости

Аудит, подразумевающий регистрацию и хранение информации об операциях в системе, повышает загрузку системы. Поэтому его следует сосредоточить только на событиях, представляющих интерес. Правильно сфокусированный аудит минимально влияет на производительность системы. А неправильно сфокусированный может значительно понизить производительность.

- **Обязательный аудит:** во всех БД Oracle выполняется аудит определенных операций вне зависимости от настроек или параметров аудита. Обязательные журналы аудита необходимы для записи некоторых операций в базе данных, например подключений пользователей с полномочиями.
- **Стандартный аудит БД:** включается на системном уровне с помощью параметра `AUDIT_TRAIL`. После включения аудита с помощью команды `AUDIT` можно выбрать объекты и полномочия для аудита, а также свойства аудита.
- **Аудит на основе значений:** является расширением стандартного аудита БД, при котором проверяется не только событие, подлежащее аудиту, но также и фактические значения, которые были добавлены, обновлены или удалены. Данный вид аудита реализуется посредством триггеров БД.
- **Детализированный аудит (FGA):** является расширением стандартного аудита БД, при котором проверяется также выполненный SQL-оператор, а не только факт того, что событие произошло.
- **Аудит SYSDBA (и SYSOPER):** позволяет разделить обязанности по аудиту между администратором БД и аудитором или администратором службы безопасности, который следит за действиями администратора БД по записям в журнале аудита ОС.

Стандартный аудит БД



ORACLE

© Oracle, 2009. Все права защищены.

Стандартный аудит БД

Чтобы использовать аудит БД, предварительно необходимо присвоить статическому параметру инициализации `AUDIT_TRAIL` значение пути к каталогу для хранения записей аудита. После этого аудит БД будет включен. После включения и настройки параметров аудита БД (события входа в систему, использование системных и объектных полномочий или SQL-операторов) в базе данных начинается сбор информации для аудита.

Если параметр `AUDIT_TRAIL` имеет значение `OS`, записи аудита сохраняются в системе аудита ОС. В среде Windows для этого используется журнал событий. В среде UNIX или Linux записи аудита хранятся в файле, определяемом значением параметра `AUDIT_FILE_DEST`.

Если параметру `AUDIT_TRAIL` присвоено значение `DB` или `DB, EXTENDED`, то записи аудита можно просмотреть в представлении `DBA_AUDIT_TRAIL`, входящем в схему `SYS`.

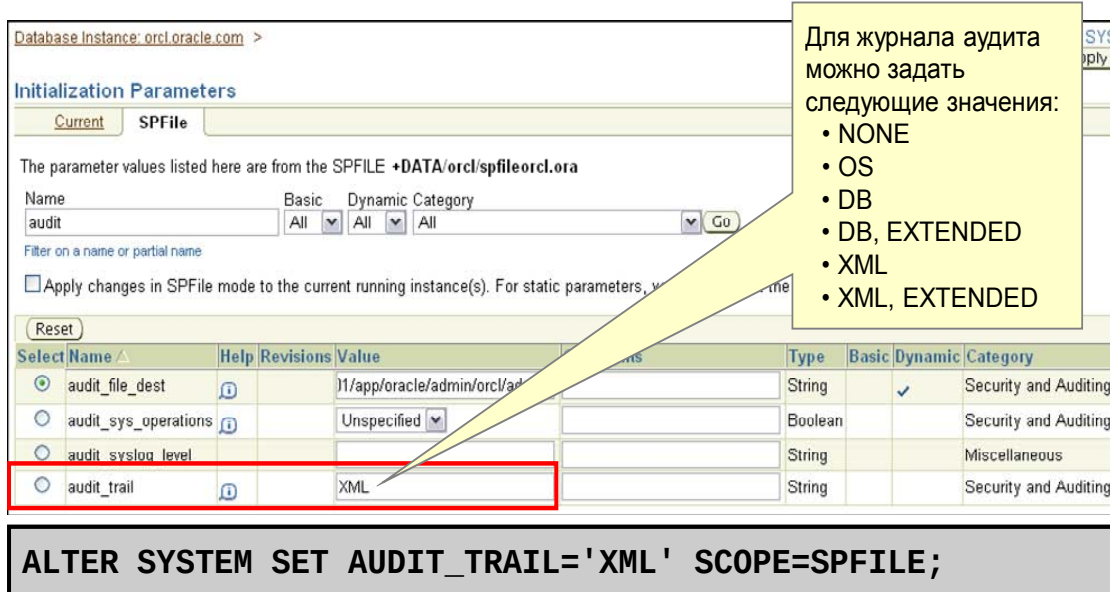
Если параметр `AUDIT_TRAIL` имеет значение `XML` или `XML, EXTENDED`, то записи аудита сохраняются в XML-файлах в каталоге, на который указывает параметр `AUDIT_FILE_DEST`. С помощью представления `V$XML_AUDIT_TRAIL` можно просмотреть все XML-файлы в данном каталоге.

Обслуживание журнала аудита является важнейшей задачей администратора.

В зависимости от областей, затронутых с помощью параметров аудита, журнал аудита может разрастись до значительных размеров. При неправильном обслуживании журнал аудита может включать так много записей, что это снизит производительность системы. Избыточность аудита напрямую связана с количеством создаваемых записей.

Настройка журнала аудита

Используйте AUDIT_TRAIL для включения аудита БД.



Для журнала аудита можно задать следующие значения:

- NONE
- OS
- DB
- DB, EXTENDED
- XML
- XML, EXTENDED

Select	Name	Help	Revisions	Value	Comments	Type	Basic	Dynamic	Category
☒	audit_file_dest			D1/app/oracle/admin/orcl/ad		String	✓		Security and Auditing
☐	audit_sys_operations			Unspecified		Boolean			Security and Auditing
☐	audit_syslog_level					String			Miscellaneous
☐	audit_trail			XML		String			Security and Auditing

```
ALTER SYSTEM SET AUDIT_TRAIL='XML' SCOPE=SPFILE;
```

После изменения этого статического параметра инициализации БД необходимо перезапустить.

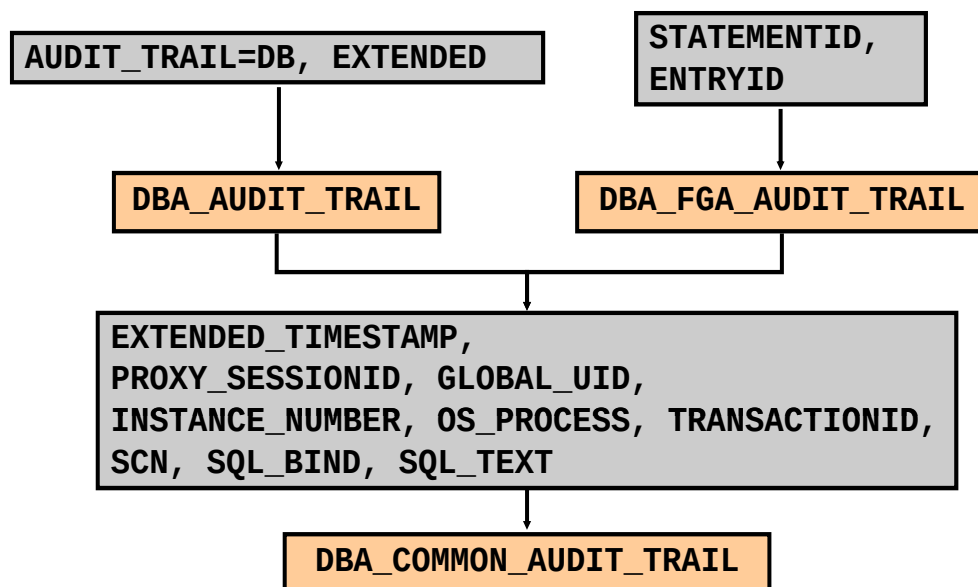
ORACLE

© Oracle, 2009. Все права защищены.

Настройка журнала аудита

Можно задать параметр AUDIT_TRAIL с помощью Enterprise Manager (на странице «Initialization Parameters» (Параметры инициализации)) или с помощью команды ALTER SYSTEM SET в SQL*Plus. Поскольку это статический параметр, необходимо перезапустить базу данных, чтобы изменение вступило в силу. Если вы создали базу данных с использованием помощника Database Configuration Assistant (DBCA), параметру audit_trail по умолчанию присваивается значение DB. Если параметр AUDIT_TRAIL имеет значение DB, журнал аудита по умолчанию записывается в таблицу AUD\$ базы данных. В большинстве отделений эти операции аудита не оказывают значительного влияния на производительность базы данных. Oracle рекомендует использовать файлы журнала аудита OS. Если база данных создана вручную (с помощью команды CREATE DATABASE), для параметра AUDIT_TRAIL по умолчанию устанавливается значение NONE.

Унифицированные журналы аудита



ORACLE

© Oracle, 2009. Все права защищены.

Унифицированные журналы аудита

В БД Oracle при стандартном и детализированном аудите отслеживаются одни и те же поля, что упрощает анализ операций в базе данных. Для этого стандартный и детальный журналы аудита обладают взаимодополняющими атрибутами.

При стандартном аудите собирается следующая дополнительная информация:

- системный номер изменения (SCN), регистрирующий любое изменение в системе;
- точный текст SQL, выполненный пользователем, а также связанные переменные, использованные в данном тексте SQL. Эти столбцы отображаются, только если в файле параметров инициализации указана строка **AUDIT_TRAIL=DB, EXTENDED**.

При детализированном аудите собирается следующая дополнительная информация:

- порядковый номер каждой записи аудита;
- номер оператора, связывающий несколько записей аудита, порожденных одним оператором.

Общие атрибуты:

- глобальная отметка времени в координатах UTC. Это поле удобно для слежения за серверами, находящимися в разных географических областях и часовых поясах.
- уникальный номер каждого экземпляра Real Application Clusters (RAC);
- идентификатор транзакции, помогающий группировать записи аудита для отдельной транзакции.

В представлении **DBA_COMMON_AUDIT_TRAIL** объединены записи стандартного и детализированного журналов аудита.

Определение параметров аудита

- Аудит SQL-операторов:

```
AUDIT table;
```

- Аудит системных полномочий (нефокусный и фокусный):

```
AUDIT select any table, create any trigger;  
AUDIT select any table BY hr BY SESSION;
```

- Аудит объектных полномочий (нефокусный и фокусный):

```
AUDIT ALL on hr.employees;  
AUDIT UPDATE,DELETE on hr.employees BY ACCESS;
```

ORACLE

© Oracle, 2009. Все права защищены.

Определение параметров аудита

Аудит SQL-операторов: приведенный на слайде оператор позволяет выполнить аудит любого оператора языка определения данных (DDL), который влияет на таблицу, включая CREATE TABLE, DROP TABLE, TRUNCATE TABLE и т. д. Аудит SQL-операторов можно фокусировать на имени пользователя, на успешном выполнении или сбое:

```
SQL> AUDIT TABLE BY hr WHENEVER NOT SUCCESSFUL;
```

Аудит системных полномочий: может использоваться для аудита использования любых системных полномочий (например, DROP ANY TABLE). Его можно фокусировать на имени пользователя, на успешном выполнении или сбое. По умолчанию используется параметр аудита BY ACCESS. При каждом использовании системных полномочий, подверженных аудиту, происходит создание записи аудита. Можно группировать эти записи с помощью фразы BY SESSION, тогда для каждого сеанса будет создаваться одна запись. (В этом случае, если пользователь выполняет несколько операторов UPDATE в таблице другого пользователя, будет получена только одна запись аудита.) Рекомендуется использовать фразу BY SESSION, чтобы ограничить воздействие аудита системных полномочий на производительность и объем свободной памяти.

Аудит объектных полномочий: может использоваться для аудита операций с таблицами, представлениями, процедурами, последовательностями, каталогами и пользовательскими типами данных. Данный тип аудита можно фокусировать на успешном выполнении или сбое, а также группировать по сеансу или обращению. В отличие от аудита системных полномочий, по умолчанию используется группирование по сеансу. Если необходимо создавать отдельные записи аудита для каждой операции, следует явным образом указать параметр BY ACCESS.

Аудит по умолчанию

Полномочия, подвергнутые аудиту по умолчанию		
ALTER ANY PROCEDURE	CREATE ANY LIBRARY	GRANT ANY PRIVILEGE
ALTER ANY TABLE	CREATE ANY PROCEDURE	GRANT ANY ROLE
ALTER DATABASE	CREATE ANY TABLE	DROP ANY PROCEDURE
ALTER PROFILE	CREATE EXTERNAL JOB	DROP ANY TABLE
ALTER SYSTEM	CREATE PUBLIC DATABASE LINK	DROP PROFILE
ALTER USER	CREATE SESSION	DROP USER
AUDIT SYSTEM	CREATE USER	EXEMPT ACCESS POLICY
CREATE ANY JOB	GRANT ANY OBJECT PRIVILEGE	
Инструкции, подвергнутые аудиту по умолчанию		
SYSTEM AUDIT BY ACCESS		
ROLE BY ACCESS		

ORACLE

© Oracle, 2009. Все права защищены.

Аудит по умолчанию

Если в базе данных Oracle Database 11g включен аудит, некоторые полномочия и операторы, имеющие особое значение с точки зрения безопасности, подвергаются аудиту по умолчанию. Эти полномочия и операторы перечислены на слайде, и их аудит выполняется для всех пользователей при успешном выполнении операции, при сбое и при получении доступа.

Страница аудита в Enterprise Manager

Security

- [Users](#)
- [Roles](#)
- [Profiles](#)
- [Audit Settings](#)**
- [Transparent Data Encryption](#)

Audit Settings

Audit information can be located in the database or in an OS file. Some information is always written to the OS audit file. Other information can optionally be written to either the OS audit file or to the database.

Configuration

Audit Trail: [DB](#)

Audit SYS User Operations: [FALSE](#)

Audit File Directory: [/u01/app/oracle/admin/orcl/adump](#)
Audit File Directory value is effective only when Audit Trail is set to "OS" or "XML".

Default Options For Future Audited Objects: [0](#)

Audit Trails

Database Audit Trail

[Audited Failed Logins](#)

[Audited Privileges](#)

[Audited Objects](#)

Operating System Audit Trail

[View OS Audit Trails](#)

Audited Privileges (23)

[Audited Objects \(0\)](#)

[Audited Statements \(2\)](#)

Select	Privilege	User	Proxy	Success	Failure
☐	DROP PROFILE			BY ACCESS	BY ACCESS
☐	ALTER ANY TABLE			BY ACCESS	BY ACCESS
☐	ALTER SYSTEM			BY ACCESS	BY ACCESS
☐	ALTER DATABASE			BY ACCESS	BY ACCESS
☐	DROP USER			BY ACCESS	BY ACCESS

ORACLE

© Oracle, 2009. Все права защищены.

Страница аудита в Enterprise Manager

Чтобы открыть страницу аудита, на домашней странице Database Control следует щелкнуть вкладку «Server» (Сервер) и в области «Security» (Безопасность) щелкнуть ссылку «Audit Settings» (Настройки аудита).

Страница аудита содержит следующие области:

- **Конфигурация:** отображает текущие значения параметров конфигурации и ссылки для их изменения.
- **Журналы аудита:** обеспечивает простой доступ к собранной информации аудита.

С помощью этих вкладок можно настраивать параметры аудита:

- **Подвергнутые аудиту полномочия:** отображает полномочия, подвергнутые аудиту.
- **Подвергнутые аудиту объекты:** отображает объекты, подвергнутые аудиту.
- **Подвергнутые аудиту операторы:** отображает операторы, подвергнутые аудиту.

Использование и обслуживание информации аудита

Schema	Object Name	User Name	Action	Time
INVENTORY	PRODUCT_MASTER	DBA1	ALTER TABLE	2008-08-13 22:47:56.0
INVENTORY	PRODUCT_ON_HAND	DBA1	CREATE TABLE	2008-08-13 16:45:49.0

Неиспользуемые параметры аудита следует отключать.

Are you sure you want to remove the 4 selected audited objects?
The audited statements you remove will no longer be audited on the objects.

NOAUDIT COMMENT ON HR.EMPLOYEES
NOAUDIT INDEX ON HR.EMPLOYEES
NOAUDIT LOCK ON HR.EMPLOYEES
NOAUDIT RENAME ON HR.EMPLOYEES

No Yes

ORACLE

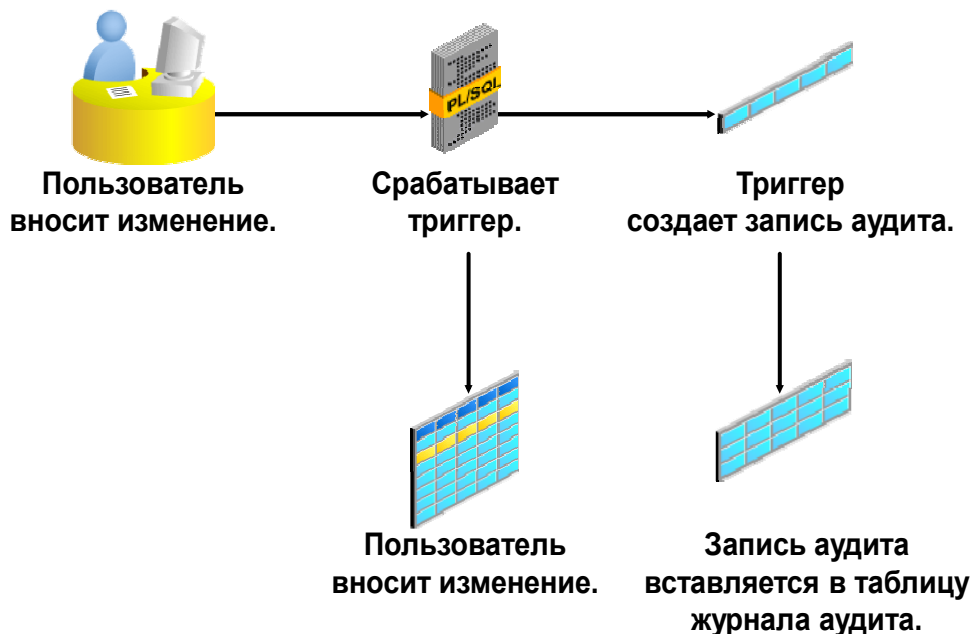
© Oracle, 2009. Все права защищены.

Использование и обслуживание информации аудита

Практические рекомендации:

Аудит снижает производительность прямо пропорционально количеству записей в журнале аудита. Чтобы оптимизировать параметры аудита, исходя из требований вашего Web-узла, включайте только те параметры, которые необходимы с точки зрения политики безопасности. Аудит необходимо сфокусировать таким образом, чтобы уменьшить количество записей в журнале аудита.

Аудит на основе значений



ORACLE

© Oracle, 2009. Все права защищены.

Аудит на основе значений

При аудите базы данных записываются операции добавления, обновления и удаления, выполненные над подверженными аудиту объектами, однако измененные фактические значения не фиксируются. Чтобы расширить аудит БД, при аудите на основе значений регистрируются измененные значения. Для этого используются триггеры БД (управляемые событиями построения PL/SQL).

Если пользователь добавляет, обновляет или удаляет данные из таблицы с соответствующим прикрепленным триггером, этот триггер срабатывает в фоновом режиме, и выполняется копирование информации аудита в предназначенную для этого таблицу.

Аудит на основе значений больше нагружает систему, чем стандартный аудит БД, поскольку код триггера аудита выполняется при каждой операции добавления, обновления или удаления. Степень снижения производительности зависит от эффективности кода триггера. Данный тип аудита следует использовать только тогда, когда получаемой при стандартном аудите информации недостаточно.

Аудит на основе значений реализуется кодом пользователя или сторонних разработчиков. БД Oracle содержит построения PL/SQL, позволяющие создавать системы с аудитом на основе значений.

Аудит на основе значений (продолжение)

Важнейшей составляющей аудита на основе значений является триггер аудита. Это обычный триггер PL/SQL, предназначенный для записи информации аудита.

Вот пример типичного триггера аудита:

```
CREATE OR REPLACE TRIGGER system.hrsalary_audit
  AFTER UPDATE OF salary
  ON hr.employees
  REFERENCING NEW AS NEW OLD AS OLD
  FOR EACH ROW
BEGIN
  IF :old.salary != :new.salary THEN
    INSERT INTO system.audit_employees
    VALUES (sys_context('userenv','os_user'), sysdate,
    sys_context('userenv','ip_address'),
    :new.employee_id ||
    ' salary changed from '||:old.salary||
    ' to '||:new.salary);
  END IF;
END;
/
```

Триггер фокусирует аудит на регистрации изменений в столбце salary таблицы `hr.employees`. При обновлении строки триггер проверяет столбец salary. Если старое значение в столбце salary не совпадает с новым, триггер добавляет запись аудита в таблицу `audit_employees` (которая отдельно создается в схеме `SYSTEM`). Запись аудита содержит имя пользователя, адрес IP, с которого было внесено изменение, первичный ключ, определяющий измененную запись, и фактические значения столбца salary, подвергнутые изменению.

В случае, если стандартный аудит БД не дает достаточно данных, с помощью триггеров базы данных также можно собирать информацию о подключениях пользователей. С помощью триггеров входа в систему администратор может регистрировать данные о пользователях, соединяющихся с базой данных. Например, могут записываться:

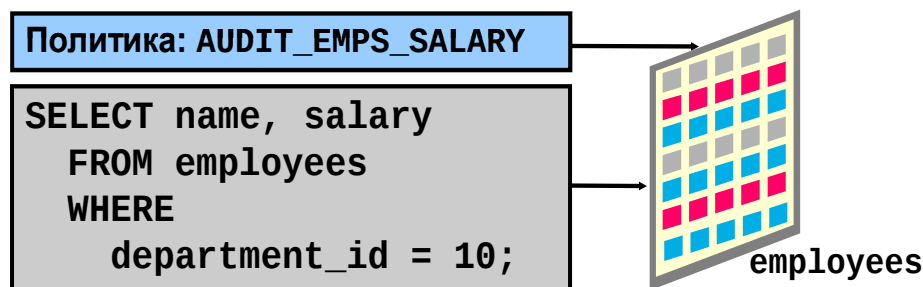
- адрес IP пользователя, выполняющего вход в систему;
- первые 48 символов имени программы, использованной для соединения с экземпляром;
- имя терминала, использованное для соединения с экземпляром.

Полный список пользовательских параметров содержится в разделе «SYS_CONTEXT» руководства *Oracle Database SQL Reference*.

Во многих случаях триггеры на основе значений могут быть заменены функцией детализированного аудита (FGA).

Детализированный аудит

- Мониторинг доступа к данным на основе содержимого.
- Аудит операторов SELECT, INSERT, UPDATE, DELETE и MERGE
- **Может быть связан с одним или более столбцами таблицы или представления.**
- Позволяет вызвать процедуру.
- Управляется пакетом DBMS_FGA.



ORACLE

© Oracle, 2009. Все права защищены.

Детализированный аудит

В записях аудита БД регистрируется факт операции, однако не сохраняется информация о вызвавшем ее операторе. Детализированный аудит (FGA) дополнен возможностью сохранения фактического SQL-оператора, выполняющего запрос или обработку данных.

FGA также позволяет проводить более узкий аудит, чем стандартный аудит БД или аудит на основе значений.

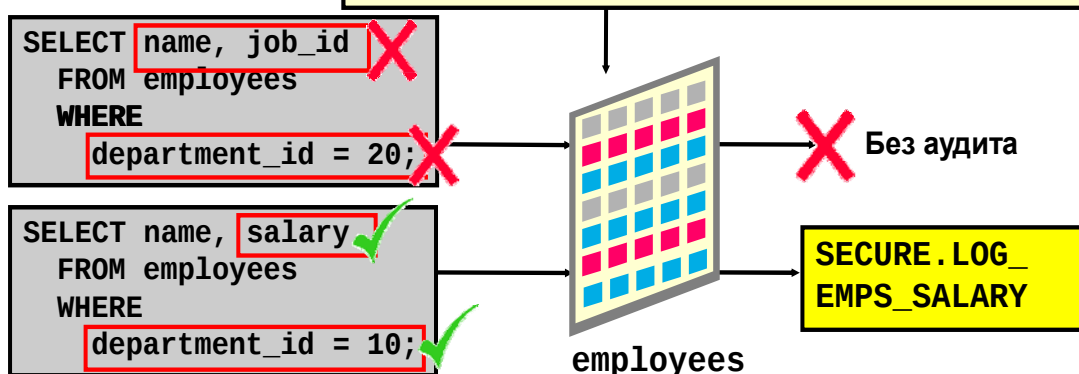
Параметры FGA позволяют направить аудит на отдельные столбцы таблицы или представления и даже задать условия, то есть данные аудита будут сохраняться только при соответствии определенным спецификациям, заданным администратором. Политика FGA поддерживает несколько значимых столбцов. По умолчанию один из этих столбцов присутствует в SQL-операторе, подверженном аудиту. Параметры DBMS_FGA.ALL_COLUMNS и DBMS_FGA.ANY_COLUMNS позволяют проводить аудит на основе присутствия или отсутствия в операторе всех значимых столбцов.

С помощью пакета PL/SQL DBMS_FGA можно создать политику аудита для выбранной таблицы или представления. Если хотя бы одна из строк в блоке запроса совпадает с подверженным аудиту столбцом и соответствует условию аудита, событие аудита вызывает создание записи в журнале аудита. Дополнительно событие аудита может вызвать процедуру. При использовании FGA аудит автоматически фокусируется на уровне операторов. Оператор SELECT может возвращать тысячи строк, но при этом создается только одна запись аудита.

Политика FGA

- Определяет:
 - критерии аудита;
 - действие при аудите.
- Создается с помощью DBMS_FGA .ADD_POLICY

```
dbms_fga.add_policy (  
  object_schema => 'HR',  
  object_name   => 'EMPLOYEES',  
  policy_name  => 'audit_emps_salary',  
  audit_condition=> 'department_id=10',  
  audit_column  => 'SALARY, COMMISSION_PCT',  
  handler_schema => 'secure',  
  handler_module => 'log_emps_salary',  
  enable       => TRUE,  
  statement_types => 'SELECT, UPDATE');
```



ORACLE

© Oracle, 2009. Все права защищены.

Политика FGA

В примере на слайде показано создание политики детализированного аудита с процедурой DBMS_FGA.ADD_POLICY, которая принимает следующие аргументы.

Имя политики

Каждой создаваемой политике FGA присваивается имя. В примере на слайде политике присваивается имя AUDIT_EMPS_SALARY с помощью аргумента:

```
policy_name => 'audit_emps_salary'
```

Условие аудита

Условие аудита — это предикат SQL, который определяет условия запуска события аудита.

В примере аудиту подвергаются все строки для отдела 10, для этого используется аргумент:

```
audit_condition => 'department_id = 10'
```

Примечание. Детализированный аудит предназначен для просмотра набора результатов запроса, поэтому при использовании политики FGA, представленной на слайде, запросы, которые возвращают строки, соответствующие спецификациям политики, приводят к созданию записи аудита. Например, в запросе «select * from employees» могут возвращаться все строки, включающие значение «10» в столбце department_id, поэтому создается строка аудита.

Политика FGA (продолжение)

Столбец аудита

Столбец аудита определяет данные, подвергаемые аудиту. Событие аудита происходит, если этот столбец добавлен в оператор **SELECT** или же если условие аудита позволяет **выборку**.

В примере на слайде выполняется аудит двух столбцов. Для этого используются два аргумента:

```
audit_column => 'SALARY, COMMISSION_PCT'
```

Данный аргумент необязателен. Если его не указать, то возникновение события аудита определяется только аргументом **AUDIT_CONDITION**.

Объект

Объектом может быть таблица или представление, подверженное аудиту. Объект передается в виде двух аргументов:

- схема, содержащая объект;
- имя объекта.

В примере на слайде выполняется аудит таблицы `hr.employees`. Для этого используются следующие аргументы:

```
object_schema => 'hr'
object_name => 'employees'
```

Обработчик

Необязательный обработчик события — это процедура PL/SQL, определяющая дополнительные операции, которые будут выполняться при аудите. Например, обработчик событий может отправлять администратору страницу с предупреждением. Если обработчик не определен, в журнал аудита будет просто добавляться запись о событии аудита. Если обработчик события определен, то в журнал аудита будет добавляться запись о событии аудита, а также будет выполняться обработчик этого события.

Запись о событии аудита содержит имя политики FGA, вызвавшей событие, имя пользователя, выполнившего SQL-оператор, а также сам SQL-оператор и переменные привязки.

Обработчик события передается в виде двух аргументов:

- имя схемы, содержащей программный модуль PL/SQL;
- имя программного модуля PL/SQL.

В примере на слайде выполняется процедура `SECURE.LOG_EMPS_SALARY` с помощью аргументов:

```
handler_schema => 'secure'
handler_module => 'log_emps_salary'
```

По умолчанию в журнал аудита всегда записываются текст SQL и информация о привязках **SQL с объектами LOB**. Настройки по умолчанию можно изменить (например, если в системе ожидается снижение производительности).

Статус

Состояние показывает, включена ли политика FGA. В примере на слайде политика включена с помощью следующего аргумента:

```
enable => TRUE
```

Аудит DML-операторов: вопросы

- Записи подвергаются аудиту, если выполнен предикат FGA и произошло обращение к значимым столбцам.
- Аудит операторов DELETE выполняется независимо от указанных столбцов.
- При аудите операторов MERGE также выполняется аудит вложенных операторов INSERT, UPDATE и DELETE.

Аудит не выполняется, так как ни одна из записей не относится к отделу 10.

```
UPDATE hr.employees  
SET salary = 1000  
WHERE commission_pct = .2;
```



```
UPDATE hr.employees  
SET salary = 1000  
WHERE employee_id = 200;
```

ORACLE

© Oracle, 2009. Все права защищены.

Аудит DML-операторов: вопросы

Если для DML-операторов определена политика FGA, то аудит DML-оператора происходит при обработке строк данных (старых и новых) в соответствии с критериями предиката политики.

Однако если определение политики также содержит значимые столбцы, то аудит оператора выполняется, когда данные соответствуют предикату политики FGA и данный оператор обращается к этим значимым столбцам.

Для операторов DELETE указывать значимые столбцы бессмысленно, так как оператор DELETE может обрабатывать любые столбцы таблицы. Поэтому аудит оператора DELETE выполняется всегда вне зависимости от значимых столбцов.

FGA поддерживает операторы MERGE. Для вложенных операторов INSERT, UPDATE и DELETE также выполняется аудит, если они соответствуют политикам FGA INSERT, UPDATE или DELETE.

Если бы использовалась представленная ранее политика FGA, то аудиту подвергся бы только первый из операторов. Никто из сотрудников отдела 10 не получает комиссионных, однако employee_id=200 указывает на сотрудника из этого отдела.

Руководящие принципы FGA

- Для аудита всех строк используется условие аудита `null`.
- Для аудита всех столбцов используется столбец аудита `null`.
- У политик должны быть уникальные имена.
- При создании политики аудит можно настраивать только для уже существующих таблиц или представлений.
- Если условие аудита содержит синтаксическую ошибку, при обращении к подверженному аудиту объекту возникает ошибка `ORA - 28112`.
- Если таблица не содержит подверженный аудиту столбец, для ее строк аудит не выполняется.
- Если обработчик события не задан, сообщение об ошибке не отображается, а запись аудита все равно создается.

ORACLE

© Oracle, 2009. Все права защищены.

Руководящие принципы FGA

При аудите FGA операторов `SELECT` регистрируется сам оператор, а не фактические строки. Тем не менее, если FGA совместить с запросом `Flashback`, то будет возможно восстановить эти строки на тот момент времени.

Подробнее о запросах `Flashback` описано в уроке «Выполнение мгновенного отката».

Более подробную информацию о пакете `DBMS_FGA` см. в руководстве *Oracle Database PL/SQL Packages and Types Reference*.

Аудит SYSDBA

Пользователи с полномочиями SYSDBA или SYSOPER могут соединяться с закрытой базой данных.

- Журнал аудита должен храниться за пределами БД.
- Соединения от имени SYSDBA или SYSOPER всегда подвергаются аудиту.
- Можно включить дополнительные операции аудита для SYSDBA или SYSOPER с помощью AUDIT_SYS_OPERATIONS.
- Для контроля журнала аудита можно использовать AUDIT_FILE_DEST.



ORACLE

© Oracle, 2009. Все права защищены.

Аудит SYSDBA

Пользователи SYSDBA и SYSOPER обладают полномочиями для запуска и остановки базы данных. Так как им доступно внесение изменений в закрытую базу данных, журнал аудита для этих полномочий должен храниться за пределами базы данных. БД Oracle автоматически регистрирует события входа в систему пользователей SYSDBA и SYSOPER. Это дает ценную возможность следить за санкционированными и несанкционированными действиями SYSDBA и SYSOPER, однако в этом есть смысл только при просмотре журнала аудита ОС.

В БД Oracle всегда регистрируется вход в систему пользователей с полномочиями. Прочие их действия регистрируются, если был отдельно включен аудит администраторов БД. Для включения аудита пользователей SYSDBA и SYSOPER необходимо настроить следующий параметр инициализации:

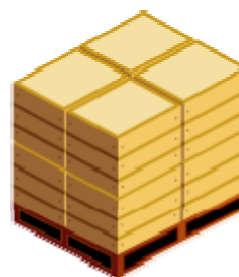
AUDIT_SYS_OPERATIONS=TRUE (по умолчанию используется значение FALSE.)

Если производится аудит действий SYS, параметр инициализации AUDIT_FILE_DEST задает каталог для хранения записей аудита. На платформах Windows для записей аудита по умолчанию используется журнал событий Windows. На платформах UNIX и Linux записи аудита хранятся в AUDIT_FILE_DEST.

Обслуживание журнала аудита

При обслуживании журнала аудита следует придерживаться следующих рекомендуемых принципов:

- просмотр и хранение старых записей;
- **предотвращение неполадок хранения;**
- предотвращение утраты записей.



ORACLE

© Oracle, 2009. Все права защищены.

Обслуживание журнала аудита

Журнал аудита любого типа требует обслуживания. Общее обслуживание включает просмотр записей аудита и удаление старых записей из базы данных или операционной системы. Журналы аудита могут увеличиваться и заполнять все доступное пространство. Заполнение файловой системы может привести к отказу системы или просто снизить производительность. Если журнал аудита БД полностью заполняет табличное пространство, операции аудита не выполняются. Если журнал аудита заполняет системное табличное пространство, то прежде чем аудит будет остановлен, ухудшается производительность всех других операций.

При стандартном аудите журнал аудита хранится в таблице AUD\$. Журнал аудита FGA хранится в таблице FGA_LOG\$. По умолчанию обе эти таблицы создаются в табличном пространстве SYSTEM. Чтобы переместить эти таблицы в другое табличное пространство, можно воспользоваться служебными программами экспорта и импорта насоса данных.

Примечание. Перемещение таблиц аудита за пределы табличного пространства SYSTEM не поддерживается.

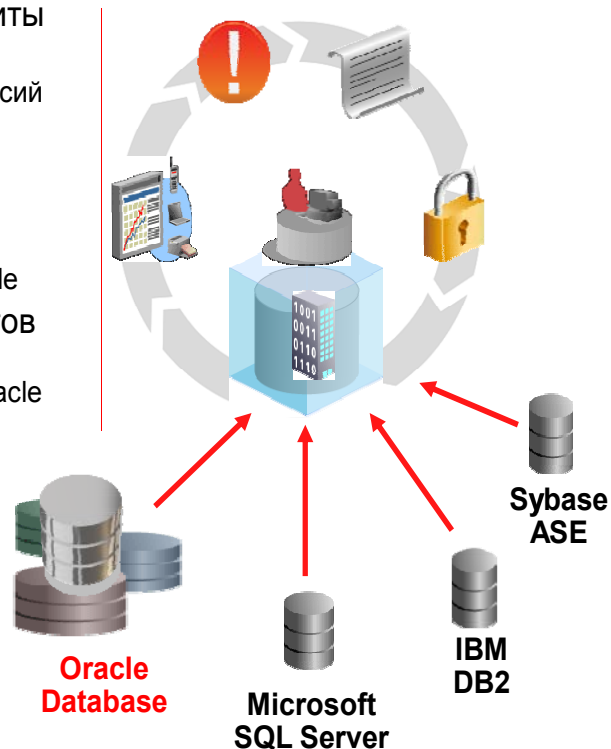
При удалении записей из таблиц аудита возможна потеря записей аудита.

Практические рекомендации:

следует использовать экспорт на основе отметки времени, а затем удалить строки из журнала аудита по совпадающей отметке времени.

Oracle Audit Vault

- Консолидация и обеспечение защиты данных аудита
 - Oracle 9i версии 2 и более поздних версий
 - SQL Server 2000, 2005
 - IBM DB2 UDB 8.5 и 9.2
 - Sybase ASE 12.5–15.0
 - **Защита и масштабируемость**
 - Очистка исходных данных аудита Oracle
- Централизованное создание отчетов
 - Обновленный интерфейс отчетов с использованием популярной среды Oracle Application Express
 - Стандартные отчеты для обеспечения соответствия
 - Новые пользовательские отчеты
- Уведомление об угрозах безопасности
 - Обнаружение событий, имеющих отношение к безопасности, и создание уведомлений о них



ORACLE

© Oracle, 2009. Все права защищены.

Oracle Audit Vault

К ключевым преимуществам Oracle Audit Vault относятся следующие:

- Oracle Audit Vault осуществляет прозрачный сбор и консолидацию данных аудита из баз данных Oracle, начиная с базы данных Oracle9i версии 2, Microsoft SQL Server 2000 и 2005, IBM DB2 Unix, Linux, Windows 8.2 и 9.5 и Sybase ASE 12.5–15.0.
- Oracle Audit Vault помогает организациям упрощать создание отчетов по соответствию требованиям за счет встроенных и настраиваемых отчетов. Кроме того, Oracle Audit Vault предоставляет открытую схему хранения журналов аудита, доступ к которым можно получить с помощью Oracle BI Publisher, Oracle Application Express или любых средств создания отчета сторонних разработчиков.
- Oracle Audit Vault помогает обнаруживать и предотвращать угрозы нарушения безопасности внутрикорпоративной информации (инсайдерские атаки), уведомляя вас о подозрительных действиях.
- Центральным компонентом Oracle Audit Vault является надежное и масштабируемое хранилище данных аудита, созданное на основе технологии хранения данных Oracle и защищенное с помощью продуктов обеспечения безопасности баз данных Oracle, включая Oracle Database Vault и Oracle Advanced Security. Oracle Audit Vault включает Oracle Partitioning для повышения производительности и расширения возможностей управления.
- Oracle Audit Vault помогает организациям сокращать ИТ-расходы за счет централизованного управления настройками (политиками) аудита баз данных, благодаря чему значительно упрощаются задачи сотрудников, ответственных за ИТ-безопасность, и внутренних аудиторов.

Для получения дополнительных сведений см. документацию по Audit Vault.

Экспресс-тест

В ходе стандартной операции аудита базы данных осуществляется сбор данных о предыдущих и последующих изменениях транзакции DML.

1. Да
2. Нет

ORACLE

© Oracle, 2009. Все права защищены.

Ответ: 2

Экспресс-тест

Аудит SYSDBA и SYSOPER включен по умолчанию.

1. Да
2. Нет

ORACLE

© Oracle, 2009. Все права защищены.

Ответ: 2

Заключение

В ходе прохождения урока вы научились:

- описывать обязанности АБД по обеспечению безопасности и проведению аудита;
- **включать стандартный аудит БД;**
- задавать параметры аудита;
- просматривать информацию об аудите;
- обслуживать журнал аудита.



ORACLE

© Oracle, 2009. Все права защищены.

Обзор упражнения 11: внедрение защиты базы данных Oracle

В данном упражнении рассматриваются следующие темы:

- включение стандартного аудита БД;
- определение параметров аудита для таблицы **HR.JOBS**;
- обновление таблицы;
- просмотр информации об аудите;
- обслуживание журнала аудита.

ORACLE

© Oracle, 2009. Все права защищены.

12

Обслуживание базы данных

ORACLE®

© Oracle, 2009. Все права защищены.

Цели

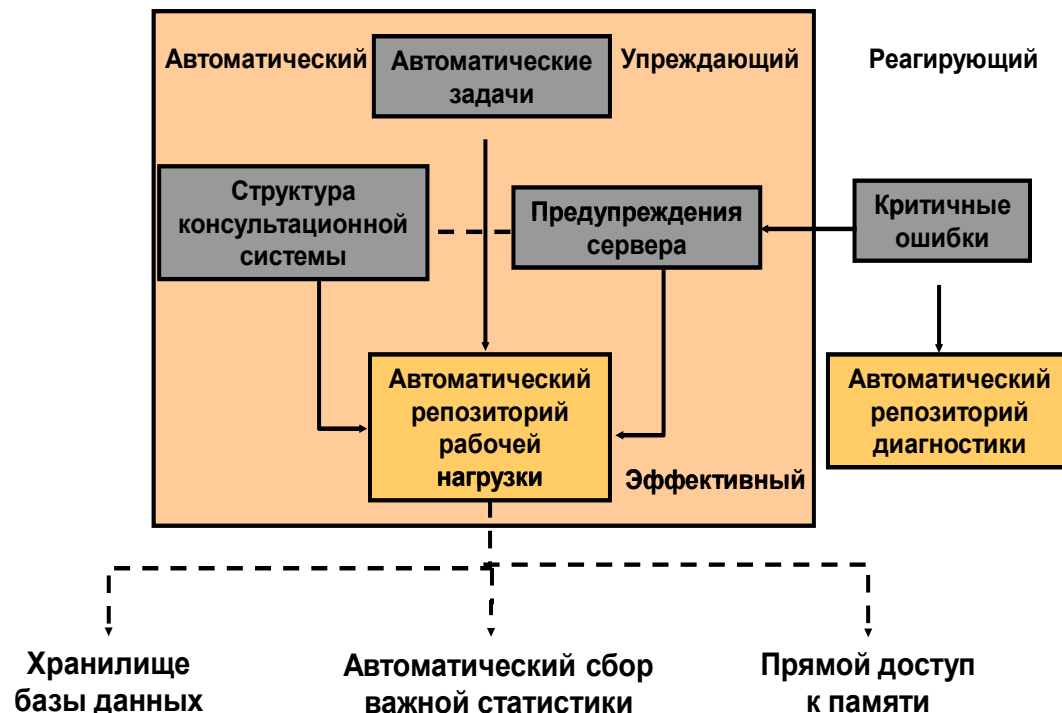
По завершении данного урока вы сможете:

- управлять статистикой оптимизатора;
- управлять автоматическим репозиторием рабочей нагрузки (AWR);
- использовать монитор автоматической диагностики базы данных (ADDM);
- описать консультационную систему;
- устанавливать пороговые значения предупреждений;
- использовать предупреждения, генерируемые сервером;
- использовать автоматические задачи.

ORACLE

© Oracle, 2009. Все права защищены.

Обслуживание базы данных



ORACLE

© Oracle, 2009. Все права защищены.

Обслуживание базы данных

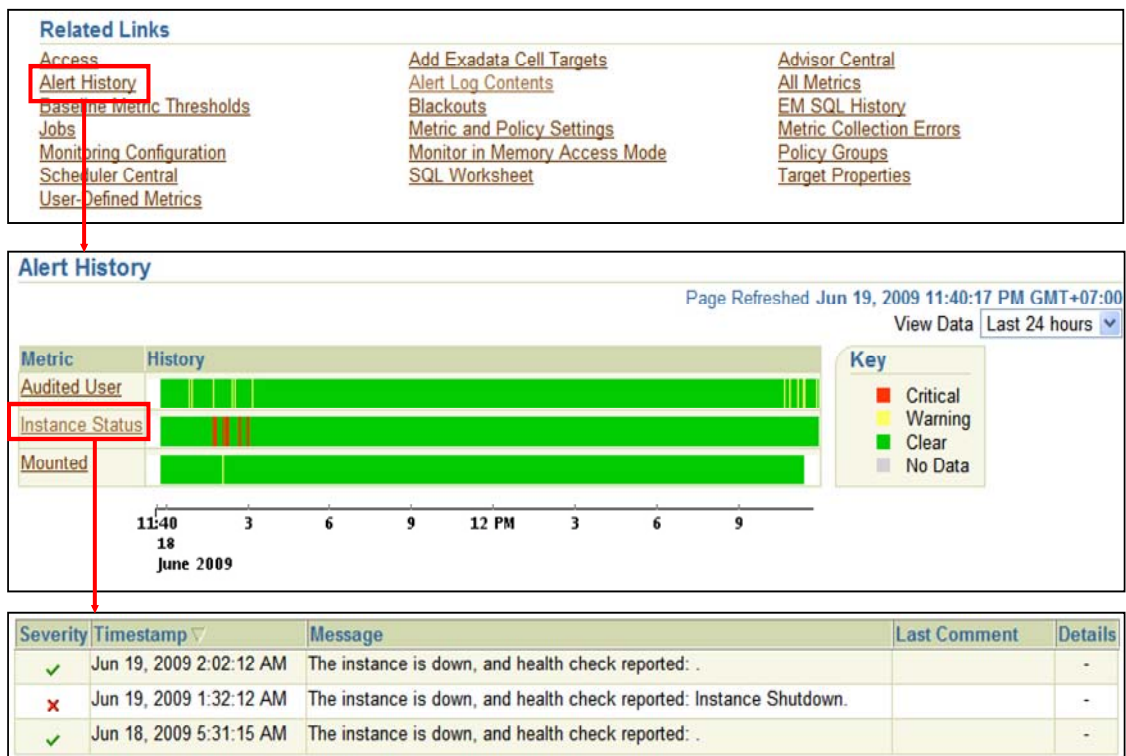
Выполнение упреждающего обслуживания базы данных не представляет особой сложности благодаря тщательно продуманной инфраструктуре базы данных Oracle, в которую включены описанные ниже основные элементы.

- Автоматический репозиторий рабочей нагрузки (AWR) встроен в каждую базу данных Oracle. Через равные интервалы времени сервер базы данных Oracle выполняет снимки критически важных данных статистики и информации о рабочей нагрузке и сохраняет полученные данные в репозитории AWR. Такие полученные данные могут использоваться для самостоятельного анализа и для выполнения анализа самой базой данных.
- Используя автоматические задачи, база данных выполняет текущие операции по **обслуживанию, такие как создание регулярных резервных копий, обновление статистики оптимизатора, а также проверки работоспособности базы данных.**

Реагирующее обслуживание базы данных включает в себя устранение критических ошибок и условий, выявленных средствами проверки работоспособности базы данных:

- При возникновении проблем, которые не могут быть устранены автоматически и которые требуют уведомления администраторов (например, недостаток места на диске), базы данных Oracle предоставляют механизм предупреждений, генерируемых сервером. По умолчанию база данных Oracle выполняет наблюдение за собственным состоянием и отправляет предупреждения о возникающих проблемах. Предупреждения уведомляют о возникших проблемах и часто содержат рекомендации по их устранению.
- Такие рекомендации формируются рядом помощников, каждый из которых отвечает за собственную подсистему. Например, существуют помощники, ответственные за память, сегмент данных и подсистему SQL.

Просмотр хронологии предупреждений



ORACLE

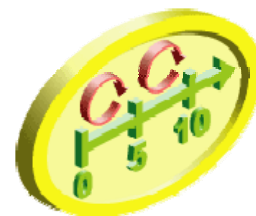
© Oracle, 2009. Все права защищены.

Просмотр хронологии предупреждений

На странице «Alert History» (Хронология предупреждений) находится таблица с хронологией предупреждений текущей базы данных в указанном диапазоне времени. Предупреждение указывает на наличие потенциальной проблемы: простое предупреждение, критический порог отслеживаемой метрики или же недоступность ресурса. Щелкните имя метрики на странице «Alert History» (Хронология предупреждений), чтобы отобразить подробную статистику, графики и действительные временные метки для каждого предупреждения. Доступно также поле для ввода комментариев, относящихся к предупреждению, например информации о разрешении.

Терминология

- Автоматический репозиторий рабочей нагрузки (AWR): инфраструктура, предназначенная для сбора данных, их анализа и выдачи рекомендаций
- Базовая конфигурация AWR: набор снимков репозитория AWR для целей сравнения работоспособности
- Метрика: степень изменений в накопительной статистике
- Статистика: наборы данных, предоставляющие подробные сведения о базе данных и объектах
 - Статистика оптимизатора: используется оптимизатором запросов
 - Статистика базы данных: используется для наблюдения за производительностью
- Пороговое значение: предельное значение, с которым сравниваются значения метрик



ORACLE

© Oracle, 2009. Все права защищены.

Терминология

Автоматический репозиторий рабочей нагрузки (AWR) используется внутренними компонентами сервера Oracle для сбора, обработки, хранения и использования статистики производительности с целью обнаружения проблем и выполнения самонастройки. *История активных сеансов (ASH)* представляет собой историю активности последних сеансов, сохраненную в репозитории AWR.

Статистика представляет собой наборы подробных данных о базе данных и находящихся в ней объектах. Статистика оптимизатора используется оптимизатором запросов для выбора оптимального плана выполнения каждого оператора SQL. Статистика базы данных используется для наблюдения за производительностью.

Снимки репозитория AWR включают в себя статистику и показатели базы данных, статистику приложений (размеры транзакций, время ответа), статистику операционной системы и прочие показатели. *Базовая конфигурация AWR* представляет собой набор снимков репозитория AWR, собранных за какой-либо период времени. Базовая конфигурация используется для сравнения производительности — либо текущей производительности с базовой конфигурацией, либо одной базовой конфигурации с другой.

Сохранение базовой конфигурации *окна системных изменений* в Oracle Database 11g выполняется по умолчанию. Базовая конфигурация окна системных изменений представляет собой меняющийся набор снимков по умолчанию за последние восемь дней. Такая базовая конфигурация вступает в силу после сбора достаточного количества данных и выполнения статистических расчетов. По умолчанию расчеты статистики выполняются по расписанию каждую субботу в полночь.

Оптимизатор Oracle: обзор

Оптимизатор Oracle определяет наиболее эффективный план выполнения и отвечает за наиболее важный этап в обработке любых операторов SQL.

Оптимизатор:

- выполняет вычисление выражений и условий;
- использует статистику объектов и системы;
- принимает решения о способе доступа к данным;
- принимает решения о порядке объединения таблиц;
- определяет наиболее эффективный путь.

ORACLE

© Oracle, 2009. Все права защищены.

Оптимизатор Oracle: обзор

Оптимизатор представляет собой компонент базы данных Oracle, который создает план выполнения оператора SQL. Определение плана выполнения является важным шагом в обработке любого оператора SQL, поскольку может в большой степени повлиять на время выполнения.

План выполнения представляет собой ряд последовательных операций с целью выполнения оператора. Оптимизатор учитывает много факторов, имеющих отношение к используемым объектам и условиям, заданным в запросе. Необходимая для оптимизатора информация включает в себя:

- статистику, собранную о системе (операции ввода/вывода, характеристики процессора и прочее), а также об объектах схемы (количество строк, наличие индексов и т.д.);
- информацию из словаря;
- квалификаторы фразы WHERE;
- указанные разработчиком подсказки оптимизатору.

Просмотреть выбранный оптимизатором план выполнения можно с помощью таких диагностических инструментов, как диспетчер Enterprise Manager, EXPLAIN PLAN и SQL*Plus AUTOTRACE.

Примечание. В зависимости от функциональных возможностей оптимизатор Oracle может называться *оптимизатором запросов* или *оптимизатором автоматической настройки*.

Статистика оптимизатора

Статистика оптимизатора:

- представляет собой снимок на какой-либо момент времени;
- не сбрасывается до следующего перезапуска экземпляра;
- собирается автоматически.

```
SQL> SELECT COUNT(*) FROM hr.employees;
COUNT(*)
-----
214
SQL> SELECT num_rows FROM dba_tables
2 WHERE owner='HR' AND table_name = 'EMPLOYEES';
NUM_ROWS
-----
107
```

ORACLE

© Oracle, 2009. Все права защищены.

Статистика оптимизатора

Статистика оптимизатора состоит из статистики таблиц, столбцов, индексов и системы. Статистика таблиц и индексов хранится в словаре данных. Эти виды статистики не предназначены для выдачи результатов в режиме реального времени. Они предоставляют оптимизатору *статистически* верный снимок мест хранения данных и их распределения, который используется оптимизатором для принятия решений о том, как получить требуемые данные.

Собираются следующие данные статистики:

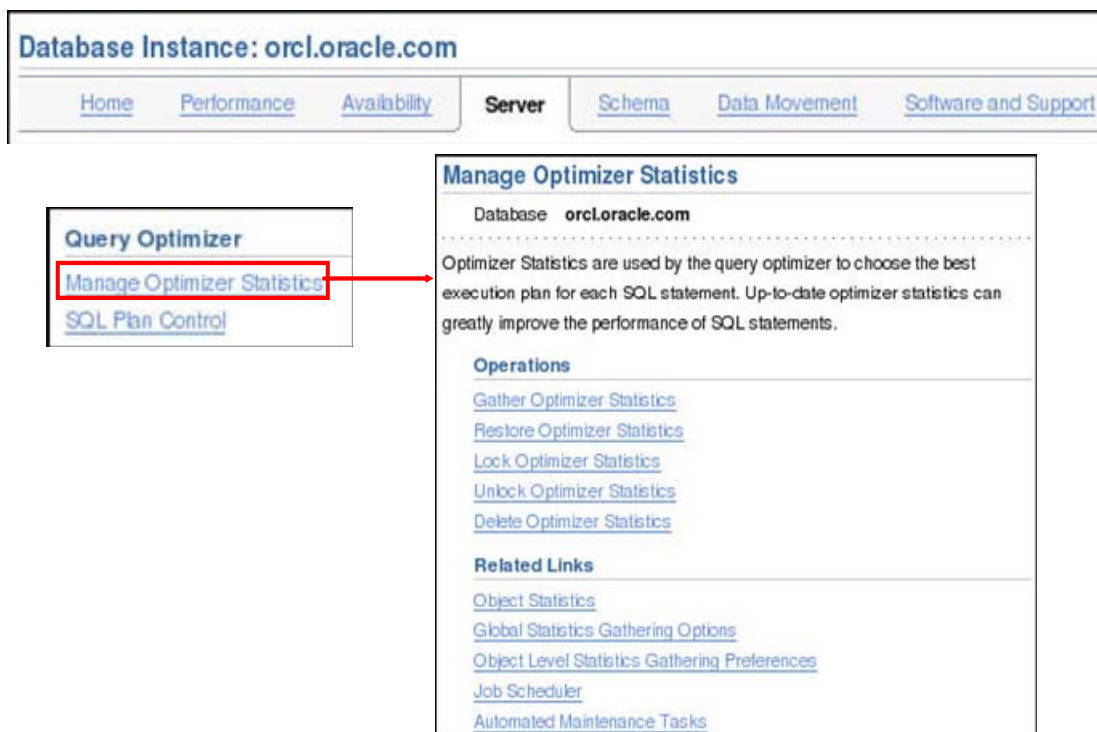
- размер таблицы или индекса в блоках базы данных;
- число строк;
- средний размер строки и количество цепочек (только для таблиц);
- высота и количество удаленных рядов листьев (только для индексов).

Указанные данные изменяются при вставке, удалении и изменении данных. Поскольку поддержание статистических данных в актуальном состоянии не выполняется в связи с отрицательным влиянием на производительность, обновление статистики таблиц и индексов выполняется периодически.

За автоматический сбор статистики оптимизатора отвечает автоматическое задание, по умолчанию запускаемое ежедневно в течение заданных интервалов обслуживания. Системная статистика представляет собой показатели операционной системы, используемые оптимизатором. Сбор такой статистики автоматически не выполняется. Дополнительные сведения о сборе системной статистики см. в руководстве *Oracle Database Performance Tuning Guide*.

Статистика оптимизатора — это не статистика производительности базы данных, сбор которой осуществляется на основе снимков репозитория AWR.

Использование страницы управления статистикой оптимизатора



ORACLE

© Oracle, 2009. Все права защищены.

Использование страницы управления статистикой оптимизатора

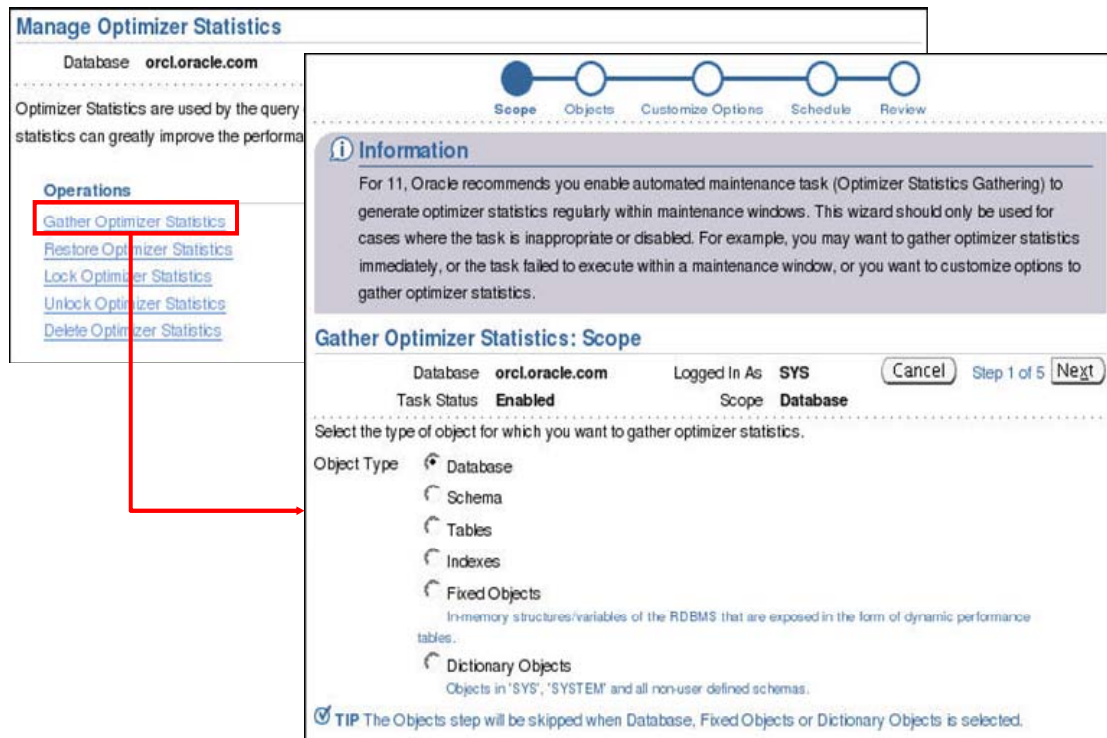
Для управления статистикой оптимизатора в диспетчере Enterprise Manager перейдите на вкладку «Server» (Сервер) и в разделе «Query Optimizer» (Оптимизатор запросов) нажмите ссылку **Manage Optimizer Statistics** (управление статистикой оптимизатора). С помощью данной страницы можно выполнять перечисленные ниже задачи, связанные со статистикой.

- Ручной сбор статистики оптимизатора.
- Восстановление статистики оптимизатора на определенный момент времени в прошлом. Выбранный момент времени должен находиться в пределах периода хранения статистики оптимизатора, длительность которого по умолчанию составляет 30 дней.
- Блокировка статистики оптимизатора для обеспечения того, что статистика для некоторых объектов не будет изменена. Это бывает полезным в случаях, когда был выполнен расчет статистики для некоторой таблицы в то время, когда имелись достаточно репрезентативные данные, и после этого требуется всегда использовать такие статистические данные. Если статистика заблокирована, то никакие изменения содержимого таблицы не окажут на нее никакого влияния.
- Отмена ранее установленной блокировки статистики оптимизатора.
- Удаление статистики оптимизатора.

Практические рекомендации:

Для сбора статистики оптимизатора используются автоматические задачи. Чтобы активировать задачу по сбору статистики оптимизатора, необходимо убедиться, что параметр инициализации **STATISTICS_LEVEL** имеет значение **TYPICAL** или **ALL**.

Ручной сбор статистики оптимизатора



ORACLE

© Oracle, 2009. Все права защищены.

Ручной сбор статистики оптимизатора

Иногда возникает необходимость получить статистику вручную, например, когда содержимое таблицы в значительной степени изменяется в промежутке между запусками задач и статистика таблицы перестает быть репрезентативной. Эта процедура является обычной для больших таблиц, размер которых изменяется более чем на 10 процентов в течение 24 часов.

Практические рекомендации: установите интервал сбора статистики таким, чтобы в течение такого интервала изменение размера таблицы не превышало 10 процентов. Для этого может потребоваться ручной сбор статистики либо дополнительные интервалы обслуживания.

Сбор статистики вручную можно выполнить с помощью диспетчера Enterprise Manager либо пакета DBMS_STATS. Сбор системной статистики может быть выполнен только с помощью пакета DBMS_STATS. Системная статистика описывает аппаратные характеристики системы, такие как производительность и использование ввода-вывода и ЦП, для оптимизатора запросов

Ручной сбор статистики оптимизатора (продолжение)

С помощью меню «Gather Optimizer Statistics» (Сбор статистики оптимизатора) можно запустить мастер, позволяющий выбрать область, объекты, параметры и расписание задания для сбора статистики оптимизатора. Мастер создает задание с именем `DBMS_STATS.GATHER_*_STATS` с областью действия в пределах таблицы, схемы или всей базы данных. В данном мастере устанавливаются необходимые параметры, которые используются пакетом `DBMS_STATS`, а также расписание запуска задания в определенное время.

Ручной сбор статистики не рекомендуется выполнять в качестве стандартного метода сбора, поскольку сбор статистики в интервалах обслуживания выполняется более эффективно и с минимальным влиянием на работу пользователей. Такое задание можно создать вручную, если автоматическое задание не выполнилось либо было отключено.

Статистику оптимизатора также можно собрать непосредственно с помощью пакета `DBMS_STATS`:

```
SQL> EXEC dbms_stats.gather_table_stats('HR','EMPLOYEES');
SQL> SELECT num_rows FROM dba_tables
       2  WHERE owner='HR' AND table_name = 'EMPLOYEES';
       NUM_ROWS
       -
       214
```

Обратите внимание, что количество строк сейчас правильно отражает состояние таблицы на момент сбора статистики. Пакет `DBMS_STATS` также позволяет выполнить ручной сбор статистики для всей схемы и для всей базы данных.

Изменение системной статистики происходит только при значительном изменении рабочей нагрузки. Как следствие, системная статистика не требует частой корректировки. Для сбора системной статистики за указанный период используется процедура `DBMS_STATS.GATHER_SYSTEM_STATS`, однако можно вручную выполнить сбор системной статистики либо остановить его.

Практические рекомендации: при создании базы данных выполните следующую команду:

```
SQL> EXEC dbms_stats.gather_system_stats('NOWORKLOAD');
```

При использовании параметра `NOWORKLOAD` требуемое время выполнения увеличивается на несколько минут (в зависимости от размера базы данных), но при этом выполняется оценка характеристик ввода/вывода, таких как среднее время чтения при поиске и скорость передачи данных при вводе/выводе.

Предпочтения для выполнения сбора статистики



ORACLE

© Oracle, 2009. Все права защищены.

Предпочтения для выполнения сбора статистики

Процедуру `DBMS_STATS.GATHER_*_STATS` можно вызывать на различных уровнях и выполнять сбор статистики для всей базы данных либо для отдельных объектов, например таблиц. При вызове процедуры `GATHER_*_STATS` некоторым параметрам задаются значения по умолчанию. Такие значения по умолчанию подходят для большинства объектов базы данных, однако для некоторых объектов, таких как схемы, значения по умолчанию должны быть изменены. Чтобы не выполнять ручную установку для каждого из таких объектов, Oracle Database 11g дает возможность установить значения (называемые *предпочтениями*) для отдельных объектов, схем или баз данных, либо изменить значения по умолчанию с помощью глобальной команды.

Предпочтения устанавливают параметры, передаваемые процедурам сбора статистики. Процедуры `SET_*_PREFS` создают значения предпочтений для всех объектов, которые не принадлежат `SYS` или `SYSTEM`. Как правило, администратор базы данных устанавливает глобальные предпочтения для всех параметров, область действия которых распространяется на всю базу данных. Это применяется для всех параметров, которым разрешено присваивать значение по умолчанию.

Процедура `SET_DATATBASE_PREFS` обрабатывает все таблицы и схемы базы данных и устанавливает заданное значение предпочтения. Процедура `SET_SCHEMA_PREFS` используется для обработки таблиц заданной схемы. Процедура `SET_TABLE_PREFS` используется для установки значения предпочтения для одной таблицы.

Предпочтения для выполнения сбора статистики (продолжение)

Все предпочтения для объектов, независимо от того, задаются ли они на уровне базы данных, схемы или таблицы, хранятся в одной таблице. Изменение предпочтений на уровне схемы имеет более высокий приоритет, чем предпочтения, которые были ранее установлены на уровне таблицы.

В ходе выполнения процедуры сбора статистики получают значения предпочтений, относящиеся к каждому конкретному объекту. Просмотреть предпочтения для каждого объекта можно с помощью представления `DBA_TAB_STAT_PREFS`. Все значения предпочтений, которые не заданы самому объекту, будут установлены в качестве предпочтений глобального уровня. Просмотреть значения глобальных предпочтений можно путем вызова процедуры `DBMS_STATS.GET_PREFS` для каждого значения предпочтения.

Можно устанавливать, получать, удалять, экспортировать и импортировать такие значения предпочтений на уровне таблицы, схемы, базы данных и глобальном уровне.

Предполагается, что значения предпочтений будут задаваться от глобального уровня до уровня таблицы. В последнюю очередь значения задаются для наименьшей группы.

Настройки в Oracle Database 11g:

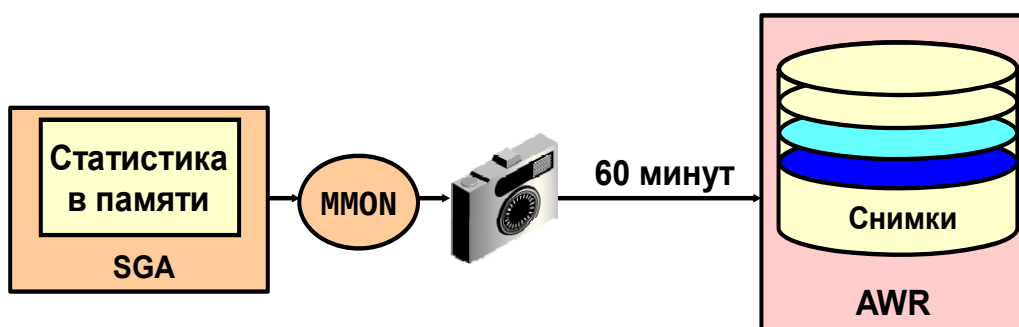
- **CASCADE** определяет, будет ли сбор статистики индекса выполняться как часть процедуры сбора статистики таблиц.
- **DEGREE** устанавливает степень параллелизма, используемого при сборе статистики.
- **PUBLISH** используется для указания того, будет ли статистика опубликована в словаре или же сохранена в частной области. Это позволяет администратору базы данных проверять достоверность статистики перед публикацией в словаре данных с помощью процедуры `PUBLISH_PENDING_STATS`.
- **STALE_PERCENT** используется для определения порогового уровня, при достижении которого считается, что статистика объекта устарела. Значение содержит процент строк, измененных с момента последнего сбора статистики. В примере выполняется изменение значения по умолчанию с 10 до 13 процентов для таблицы `SH.SALES`.
- **INCREMENTAL** используется для сбора глобальной статистики по секционированным таблицам инкрементным способом.
- **METHOD_OPT** определяет столбцы и параметры гистограммы, которые используются для сбора статистики столбцов.
- **GRANULARITY** определяет степень детализации статистики для сбора (относится только к секционированным таблицам).
- **NO_INVALIDATE** используется для указания того, следует ли объявлять курсоры недействительными.
- **ESTIMATE_PERCENT** используется для определения количества строк выборки, гарантирующего получение достоверной статистики. Значение представляет собой процент от количества строк таблицы.

Примечание. Дополнительные сведения об указанных предпочтениях см. в документации `DBMS_STATS` в руководстве *Oracle Database PL/SQL Packages and Types Reference*.

Значения предпочтений на уровне таблицы, схемы или базы данных могут быть удалены с помощью соответствующих процедур `DBMS_STATS.DELETE_*_PREFS`. Глобальные предпочтения можно сбросить и восстановить рекомендуемые значения с помощью процедуры `DBMS_STATS.RESET_PARAM_DEFAULTS`.

Автоматический репозиторий рабочей нагрузки (AWR)

- Встроенный репозиторий информации о производительности
- Снимки параметров базы данных выполняются каждые **60 минут** и хранятся **восемь дней**
- Основа для всех функций самообслуживания



ORACLE

© Oracle, 2009. Все права защищены.

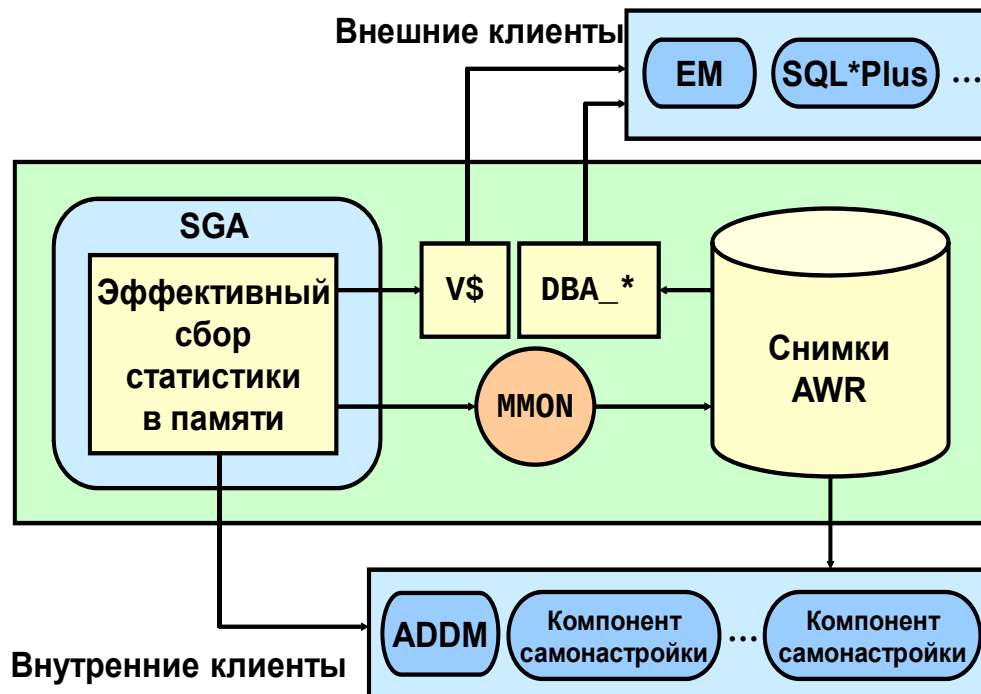
Автоматический репозиторий рабочей нагрузки (AWR)

Репозиторий AWR представляет собой инфраструктуру, предоставляющую компонентам сервера Oracle Database 11g службы для сбора, хранения и использования статистики производительности с целью обнаружения проблем и выполнения самонастройки. Его можно рассматривать в качестве хранилища данных для статистики базы данных, показателей и прочей информации.

По умолчанию каждые 60 минут база данных автоматически выполняет сбор данных статистики с помощью агентов по сбору статистики SGA и сохраняет их в репозитории AWR в виде снимков. Указанные снимки сохраняются на диске с помощью фонового процесса, называемого Manageability Monitor (MMON). По умолчанию такие снимки хранятся в течение восьми дней. Интервал сбора снимков и период их хранения можно изменить.

Репозиторий AWR содержит сотни таблиц, принадлежащих схеме SYSMAN и хранящихся в табличном пространстве SYSAUX. Вместо этого для работы с репозиторием AWR следует использовать диспетчер Enterprise Manager или пакет DBMS_WORKLOAD_REPOSITORY. Прямые DML для таблиц репозитория не поддерживаются.

Инфраструктура AWR



ORACLE

© Oracle, 2009. Все права защищены.

Инфраструктура AWR

Инфраструктура репозитория AWR включает две основные части:

- средство сбора статистики в памяти, которое используется компонентами Oracle Database 11g для сбора статистики. Данная статистика хранится в памяти для целей оценки производительности. Хранящуюся в памяти статистику можно просмотреть с помощью представлений динамической производительности (V\$).
- снимки репозитория AWR, которые представляют собой неизменяемую часть указанного средства. Снимки репозитория AWR можно просмотреть с помощью представлений данных словаря и средства управления базой данных Enterprise Manager Database Control.

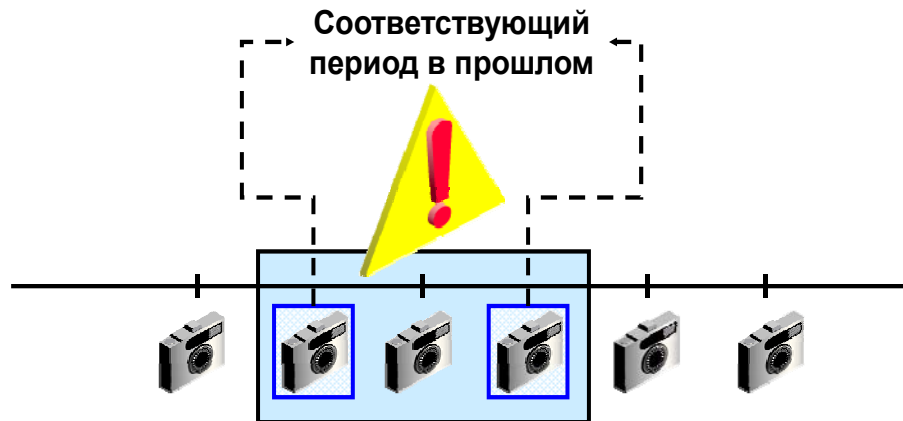
Статистика хранится в постоянном хранилище по нескольким причинам:

- Статистика необходима для восстановления после аварийных завершений работы экземпляра.
- Для некоторых видов анализа необходимы хронологические данные для целей сравнения базовых конфигураций.
- Переполнение памяти. Когда старая статистика заменяется новой из-за недостатка памяти, замененные данные могут быть сохранены для дальнейшего использования.

Фоновый процесс MMON регулярно сохраняет версии памяти статистики на диске.

С помощью репозитория AWR база данных Oracle предоставляет способ автоматического получения хронологических данных статистики без участия администратора базы данных.

Базовые конфигурации AWR



```
DBMS_WORKLOAD_REPOSITORY.CREATE_BASELINE ( -  
    start_snap_id IN NUMBER,  
    end_snap_id   IN NUMBER,  
    baseline_name IN VARCHAR2);
```

ORACLE

© Oracle, 2009. Все права защищены.

Базовые конфигурации AWR

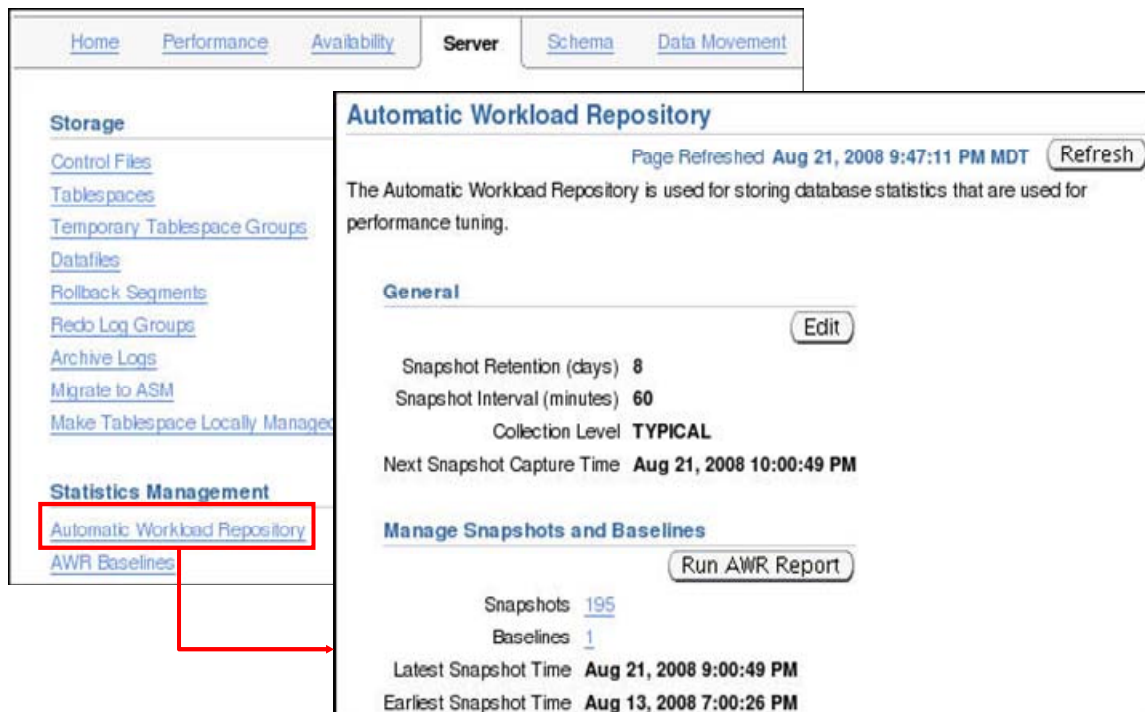
Базовая конфигурация AWR представляет собой набор снимков репозитория AWR. Как правило, это набор данных снимков за важный период, которые были отмечены и хранятся в AWR. Базовая конфигурация определяется парой снимков. Снимки идентифицируются по их порядковым номерам (`snap_id`) или времени начала и завершения. Каждый набор снимков включает начальный и конечный снимки, а также все снимки, созданные между ними. Наборы снимков используются для хранения данных, содержащихся в снимках. Поэтому по умолчанию снимки, входящие в набор, хранятся до удаления такого набора. В качестве значения срока действия можно задать количество дней, на протяжении которых должен храниться снимок.

Для идентификации базовой конфигурации используется имя, указанное пользователем. Выполните процедуру `CREATE_BASELINE`, чтобы создать базовую конфигурацию из набора снимков, и укажите имя и пару идентификаторов снимков. Идентификатор базовой конфигурации, который уникален за весь период существования базы данных, назначается недавно созданной базовой конфигурации. Как правило, базовые конфигурации создаются для прошедших репрезентативных периодов в целях сравнения с текущим состоянием системы. Кроме того, в средстве управления Database Control можно установить предупреждения о достижении пороговых значений на основе использования базовых конфигураций. Чтобы задать срок действия, укажите количество дней для параметра срока действия этой процедуры. По умолчанию используется значение `NULL`, то есть «без срока действия».

Идентификаторы снимков `snap_id` можно непосредственно получить с помощью `DBA_HIST_SNAPSHOT` или средства управления Database Control.

Примечание. Дополнительные сведения о пакете `DBMS_WORKLOAD_REPOSITORY` см. в руководстве *Oracle Database PL/SQL Packages and Types Reference*.

Диспетчер Enterprise Manager и репозиторий AWR



ORACLE

© Oracle, 2009. Все права защищены.

Диспетчер Enterprise Manager и репозиторий AWR

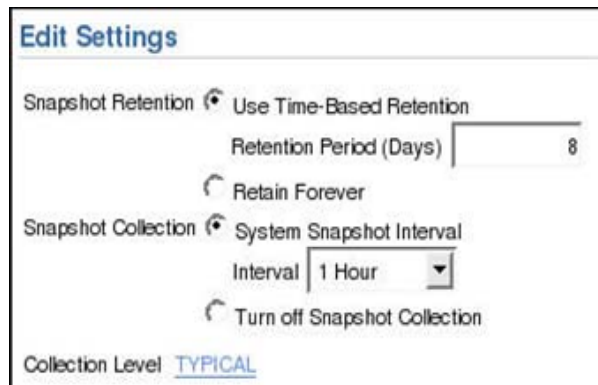
Перейдите на вкладку «Server» (Сервер) и в разделе «Statistics Management» (Управление статистикой) щелкните пункт «Automatic Workload Repository» (Автоматический репозиторий рабочей нагрузки). На странице «Automatic Workload Repository» (Автоматический репозиторий рабочей нагрузки) нажмите кнопку «Edit» (Редактировать) для изменения настроек.

На странице «Automatic Workload Repository» (Автоматический репозиторий рабочей нагрузки) можно выполнять следующие действия:

- редактировать настройки репозитория рабочей нагрузки;
- просматривать подробные сведения о созданных снимках и вручную создавать новые снимки;
- создавать базовые конфигурации репозитория AWR;
- формировать отчет по репозиторию AWR.

Управление репозиторием AWR

- Период удержания
 - По умолчанию: восемь дней
 - Оцените потребности в хранении
- Интервал сбора
 - По умолчанию: 60 минут
 - Оцените потребности в хранении и влияние на производительность
- Уровень сбора
 - **Basic** (большая часть функциональных возможностей ADDM отключена)
 - Typical (рекомендуется)
 - All (в снимки включается дополнительная информация для настройки подсистемы SQL)



ORACLE

© Oracle, 2009. Все права защищены.

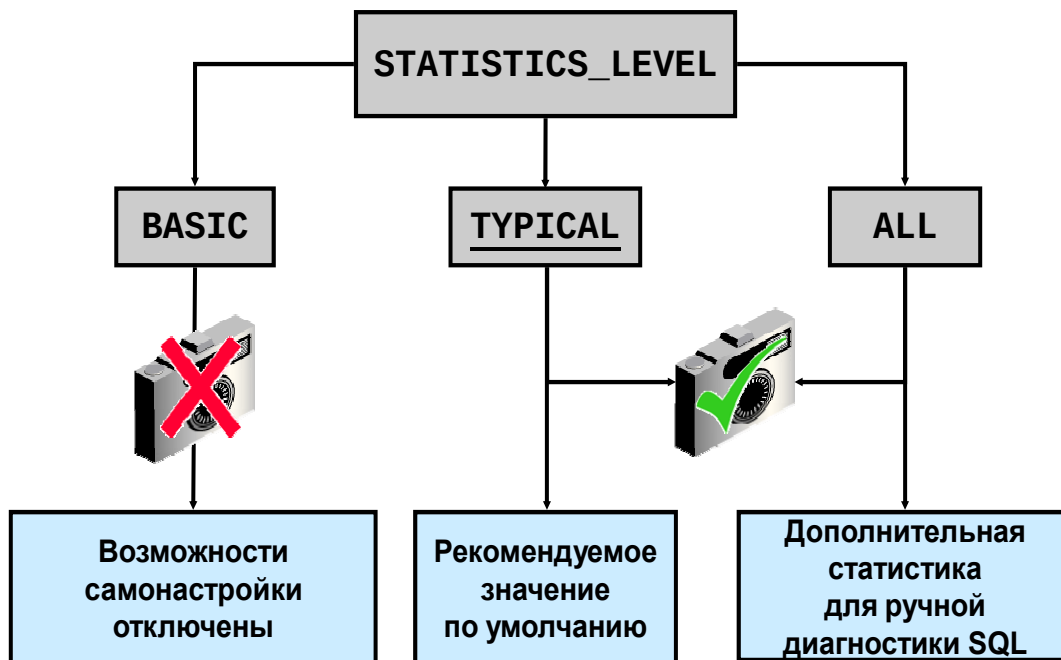
Управление репозиторием AWR

К настройкам репозитория AWR относятся период хранения, интервал сбора и уровень сбора. Примите к сведению, что уменьшение указанных значений повлияет на функциональность компонентов, зависящих от репозитория AWR, включая помощников.

Увеличение значений может привести к улучшению рекомендаций помощников, однако за счет увеличения дискового пространства для хранения снимков и за счет использования дополнительных ресурсов, используемых при сборе информации для снимков.

При настройке нового приложения, возможно, потребуется установить уровень сбора ALL. При установленном значении ALL производится сбор планов выполнения SQL и **хронометрические данные, которые могут повысить качество рекомендаций помощников по SQL**. После окончания настройки значение настройки следует установить равным TYPICAL.

Уровни статистики



ORACLE

© Oracle, 2009. Все права защищены.

Уровни статистики

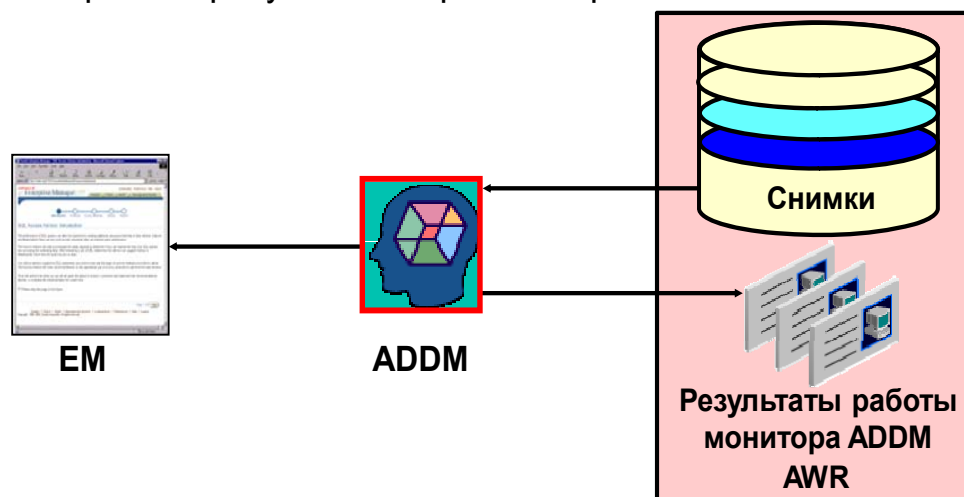
Параметр инициализации `STATISTICS_LEVEL` используется для управления сбором различной статистической информации и работой различных помощников, включая автоматические задачи обслуживания. Автоматические задачи обслуживания, кроме того, отвечают за сбор статистики оптимизатора. Параметру `STATISTICS_LEVEL` могут быть заданы следующие уровни:

- **BASIC:** расчет статистики репозитория AWR и показателей отключен. Автоматическое задание сбора статистики оптимизатора, все помощники и функция генерации предупреждений сервера отключены.
- **TYPICAL:** выполняется сбор основных статистических данных, необходимых для самообслуживания базы данных. Такие данные, как правило, требуются для наблюдения за работой базы данных Oracle. При этом выполняется автоматический сбор статистики для снижения вероятности неэффективного выполнения операторов SQL по причине устаревшей или некорректной статистики;
- **ALL:** выполняется сбор всей статистики. На этом уровне происходит регулярный сбор статистики операционной системы и статистики плана выполнения. В большинстве случаев указанные виды статистики не являются необходимыми для целей увеличения производительности, а иногда используются для специфических диагностических тестов.

Рекомендуется установить для параметра инициализации `STATISTICS_LEVEL` значение по умолчанию `TYPICAL`. Установка значения `BASIC` отключает автоматический сбор статистики оптимизатора.

Монитор автоматической диагностики баз данных (ADDM)

- Запускается после выполнения каждого снимка репозитория AWR.
- Используется для наблюдения за экземпляром. Определяет узкие места.
- Сохраняет результаты в репозитории AWR.



ORACLE

© Oracle, 2009. Все права защищены.

Монитор автоматической диагностики баз данных (ADDM)

В отличие от других помощников, монитор ADDM запускается автоматически после выполнения каждого снимка репозитория AWR. Каждый раз, когда делается снимок, монитор ADDM анализирует интервал между двумя последними снимками. Монитор ADDM выполняет упреждающее наблюдение за экземпляром и выявляет большинство узких мест до того момента, когда они станут проблемами.

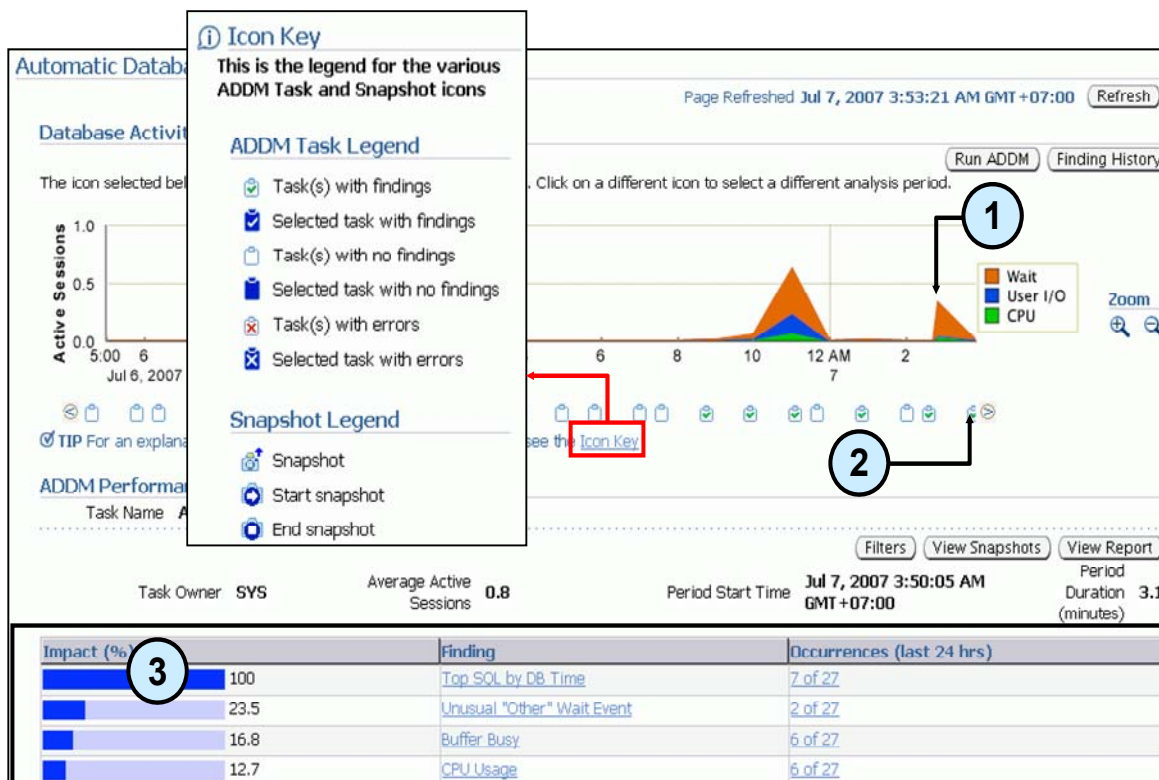
Во многих случаях монитор ADDM предоставляет рекомендации по решению обнаруженных проблем и даже рассчитывает выгоды от рекомендаций.

Некоторые общие проблемы определяются монитором ADDM:

- узкие места в работе процессора;
- управление плохим подключением Oracle Net;
- конфликты блокировок;
- скорость ввода/вывода (I/O);
- недостаточный размер структур памяти экземпляра базы данных;
- перегруженные операторы SQL;
- значительное время выполнения подсистем PL/SQL и Java;
- высокая загрузка при установке контрольной точки и ее причина (например, небольшие файлы журнала).

Результаты каждого анализа монитора ADDM сохраняются в репозитории AWR и могут быть изучены с помощью диспетчера Enterprise Manager.

Выводы монитора ADDM



ORACLE

© Oracle, 2009. Все права защищены.

Выводы монитора ADDM

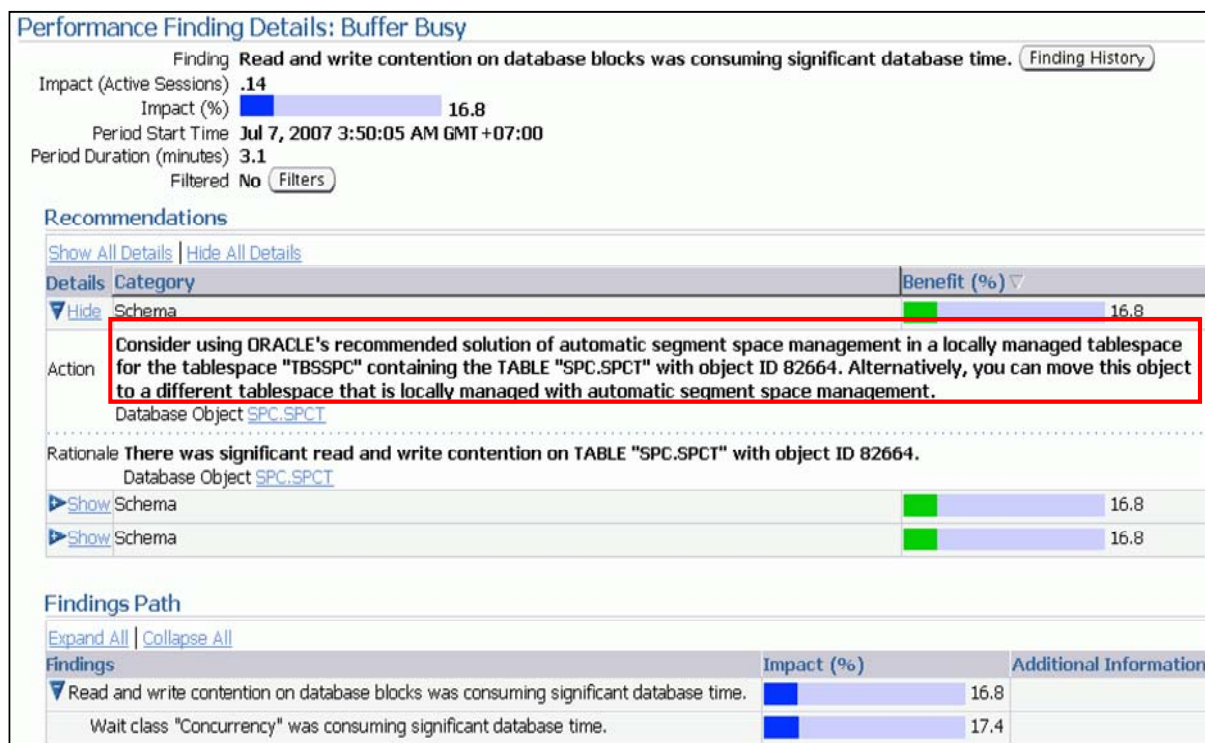
На странице монитора автоматической диагностики базы данных (ADDM) можно просмотреть подробные выводы, сделанные в ходе последнего запуска монитора ADDM. Время работы базы данных представляет собой сумму времени сеансов работы с базой за анализируемый период без учета простоев. Для каждого вывода отображается соответствующий процент влияния. Влияние представляет собой время, затраченное на выполнение соответствующей операции, по сравнению со временем работы базы данных за анализируемый период.

При рассмотрении данного примера обратите внимание на следующее:

1. Рисунок показывает, что среднее количество активных пользователей на данный момент времени в значительной степени увеличилось. Кроме того, существенной является проблема **Wait**.
2. Значок показывает, что выходные данные монитора ADDM, отображаемые в нижней части страницы, соответствуют данному моменту времени. Можно просмотреть результаты прошлых анализов с помощью соответствующих значков.
3. Выводы содержат краткий обзор того, что обнаружил монитор ADDM, и его рекомендации по настройке. При выборе какого-либо пункта открывается страница «Performance Finding Details» (Выводы о производительности).

Нажмите кнопку «View Report» (Просмотр отчета), чтобы просмотреть дополнительные сведения, полученные в результате анализа производительности, в виде текстового отчета.

Рекомендации монитора ADDM



ORACLE

© Oracle, 2009. Все права защищены.

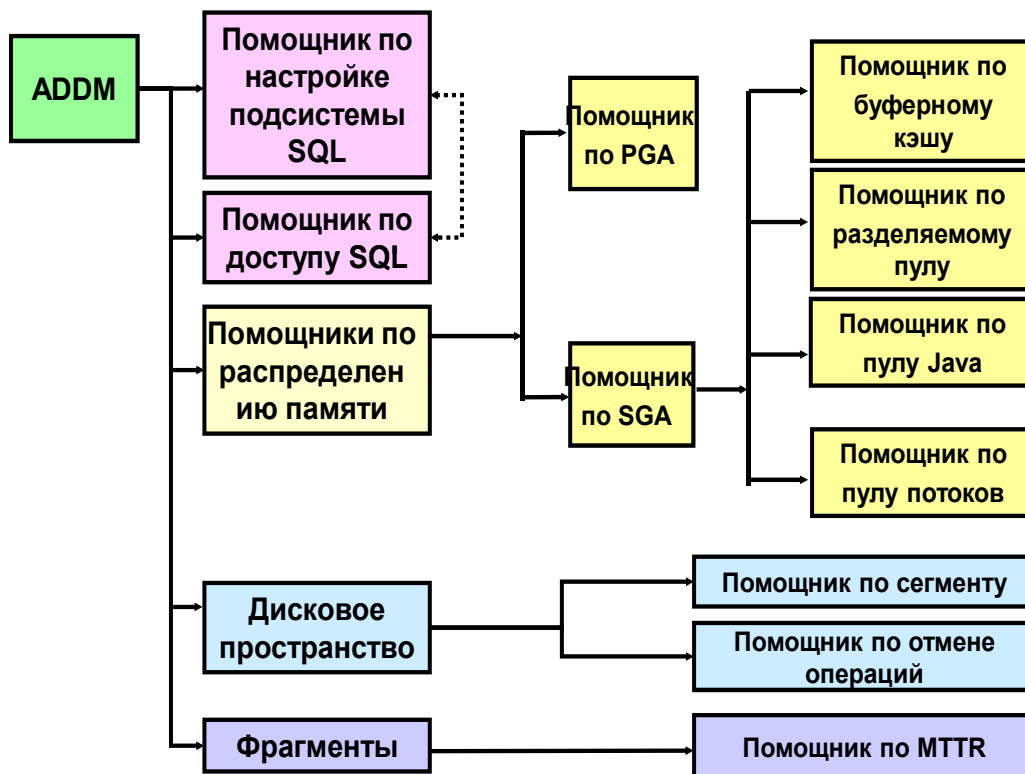
Рекомендации монитора ADDM

На странице «Performance Finding Details» (Выводы о производительности) приведены рекомендации по решению соответствующего вопроса. Рекомендации сгруппированы в различные категории, такие как «Schema» (схема), «SQL Tuning» (настройка SQL), «Database Configuration» (конфигурация базы данных) и т.д. В столбце **Benefit (%)** указывается максимальное уменьшение времени работы базы данных при реализации соответствующей рекомендации.

Монитор ADDM учитывает самые разные изменения в системе. В рекомендации монитора могут включаться:

- **Изменения в аппаратном обеспечении:** добавление процессоров либо изменения в конфигурации подсистемы ввода/вывода
- **Конфигурация базы данных:** изменения значений параметров инициализации
- **Изменения схемы:** хэш-секционирование таблицы или индекса либо переход к использованию системы автоматического управления пространством сегментов (ASSM)
- **Изменение в приложении:** использование кэша для последовательностей или использование связанных переменных
- **Использование прочих помощников:** запуск помощника по настройке SQL при перегруженных запросах SQL или помощника Segment Advisor для часто используемых объектов

Структура консультационной системы



ORACLE

© Oracle, 2009. Все права защищены.

Структура консультационной системы

Помощники предоставляют полезную обратную связь об использовании ресурсов и производительности соответствующих компонентов сервера. Например, помощник по распределению памяти сообщает рекомендованное значение параметра инициализации **MEMORY_TARGET**, который управляет общим количеством памяти, используемым экземпляром базы данных Oracle.

На основе данных, собранных в репозитории AWR, монитор ADDM позволяет базе данных Oracle выполнить диагностику собственной производительности и определить, как обнаруженные проблемы могут быть устранены. Монитор ADDM автоматически запускается после каждого сбора статистики AWR. Потенциально он может выполнить запуск других помощников.

Ниже перечислены основные преимущества, предоставляемые инфраструктурой помощников:

- Все помощники имеют унифицированный интерфейс.
- Все помощники используют общий источник данных и хранилище результатов путем использования репозитория рабочей нагрузки.

На слайде показаны не все помощники (например, отсутствует помощник по восстановлению данных и помощник по восстановлению SQL).

Монитор автоматической диагностики баз данных (ADDM)

Монитор ADDM является экспертной системой сервера, наблюдающей за производительностью базы данных каждые 60 минут. Он предназначен для раннего обнаружения возможных узких мест системы и формирования рекомендации по их исправлению, прежде чем производительность системы заметно ухудшится.

Структура консультационной системы (продолжение)

Помощники по распределению памяти

Помощник по распределению памяти на самом деле представляет собой группу из нескольких консультационных функций, которые помогают определить оптимальные настройки для общей памяти, используемой экземпляром базы данных. Глобальная системная область (SGA) содержит несколько помощников для разделяемого пула, буферного кэша базы данных, пула Java и пула потоков. Помощники по пулу Java и по пулу потоков не представлены на странице «Memory Advisor» (Помощник по распределению памяти) диспетчера EM. Здесь есть помощник, отвечающий за глобальную область программ (PGA). В дополнение к консультационным функциям в данном помощнике содержится центральный пункт управления большим пулом и пулом Java.

Помощник по среднему времени восстановления (MTTR)

Используя помощник по MTTR, можно установить необходимое время восстановления базы данных после аварийного завершения работы экземпляра.

Помощник по сегменту

Данный помощник выполняет поиск таблиц и индексов, занимающих больше места, чем требуется. Помощник проверяет эффективность использования места на уровне табличного пространства или схемы и создает сценарии, предназначенные для сокращения занимаемого пространства, где это возможно.

Помощник по доступу SQL

Этот помощник анализирует все операторы SQL, выполненные за данный период времени, и предлагает создание дополнительных индексов или материализованных представлений для улучшения производительности.

Помощник по настройке подсистемы SQL

Этот помощник анализирует отдельные операторы SQL и выдает рекомендации для улучшения их производительности. Рекомендации могут включать в себя изменение самого оператора, изменение конфигурации экземпляра либо добавление индексов. Помощник по настройке подсистемы SQL не может быть запущен непосредственно. Он вызывается другими инструментами (такими как Top SQL (основные SQL) и Top Sessions (основные сеансы)) для помощи при оптимизации сложных операторов SQL.

Помощник по управлению отменой операций

Используя помощник по управлению отменой операций, можно определить размер табличного пространства для отмены операций, соответствующий заданному периоду хранения. Управление отменой операций и использование данного помощника описывается в уроке «Управление данными об отмене операций».

Помощник по восстановлению данных Data Recovery Advisor

Данный помощник выполняет автоматическую диагностику постоянных сбоев данных, предоставляет варианты восстановления и выполняет восстановление по запросу пользователя. Помощник по восстановлению данных предназначен для сокращения среднего времени восстановления (MTTR) и предоставляет централизованное средство для автоматического восстановления данных.

Помощник по восстановлению SQL

Помощник по восстановлению SQL используется после возникновения критической ошибки в операторе SQL, в результате чего в автоматическом репозитории диагностики регистрируется проблема. Помощник выполняет анализ оператора и в большинстве случаев рекомендует использование для него исправления. При реализации рекомендации использованное исправление SQL обходит сбойный участок, предлагая оптимизатору запросов использовать в дальнейшем альтернативный план выполнения. Это все происходит без изменения самого оператора SQL.

Диспетчер Enterprise Manager и помощники

The screenshot displays the Oracle Advisor Central interface. The 'Advisors' tab is active, showing a list of advisory tasks. Below the list is a search section with filters for Advisory Type, Task Name, Advisor Runs, and Status. A table of results is partially visible, showing an ADDM task. An inset window shows the 'Checkers' tab, displaying a list of integrity checks.

ORACLE

© Oracle, 2009. Все права защищены.

Диспетчер Enterprise Manager и помощники

Страница «Advisor Central» (Центр рекомендаций) является главной страницей для всех помощников. Открыть данную страницу можно с помощью ссылки «Advisor Central» (Центр рекомендаций), расположенной в списке «Related Links» (Связанные ссылки) на домашней странице управления базой данных. Однако это место в средстве управления Database Control не единственное, откуда могут быть запущены помощники. Также возможен доступ к помощникам в некоторых контекстах.

На вкладке «Advisors» (Помощники) страницы «Advisor Central» (Центр рекомендаций) можно просмотреть список всех заданий помощников, которые зарегистрированы в репозитории рабочей нагрузки. Список можно отфильтровать по типу помощника и по предварительно заданным периодам времени.

На вкладке «Checkers» (Проверки) страницы «Advisor Central» (Центр рекомендаций) можно задать расписания выполнения различных проверок целостности базы данных. Можно просмотреть список всех выполняющихся проверок с сортировкой по имени, типу или периоду времени.

Некоторые помощники подробно описаны в уроках «Управление данными об отмене операций», «Управление производительностью» и «Понятия о резервном копировании и восстановлении».

Примечание. Чтобы изменить стандартный период окончания (в днях) для всех будущих заданий, перейдите на страницу «Change Default Parameters» (Изменение параметров по умолчанию). Также эту страницу можно использовать для изменения параметров некоторых важных помощников.

Пакет DBMS_ADVISOR

Процедура	Описание
CREATE_TASK	Создает новую задачу в репозитории
DELETE_TASK	Удаляет задачу из репозитория
EXECUTE_TASK	Запускает задачу
INTERRUPT_TASK	Приостанавливает выполняемую в текущий момент задачу
GET_TASK_REPORT	Создает и возвращает текстовый отчет об указанной задаче
RESUME_TASK	Возобновляет приостановленную задачу
UPDATE_TASK_ATTRIBUTES	Изменяет атрибуты задачи
SET_TASK_PARAMETER	Изменяет параметр задачи
MARK_RECOMMENDATION	Помечает одну или несколько рекомендаций как принятые, отклоненные или проигнорированные
GET_TASK_SCRIPT	Создает сценарий для всех принятых рекомендаций

ORACLE

© Oracle, 2009. Все права защищены.

Пакет DBMS_ADVISOR

В пакете DBMS_ADVISOR содержатся все константы и объявления процедур для всех модулей помощников. Можно использовать данный пакет для выполнения задач из командной строки.

Чтобы выполнять процедуры помощника, необходимо обладать полномочиями ADVISOR. Полномочия ADVISOR предоставляют полный доступ к процедурам и представлениям помощника.

Примечание. Дополнительные сведения обо всех процедурах, имеющихся в пакете DBMS_ADVISOR, см. в руководстве *Oracle Database PL/SQL Packages and Types Reference*.

Экспресс-тест

Статистика оптимизатора `num_rows` всегда отражает действительное количество строк таблицы.

1. Да
2. Нет

ORACLE

© Oracle, 2009. Все права защищены.

Ответ: 2

Автоматические задачи обслуживания

Процесс автоматических задач обслуживания состоит из следующих этапов:

1. Открывается окно обслуживания.
2. **Фоновый процесс автоматических задач определяет расписание заданий.**
3. Планировщик запускает задания.
4. Диспетчер ресурсов ограничивает влияние заданий автоматизации.

Стандартные автоматические задания обслуживания:

- сбор статистики оптимизатора;
- автоматический помощник по сегменту;
- автоматический помощник по SQL.



ORACLE

© Oracle, 2009. Все права защищены.

Автоматические задачи обслуживания

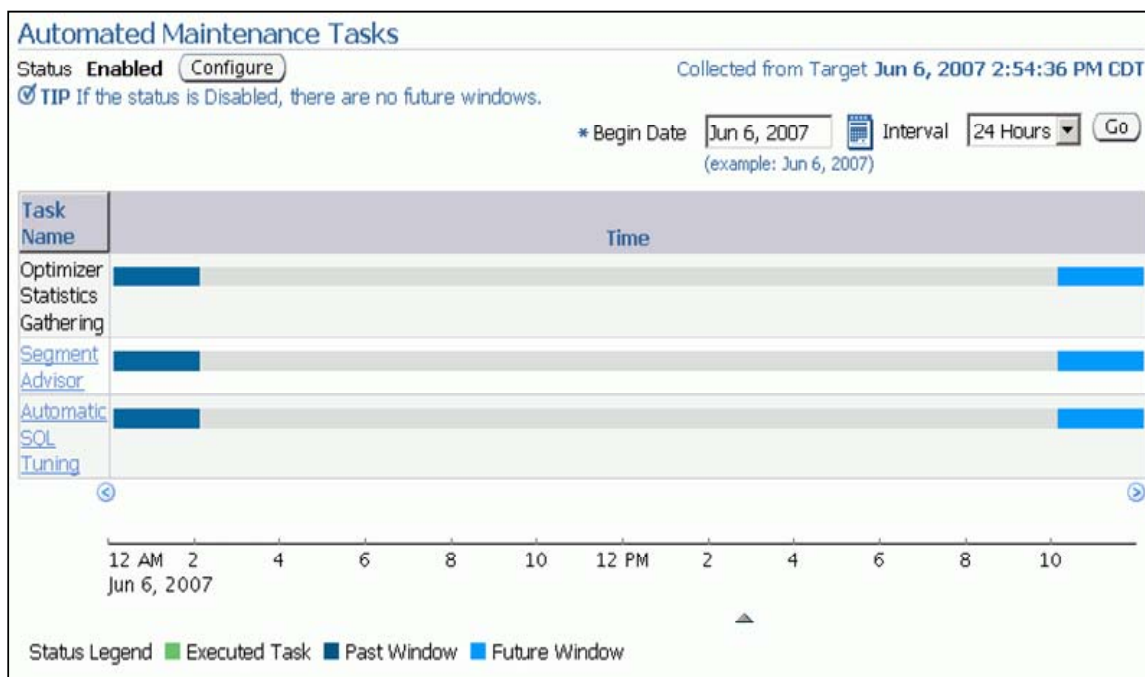
На основе анализа информации, хранящейся в репозитории AWR, база данных может определить необходимость выполнения задач по обслуживанию, например обновления статистики оптимизатора. Инфраструктура автоматических задач обслуживания позволяет базе данных Oracle автоматически выполнять такие операции. Для запуска задач в предварительно заданных окнах обслуживания используется планировщик.

По умолчанию ежедневные окна обслуживания начинаются в 22:00 часа и продолжаются 4 часа. По субботам и воскресеньям окно обслуживания начинается в 6:00 и продолжается 20 часов. Всем атрибутам окна обслуживания, включая время запуска и окончания, частоту, дни недели и др., можно задать собственные значения. В дополнение, влияние заданий автоматизированного обслуживания на обычные операции базы данных может быть ограничено путем привязки к данному окну обслуживания плана по использованию ресурсов диспетчера ресурсов базы данных.

Ниже приведены примеры обслуживания.

- Статистика оптимизатора автоматически обновляется с помощью инфраструктуры задач автоматизированного обслуживания.
- Автоматический помощник по сегментам имеет задачи по умолчанию, которые выполняются в окнах обслуживания.
- При создании базы данных с DBCA можно задать выполнение регулярного резервного копирования.

Автоматические задачи обслуживания



ORACLE

© Oracle, 2009. Все права защищены.

Автоматические задачи обслуживания (продолжение)

Щелкните ссылку «Automated Maintenance Tasks» (Автоматические задания обслуживания), расположенную под заголовком «Scheduler» (Планировщик) на странице «Server» (Сервер), чтобы открыть страницу «Automated Maintenance Task» (Автоматические задачи обслуживания), на которой можно просмотреть расписание запуска задач и историю последних запусков. На этой странице можно получить детальные сведения о некоторых заданиях. Щелкните «Configure» (Конфигурация) для перехода на страницу «Automated Maintenance Tasks Configuration» (Конфигурация автоматических задач обслуживания). Задача выполняется в определенном интервале времени (окне). На рисунке показано последнее окно, в котором выполнялась задача, и следующее окно по расписанию, в котором будет выполняться задача.

Примечание. В примере для задач показаны окна по умолчанию. Когда окно обслуживания закрывается, планировщик останавливает задачу по сбору статистики оптимизатора. Обработка оставшихся объектов будет выполнена в следующем окне обслуживания.

Конфигурация автоматических задач обслуживания

Automated Maintenance Tasks Configuration

Global Status ☒ Enabled ☐ Disabled

Task Settings

Optimizer Statistics Gathering ☒ Enabled ☐ Disabled [Configure](#)

Segment Advisor ☒ Enabled ☐ Disabled

Automatic SQL Tuning ☒ Enabled ☐ Disabled [Configure](#)

Maintenance Window Group Assignment [Edit Window Group](#)

Window	Optimizer Statistics Gathering	Segment Advisor	Automatic SQL Tuning
	Select All Select None	Select All Select None	Select All Select None
WEDNESDAY WINDOW	☒	☒	☒
THURSDAY WINDOW	☒	☒	☒
FRIDAY WINDOW	☒	☒	☒
SATURDAY WINDOW	☒	☒	☒
SUNDAY WINDOW	☒	☒	☒
MONDAY WINDOW	☒	☒	☒
TUESDAY WINDOW	☒	☒	☒

[Show SQL](#) [Revert](#) [Apply](#)

ORACLE

© Oracle, 2009. Все права защищены.

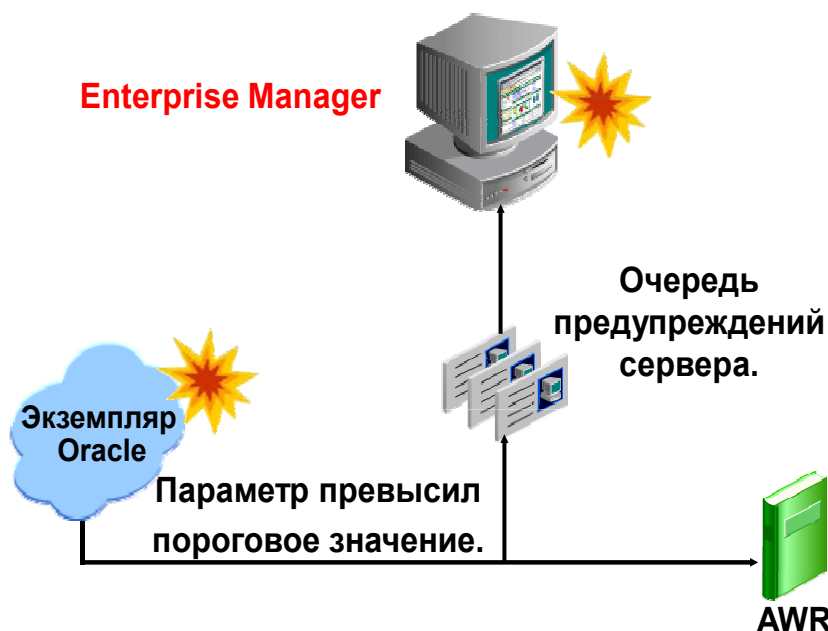
Конфигурация автоматических задач обслуживания

На странице «Automated Maintenance Tasks Configuration» (Конфигурация автоматических задач обслуживания) можно разрешить либо запретить выполнение автоматических задач обслуживания — все вместе, каждую по отдельности или относящиеся к определенным окнам. Также можно выполнить настройку параметров, используемых для сбора статистики оптимизатора, и параметров управления заданиями для помощника по настройке подсистемы SQL.

Выберите окно, чтобы просмотреть или изменить его расписание.

Чтобы добавить окно в группу окон или удалить окно из группы, щелкните «Edit Window Group» (Изменить группу окон).

Предупреждения сервера



ORACLE

© Oracle, 2009. Все права защищены.

Предупреждения сервера

Предупреждения представляют собой уведомления о том, что база данных находится в нежелательном состоянии и требует внимания. По умолчанию база данных Oracle передает предупреждения через средство управления Database Control диспетчера Enterprise Manager. Можно настроить диспетчер Enterprise Manager таким образом, чтобы предупреждения о проблемах отправлялись администратору по электронной почте или отображались в консоли.

Кроме того, можно установить пороговые значения для большинства постоянных показателей системы. Oracle Database 11g заранее уведомляет, когда показатели базы данных начинают в значительной степени отличаться от нормальных значений и достигают заданных пороговых пределов. Раннее уведомление о потенциальных проблемах дает возможность быстрого реагирования и во многих случаях устранения проблем, прежде чем пользователи смогут что-либо заметить.

По умолчанию наблюдение выполняется за приблизительно 60-ю показателями, некоторые из них перечислены ниже:

- количество прерванных задач;
- время ожидания базы данных (%);
- используемая область дампа (%);
- время отклика SQL (%) по сравнению с базовой конфигурацией;
- используемое табличное пространство (%);
- общее количество инцидентов.

Несколько дополнительных ключевых показателей могут использоваться для раннего уведомления о проблемах:

- среднее время чтения файла (в сотых долях секунды);
- время отклика (за одну транзакцию);
- время ожидания (%).

Установка пороговых значений

Metric and Policy Settings Cancel OK

Metric Thresholds **Policies**

View: Metric Thresholds All metrics Metric Thresholds

Metric	Comparison Operator	Warning Threshold	Critical Threshold	Corrective Actions	Collection Schedule	Edit
Access Violation	Matches		1	None	Every 5 Minutes	
Access Violation Status	>		0	None	Every 5 Minutes	
Archive Area Used (%)	>	80		None	Every 15 Minutes	
Archiver Hung	Matches		1	None	Every 5 Minutes	
Archiver Hung Status	>		0	None	Every 5 Minutes	
Audited User	=	SYS		None	Every 15 Minutes	
Average Users Waiting Count						
Administrative	>	10		None		
Application	>	10		None		
Cluster	>	30		None		
Commit	>	30		None		
Concurrency	>	10		None		
Configuration	>	10		None		
Network	>	10		None		

ORACLE

© Oracle, 2009. Все права защищены.

Установка пороговых значений

Чтобы установить или изменить пороговое значение для всей базы данных, щелкните ссылку «Metric and Policy Settings» (Установка показателей и политик) в области «Related Links» (Связанные ссылки), расположенной на домашней странице базы данных. Введите текст предупреждения и критическое пороговое значение. Заданное предупреждение будет выдано по достижении базой данных установленных значений.

Уже установленные пороговые значения можно просмотреть в списке «Metrics with thresholds» (Показатели с установленными порогами). По умолчанию приблизительно 60 показателей имеют предварительно заданные пороговые значения. В списке «All metrics» (Все показатели) приведены показатели, для которых пороговые значения не установлены.

Щелкните один из значков редактирования, чтобы открыть страницу, на которой можно задать дополнительные корректирующие действия при возникновении предупреждения либо при достижении пороговых значений.

Выберите ссылку «Collection Schedule» (Расписание сбора) и установите расписание интервала сбора. Помните о том, что каждое запланированное задание влияет на какую-то группу показателей.

Создание и тестирование предупреждения

1. Укажите пороговое значение.
2. Создайте тестовую ситуацию.
3. Проверьте выдачу предупреждения.

Edit Advanced Settings: Tablespace Space Used (%)

Cancel Continue

Monitored Objects

The table lists all Tablespace Name objects monitored for this metric. You can specify different threshold settings for each Tablespace Name object.

Add Reorder

Edit Remove

Tablespace Select Name	Comparison Operator	Warning Threshold	Critical Threshold	Corrective Action
All others	>=	70	75	None

Show SQL

Return

```
CREATE TABLE "HR"."FILLER" TABLESPACE "INVENTORY"
STORAGE ( INITIAL 8M) AS SELECT * FROM HR.EMPLOYEES
```

Alerts

Category All Go Critical 3 Warning 1

Severity	Category	Name	Impact	Message	Alert Triggered
x	Tablespaces Full	Tablespace Space Used (%)		Tablespace INVENTORY is 80 percent full	Jun 7, 2007 3:24:05 PM

ORACLE

© Oracle, 2009. Все права защищены.

Создание и тестирование предупреждения

Можно установить пороговые значения для определенного объекта.

Пример

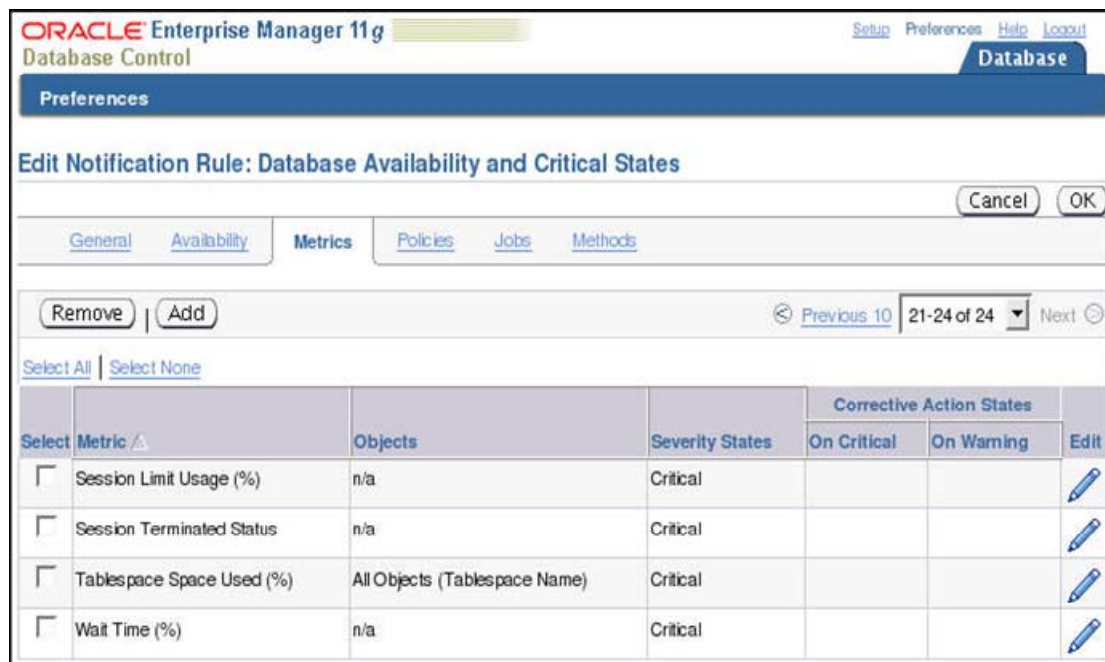
Предположим, что требуется получать важное предупреждение, если место, занимаемое табличным пространством **INVENTORY**, превышает 75 %. (Данное табличное пространство не позволяет входящим в него файлам данных автоматически увеличиваться в размере.)

Чтобы создать и протестировать предупреждение, выполните следующие шаги:

1. В диспетчере Enterprise Manager перейдите на страницу «Metrics and Policy Settings» (Настройки показателей и политик), а затем щелкните значок «Edit» (Редактировать) для порогового значения «Tablespace Used (%)» (Используемое табличное пространство (%)). Установите требуемое пороговое значение для данного табличного пространства.
2. Перейдите на вкладку «Schema» (Схема) на странице «Tables» (Таблицы) и создайте таблицу для тестирования предупреждения. Для создания копии существующей страницы используйте действие «Define using SQL» (Определить с помощью SQL). Начальное значение 8 Мбайт в предложении **STORAGE** приводит к тому, что для таблицы при создании отводится 80% от 10 Мбайт, занимаемых табличным пространством **INVENTORY**.
3. После получения сообщения об ошибке, уведомляющего о том, что размер таблицы не может быть увеличен, проверьте, есть ли соответствующее предупреждение на домашней странице базы данных. Значение показателя «Tablespace Space Used (%)» (Используемое табличное пространство (%)) обновляется по умолчанию каждые 10 минут.

Большинство предупреждений включают в себя название соответствующего помощника, к которому можно обратиться для получения более подробной рекомендации. Для каждого предупреждения средство управления базой данных Database Control предоставляет ссылку для запуска соответствующего помощника.

Уведомления о предупреждениях



ORACLE Enterprise Manager 11g Database Control

Preferences

Edit Notification Rule: Database Availability and Critical States

Cancel OK

General Availability Metrics Policies Jobs Methods

Remove Add Previous 10 21-24 of 24 Next

Select All Select None

Select	Metric	Objects	Severity States	Corrective Action States		Edit
				On Critical	On Warning	
☐	Session Limit Usage (%)	n/a	Critical			
☐	Session Terminated Status	n/a	Critical			
☐	Tablespace Space Used (%)	All Objects (Tablespace Name)	Critical			
☐	Wait Time (%)	n/a	Critical			

ORACLE

© Oracle, 2009. Все права защищены.

Уведомления о предупреждениях

В механизме уведомления используется пользовательский интерфейс диспетчера Enterprise Manager. Он основан на концепции правила уведомления, которое устанавливает соответствующий механизм уведомления для набора поступающих предупреждений.

Используя средство управления Database Control, можно изменить правила уведомления. Чтобы открыть страницу «General» (Общие), на которой можно указать адрес электронной почты для получения уведомлений, перейдите на домашнюю страницу и щелкните ссылку «Preferences» (Предпочтительные параметры).

На странице «General» (Общие) щелкните ссылку «Rules» (Правила), расположенную в области «Notification» (Уведомление). Выберите правило «Database Availability and Critical States» (Доступность базы данных и критические состояния) и нажмите кнопку «Edit» (Редактировать). Откроется страница «Edit Notification Rule Database Availability and Critical States» (Редактирование правила уведомления доступности базы данных и критических состояний). Перейдите на вкладку «Metrics» (Метрики) и укажите те метрики, для которых требуется получать уведомления.

Уведомления о предупреждениях (продолжение)

В качестве варианта можно указать, чтобы диспетчер Enterprise Manager выдавал непосредственное уведомление при возникновении определенных предупреждений. Например, если требуется получать уведомление о критических предупреждениях по электронной почте и установлено критическое пороговое значение для времени отклика системы для каждой вызываемой метрики, то такое почтовое уведомление может выглядеть следующим образом:

```
Host Name=mydb.us.mycompany.com
Metric=Response Time per Call
Timestamp=08-NOV-2005 10:10:01 (GMT -7:00)
Severity=Critical
Message=Response time per call has exceeded the threshold.
      See the latest ADDM analysis.
Rule Name= Rule
Owner=SYSMAN
```

В почтовом сообщении содержится ссылка на имя хоста и на результаты анализа монитора ADDM.

По умолчанию уведомление предоставляется для предупреждений о критичном состоянии (такие как «DB Down» (Остановка базы данных), «Generic Alert Log Error Status» (Состояние ошибки журнала общих предупреждений) и «Tablespace Used» (Используемое табличное пространство)). Однако, чтобы получать такие уведомления, необходимо ввести информацию об электронной почте:

1. На странице «Database Control» (Управление базой данных) щелкните ссылку в области заголовка и нижнего колонтитула.
2. На странице «Setup» (Настройка) выберите ссылку «Notification Methods» (Способы уведомлений).
3. Введите необходимую информацию в области «Mail Server» (Почтовый сервер), расположенной на странице «Notifications Methods» (Способы уведомлений).

Существуют другие способы уведомления, включая сценарии и прерывания протокола SNMP. Последний способ может использоваться для взаимодействия с приложениями сторонних производителей.

Чтобы получать уведомления:

1. На странице «Database Control» (Управление базой данных) щелкните ссылку «Preferences» (Предпочтения) в области заголовка и нижнего колонтитула.
2. На странице «Preferences» (Предпочтения) выберите ссылку «General» (Общие). Введите электронный почтовый адрес в области «E-mail Addresses» (Адреса электронной почты).
3. При необходимости можно изменить правила уведомления (например, изменить состояние важности для получаемого уведомления). Для этого нажмите «Notification Rules» (Правила уведомления). Откроется страница «Notification Rules».

Примечание. Дополнительные сведения о настройке правил уведомления см. в документации по настройке *Oracle Enterprise Manager Advanced Configuration*.

Реагирование на предупреждения

- При необходимости следует собрать больше исходной информации (например, с помощью монитора ADDM или другого помощника).
- **Изучите критические ошибки.**
- Выполните мероприятия по устранению проблемы.
- Подтвердите получение предупреждений, которые не очищаются автоматически.



ORACLE

© Oracle, 2009. Все права защищены.

Реагирование на предупреждения

При получении предупреждения следуйте полученным рекомендациям. В противном случае можно запустить монитор ADDM (или другой соответствующий помощник) для получения более подробной диагностической информации о системе или поведении объекта.

Предупреждения и инциденты генерируются для критичных ошибок. Критичные ошибки обычно создают инциденты, из которых формируются проблемы. Используйте рабочий интерфейс поддержки для изучения и возможного извещения о возникшей проблеме службы поддержки Oracle.

Большинство предупреждений, такие как «Out of Space» (Недостаток места), автоматически очищаются при устранении причины проблемы. Однако прочие предупреждения, например «Generic Alert Log Error» (Ошибка журнала общих предупреждений), отправляются для уведомления пользователя и должны быть подтверждены. После осуществления необходимых мероприятий по устранению проблемы следует подтвердить предупреждение путем его очистки или удаления. При очистке предупреждение перемещается в историю предупреждений, которую можно просмотреть, выбрав на домашней странице соответствующую ссылку из меню Related Links (связанные ссылки). При удалении предупреждение удаляется из истории предупреждений.

Чтобы очистить такое предупреждение, как «Generic Alert Log Error» (Общая ошибка журнала предупреждений), щелкните ссылку «Alert Log» (Журнал предупреждений), расположенную на домашней странице в разделе Diagnostic Summary (сводные данные диагностики). Откроется страница «Alert Log Errors» (Ошибки журнала предупреждений). Выберите предупреждение, которое необходимо очистить, а затем нажмите кнопку «Clear» (Очистить). Чтобы удалить предупреждение, выберите его и нажмите кнопку «Purge» (Удалить). Кроме того, также можно нажать кнопку «Clear Every Open Alert» (Очистить все открытые предупреждения) или кнопку «Purge Every Alert» (Удалить все предупреждения).

Типы предупреждений и их очистка



ORACLE

© Oracle, 2009. Все права защищены.

Типы предупреждений и их очистка

Существует два вида предупреждений, генерируемых сервером: связанные с пороговыми значениями и не связанные с ними.

Настройка большинства предупреждений, генерируемых сервером, выполняется путем установки текста предупреждения и критичных пороговых значений для показателей базы данных. Пороговые значения можно определить для более 120 показателей, включая следующие:

- физические чтения за сек.;
- пользовательские фиксации за сек.;
- время отклика службы SQL.

За исключением показателя «Tablespace Space Usage» (Использование места табличным пространством), который связан с базой данных, прочие показатели связаны с экземпляром. Предупреждения о пороговых значениях также называются *предупреждения состояний*, которые автоматически очищаются при устранении причины возникновения предупреждения. Предупреждения состояний отображаются в DBA_OUTSTANDING_ALERTS, а после очистки — в DBA_ALERT_HISTORY.

Другие предупреждения, генерируемые сервером, соответствуют определенным событиям базы данных, например ошибкам ORA-*, ошибкам «Snapshot too old» (Снимок слишком старый), «Recovery Area Low On Free Space» (Недостаток свободного места для области восстановления) и «Resumable Session Suspended» (Возобновляемый сеанс приостановлен). Эти не связанные с пороговыми значениями предупреждения называются *предупреждения без состояния*. Такие предупреждения без состояния сразу помещаются в журнал истории. Очистка предупреждений без состояния имеет смысл только в среде средства управления Database Control, поскольку предупреждения без состояния хранятся в его собственном репозитории.

Экспресс-тест

Предупреждения без состояния, такие как SNAPSHOT TOO OLD, можно найти в представлении словаря DBA_OUTSTANDING_ALERTS.

1. Да
2. Нет

ORACLE

© Oracle, 2009. Все права защищены.

Ответ: 2

Заключение

В ходе прохождения урока вы научились:

- управлять статистикой оптимизатора;
- управлять автоматическим репозиторием рабочей нагрузки (AWR);
- использовать монитор автоматической диагностики базы данных (ADDM);
- описывать консультационную систему;
- устанавливать пороговые значения предупреждений;
- использовать предупреждения, генерируемые сервером;
- использовать автоматические задачи.

ORACLE

© Oracle, 2009. Все права защищены.

Обзор упражнения 12: упреждающее обслуживание

Целью данного практического занятия является упреждающее управление базой данных с помощью монитора ADDM, включая:

- **выполнение анализа проблемы;**
- проверку производительности базы данных;
- реализацию решения.

ORACLE

© Oracle, 2009. Все права защищены.

13

Управление производительностью

ORACLE

© Oracle, 2009. Все права защищены.

Цели

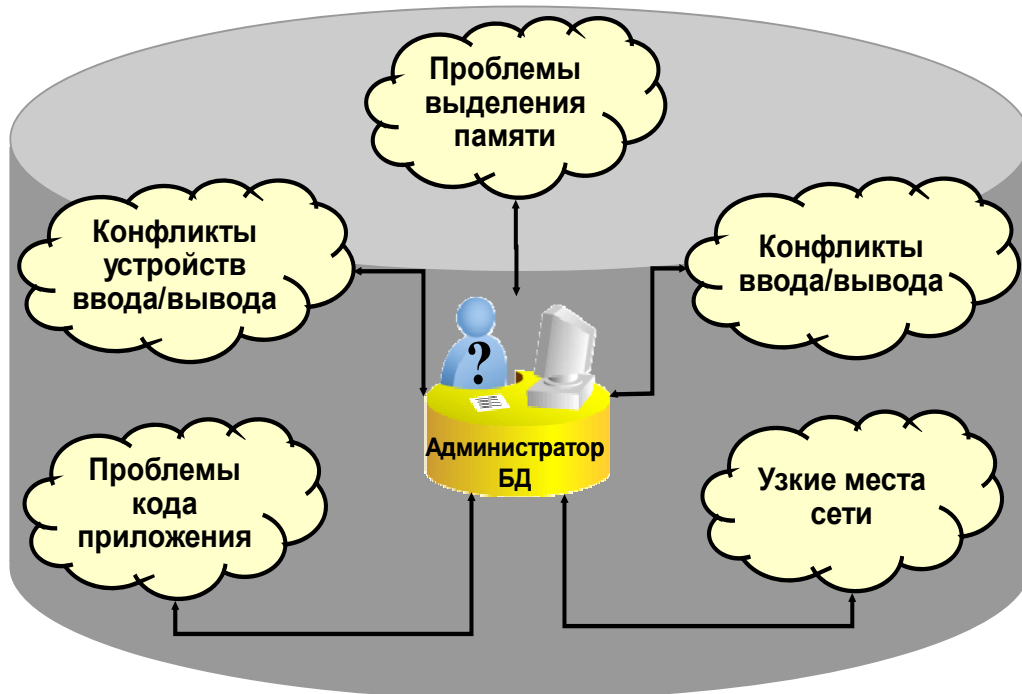
По завершении данного урока вы сможете:

- использовать Enterprise Manager для контроля производительности;
- **использовать функцию автоматического управления памятью (АММ);**
- использовать помощник по распределению памяти для определения размера буферов памяти;
- просматривать динамические представления данных **производительности;**
- устранять неполадки, связанные с недопустимыми и непригодными к использованию объектами.

ORACLE

© Oracle, 2009. Все права защищены.

Контроль производительности



ORACLE

© Oracle, 2009. Все права защищены.

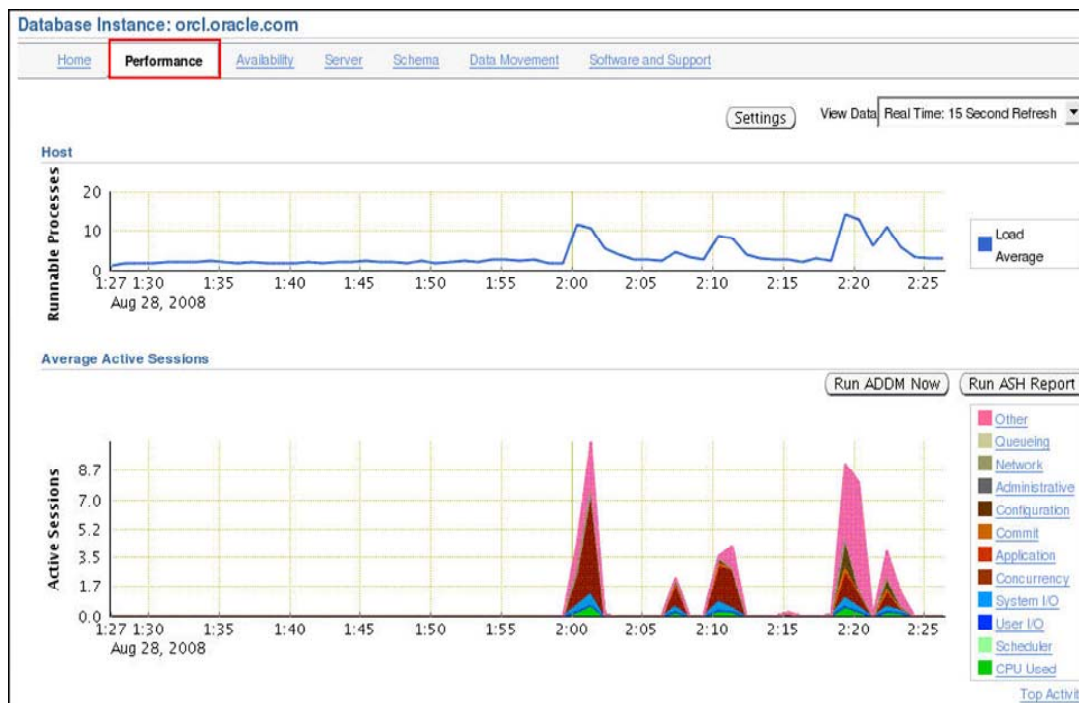
Контроль производительности

Для администрирования базы данных Oracle Database 11g и обеспечения ее успешной работы администратор базы данных (DBA) должен осуществлять регулярный контроль за производительностью, чтобы выявлять «узкие» места и исправлять проблемные области.

DBA может просматривать сотни показателей производительности, начиная с сетевой производительности и скорости дискового ввода/вывода и заканчивая временем, затраченным на выполнение операций отдельного приложения. Эти показатели производительности обычно называются *метриками базы данных*.

Примечание. Дополнительные сведения о производительности базы данных Oracle см. в курсе *Oracle Database 11g: настройка производительности*.

Страница производительности Enterprise Manager



ORACLE

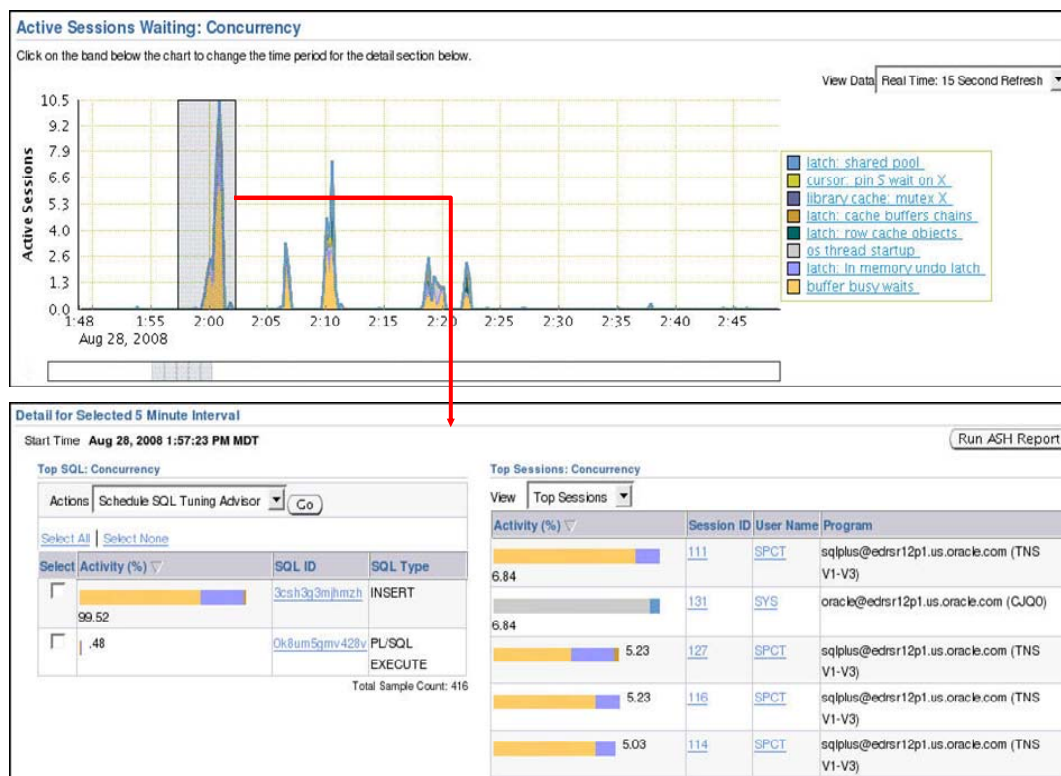
© Oracle, 2009. Все права защищены.

Страница производительности Enterprise Manager

В Enterprise Manager страница производительности открывает доступ к мощному набору инструментов для контроля и настройки производительности. Первый набор диаграмм на этой странице предоставляет обобщенные данные о процессах и действиях активных сеансов. На диаграмме «Average Active Sessions» (Среднее число активных сеансов) показаны уровень использования ЦПУ и ресурсы, вызывающие появление наибольшего количества событий ожидания.

На слайде можно увидеть, что недавно повысился уровень параллелизма и других событий ожидания. При каждом этом резком повышении также наблюдалось незначительное увеличение использования системного ввода/вывода и ЦП. Чтобы получить дополнительные сведения об ожиданиях, щелкните соответствующую категорию. Данные ввода/вывода разделяются по типам ввода/вывода (например, чтение файла журнала, запись в управляющий файл и т. п.).

Детализация до определенной категории событий ожидания



ORACLE

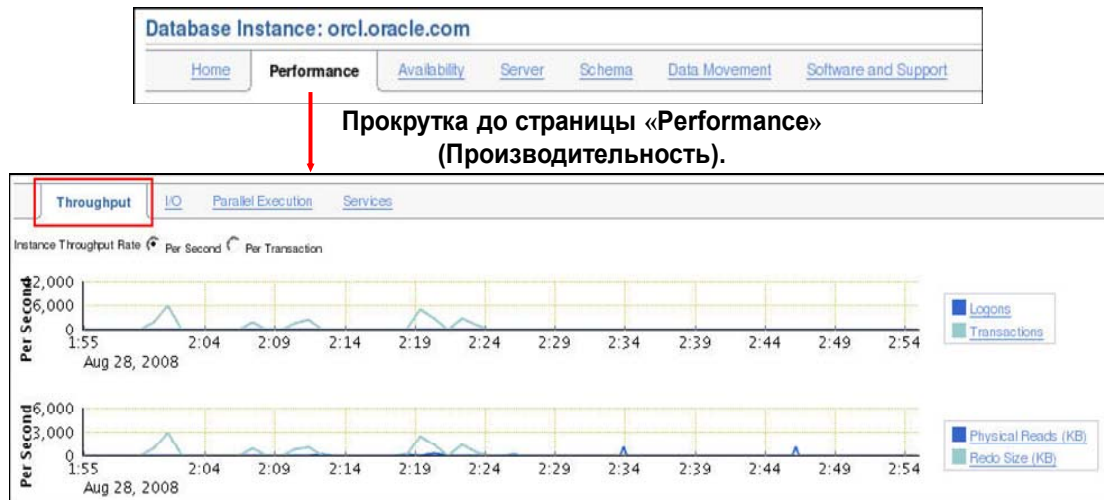
© Oracle, 2009. Все права защищены.

Детализация до определенной категории событий ожидания

При перемещении вниз по иерархии к определенной категории событий ожидания можно просматривать сведения по конкретному пятиминутному интервалу, а также «Top Working SQL» (Основные работающие операторы SQL) и «Top Working Sessions» (Основные работающие сеансы), связанные с конкретным событием ожидания для данного отрезка времени. Это позволяет выполнять последующий анализ замедлений в работе системы и определять потенциальные причины.

В примере на слайде показаны результаты детализации категории параллелизма из графика активного сеанса на предыдущем слайде.

Страница производительности: производительность



ORACLE

© Oracle, 2009. Все права защищены.

Страница производительности: производительность

Диаграммы пропускной способности экземпляра и операций ввода/вывода для диска экземпляра можно просмотреть на вкладках «Throughput» (Пропускная способность) и «I/O» (Ввод/вывод) на странице «Performance» (Производительность). На данном слайде открыта вкладка «Throughput» (Пропускная способность).

Контроль производительности: основные сеансы

Top Consumers

Overview Top Services Top Modules Top Actions Top Clients **Top Sessions**

Collected From Jul 31, 2009 8:40:17 AM GMT+07:00 To Jul 31, 2009 8:40:32 AM GMT+07:00

Kill Session View Disable SQL Trace Enable SQL Trace Show Active SQL Customize

Select	SID	DB User	CPU (1/100 sec)	PGA Memory (bytes)	Physical Reads	Logical Reads	Hard Parses	Total Parses	Disk Sorts	Status	Program	Module	OS PID	Machine	OS User	SQL Trace
☒	36	INVENTORY	1	1540712	20	228	0	0	0	ACTIVE	sqlplus@edrsr12p1.us.oracle.com (TNS V1-V3)	SOL*Plus	18269	edrsr12p1.us.oracle.com	oracle	DISABLED
☐	59	DBSNMP	12	7168500	0	56	0	10	0	ACTIVE	emagent@edrsr12p1.us.oracle.com (TNS V1-V3)	emagent_SQL_oracle_database	9753	edrsr12p1.us.oracle.com	oracle	DISABLED
☐	42	CJQ0	0	1794548	0	6	0	0	0	ACTIVE	oracle@edrsr12p1.us.oracle.com (CJQ0)		13001	edrsr12p1.us.oracle.com	oracle	DISABLED
☐	41	DBSNMP	0	2384372	0	3	0	2	0	ACTIVE	emagent@edrsr12p1.us.oracle.com (TNS V1-V3)	emagent_AQMetrics	13286	edrsr12p1.us.oracle.com	oracle	DISABLED
☐	29	SYSMAN	0	2843124	0	1	0	5	0	ACTIVE	OMS	OEM.DefaultPool	12965	edrsr12p1.us.oracle.com	oracle	DISABLED
☐	49	DBSNMP	14	1532404	0	0	0	2	0	ACTIVE	OMS	Realtime Connection	18390	edrsr12p1.us.oracle.com	oracle	DISABLED
☐	34	SYSMAN	0	2384372	0	0	0	0	0	ACTIVE	OMS	OEM.SystemPool	13061	edrsr12p1.us.oracle.com	oracle	DISABLED
☐	30	SYSMAN	0	3105268	0	0	0	3	0	ACTIVE	OMS	OEM.DefaultPool	12967	edrsr12p1.us.oracle.com	oracle	DISABLED
☐	18	MMNL	0	1270260	0	0	0	0	0	ACTIVE	oracle@edrsr12p1.us.oracle.com (MMNL)		12861	edrsr12p1.us.oracle.com	oracle	DISABLED
☐	25	QMNC	0	745972	0	0	0	0	0	ACTIVE	oracle@edrsr12p1.us.oracle.com (QMNC)	STREAMS	12963	edrsr12p1.us.oracle.com	oracle	DISABLED
☐	57	J001	0	418292	0	0	0	0	0	ACTIVE	oracle@edrsr12p1.us.oracle.com (J001)		18451	edrsr12p1.us.oracle.com	oracle	DISABLED
☐	43	SMCO	0	418292	0	0	0	0	0	ACTIVE	oracle@edrsr12p1.us.oracle.com (SMCO)	KTSJ	13544	edrsr12p1.us.oracle.com	oracle	DISABLED

ORACLE

© Oracle, 2009. Все права защищены.

Контроль производительности: основные сеансы

В разделе «Additional Monitoring Links» (Дополнительные ссылки для контроля) щелкните элемент «Top Consumers» (Основные потребители), чтобы перейти на страницу ведущих потребителей.

На странице «Top Consumers Overview» (Обзор основных потребителей) графически отображаются:

- Основные службы
- Основные модули (по службам)
- Основные действия (по службам и модулям)
- Основные клиенты

На странице «Top Consumers» (Основные потребители) перейдите на вкладку «Top Sessions» (Основные сеансы), чтобы просмотреть критическую статистику сеансов, использующих наибольшее количество ресурсов:

- ЦПУ
- Память PGA
- Операции логического чтения
- Операции физического чтения
- Число аппаратных разборов
- Число сортировок

Щелкните имя столбца, чтобы выполнить сортировку результатов по значению в этом столбце.

В таблице на этой странице перечислены сеансы, отсортированные по операциям логического чтения. Согласно данной информации пользователь INVENTORY в сеансе 36 выполняет наибольшее количество операций логического чтения в данный момент.

Контроль производительности: основные службы

Overview Top Services Top Modules Top Actions Top Clients Top Sessions					
View: Active Services					
Enable SQL Trace Disable SQL Trace View SQL Trace File					
Select All Select None					
Select	Service	Activity (% for the last 5 minutes)	SQL Trace Enabled	Delta Elapsed Time (seconds)	Cumulative Elapsed Time (seconds)
☐	SYS\$USERS	42.9	FALSE	0	227
☐	SYS\$BACKGROUND	35.7	FALSE	0	0
☐	SH	14.3	FALSE	0	2
☐	SERV1	7.1	FALSE	0	2

Delta CPU Time (seconds)	Cumulative CPU Time (seconds)	Delta Physical I/O (blocks)	Cumulative Physical I/O (blocks)
0	0	0	16031
0	137	0	14414
0	1	15	637
0	2	0	12

ORACLE

© Oracle, 2009. Все права защищены.

Контроль производительности: основные службы

В многоуровневых системах с сервером приложений, объединяющим в пул соединения с базой данных, необходимые для анализа производительности сведения могут не предоставляться при просмотре сеансов. Группирование сеансов по именам служб позволяет осуществлять более точный контроль производительности.

В примере на слайде представлены четыре службы: SYS\$USERS, SYS\$BACKGROUND, SH и SERV1. Независимо от того, какой сеанс использовался для определенного запроса, если для соединения с БД использовалась одна из этих служб, данные производительности сеанса собираются под именем этой службы. Из отображенных сведений видно, что из двух служб приложений (SH и SERV1) служба SH была более активной в течение данного пятиминутного интервала.

Управление компонентами памяти

- **Автоматическое управление памятью (АММ):**
 - позволяет указать общий объем памяти, выделенный экземпляру (включая как SGA, так и PGA).
- **Автоматическое управление разделяемой памятью:**
 - позволяет указать общий объем SGA-памяти с помощью одного параметра инициализации;
 - позволяет серверу Oracle управлять объемом памяти, выделенным для разделяемого пула, пула Java, буферного кэша, пула потоков и большого пула.
- **Настройка управления разделяемой памятью вручную:**
 - задание размера компонентов с помощью нескольких отдельных параметров инициализации;
 - использование соответствующего помощника по распределению памяти для создания рекомендаций.

ORACLE

© Oracle, 2009. Все права защищены.

Управление компонентами памяти

В Oracle Database 11g можно указать общий объем памяти, выделенный экземпляру. При этом память будет динамически перераспределяться между глобальной системной областью (SGA) и глобальной областью программ (PGA) по мере необходимости. Данный способ называется автоматическим управлением памятью (АММ) и доступен только на платформах, поддерживающих динамическое освобождение памяти. Он упрощает выполнение задач, связанных с управлением памятью.

Также доступны помощники по распределению памяти, которые помогают задавать параметры инициализации на различных уровнях. Доступность того или иного помощника зависит от уровня, на котором задаются параметры памяти. Если включена функция АММ, доступен только помощник по размеру памяти.

Функция автоматического управления разделяемой памятью (ASMM) позволяет управлять областью SGA в целом.

Эта область состоит из нескольких компонентов. Размер большинства этих компонентов настраивается динамически для обеспечения наилучшей производительности в пределах заданных параметров инициализации. При включении АММ функция ASMM включается автоматически. Если функция ASMM включена, а АММ нет, доступен только помощник по размеру SGA.

Размером отдельных компонентов можно управлять вручную, задавая параметр инициализации отдельно для каждого компонента. Если сервер Oracle уведомляет об ошибке производительности, связанной с размером компонента SGA или PGA, можно воспользоваться помощником по распределению памяти, чтобы определить новые параметры для этих компонентов. Этот помощник способен моделировать результат применения новых параметров.

Включение автоматического управления памятью (АММ)

Database Instance: orcl.oracle.com > Advisor Central > Logged in As SYS

Memory Advisors

Page Refreshed August 28, 2008 3:08:57 PM MDT [Refresh] [Show SQL] [Revert] [Apply]

Чтобы включить автоматическое управление памятью, нажмите кнопку «Enable» (Включить).

Automatic Memory Management **Disabled** **Enable**

Memory Advisors

Page Refreshed August 28, 2008 3:08:57 PM MDT

When Automatic Memory Management is enabled, the database will automatically set the optimal distribution of memory will change from time to time to accommodate changes in the workload.

Automatic Memory Management **Enabled** [Disable]

Total Memory Size 556 MB [Advice]

Maximum Memory Size 1000 MB

The database must be restarted for changes to this value to take effect.

Вспользуйтесь помощником по размеру памяти.

Memory Size Advice

Improvement in DB Time (%)

Total Memory Size (MB)

Legend:

- Percentage improvement in DB Time for various sizes of Total Mem
- Total Memory Size
- Maximum Memory Size

Total Memory Size (MB) 556

You can click on the curve in the graph to set a new value. Total Memory Size cannot be greater than the Maximum Memory Size. First modify the Maximum Memory size (from the parent page) and then select a value of Total Memory up to the Maximum Memory size.

[Cancel] [OK]

ORACLE

© Oracle, 2009. Все права защищены.

Включение автоматического управления памятью (АММ)

Если функция автоматического управления памятью (АММ) не была включена в процессе настройки базы данных, ее можно включить следующим образом:

1. На домашней странице базы данных перейдите на вкладку «Server» (Сервер).
2. В области «Database Configuration» (Конфигурация базы данных) щелкните пункт «Memory Advisors» (Помощники по распределению памяти).
Откроется страница «Memory Advisors» (Помощники по распределению памяти).
3. Нажмите кнопку «Enable» (Включить) напротив пункта «Automatic Memory Management» (Автоматическое управление памятью).
Откроется страница включения автоматического управления памятью.
4. Укажите значения в полях «Total Memory Size» (Общий объем памяти) и «Maximum Memory Size» (Максимальный объем памяти) для автоматического управления памятью.
Примечание. При изменении максимального размера памяти необходимо перезапустить экземпляр базы данных.
5. Нажмите кнопку "OK".

Размер памяти можно увеличить позже, указав большее значение в поле «Total Memory Size» (Общий объем памяти) или для параметра инициализации MEMORY_TARGET. Однако нельзя указать большее значение, чем было указано в поле «Maximum Memory Size» (Максимальный объем памяти) или же для параметра MEMORY_MAX_TARGET.

Дополнительную информацию см. в руководстве *Oracle Database Administrator's Guide*.

После включения АММ станет доступен помощник по размеру памяти, с помощью которого можно настроить максимальный и планируемый размеры памяти.

Примечание. Oracle рекомендует использовать функцию АММ для упрощения выполнения задач по управлению памятью.

Включение автоматического управления разделяемой памятью (ASMM)



SGA PGA

The System Global Area (SGA) is a group of shared memory structures that contains data and control information for one Oracle database. The SGA is allocated in memory when an Oracle database instance is started.

Automatic Shared Memory Management **Disabled** **Enable**

Shared Pool 248 MB

Buffer Cache 136 MB

Large Pool 4 MB

Java Pool 12 MB

Other (MB) 5

Total SGA (MB) 405

Maximum SGA Size

The Maximum SGA Size specifies the maximum memory that the database may allocate. If you specify the Maximum SGA Size, you can later dynamically change SGA component sizes (provided the total SGA size does not exceed the Maximum SGA Size).

Maximum SGA Size (MB) 1000

The database must be restarted before any changes to this value take effect.

Legend for pie chart:

- Shared Pool (61%)
- Buffer Cache (33.5%)
- Java Pool (3%)
- Other (1.4%)
- Large Pool (1%)

ORACLE

© Oracle, 2009. Все права защищены.

Включение автоматического управления разделяемой памятью (ASMM)

Автоматическое управление разделяемой памятью включается автоматически при включении AMM. Если AMM или ASMM не были включены в процессе настройки базы данных, автоматическое управление разделяемой памятью можно включить следующим образом:

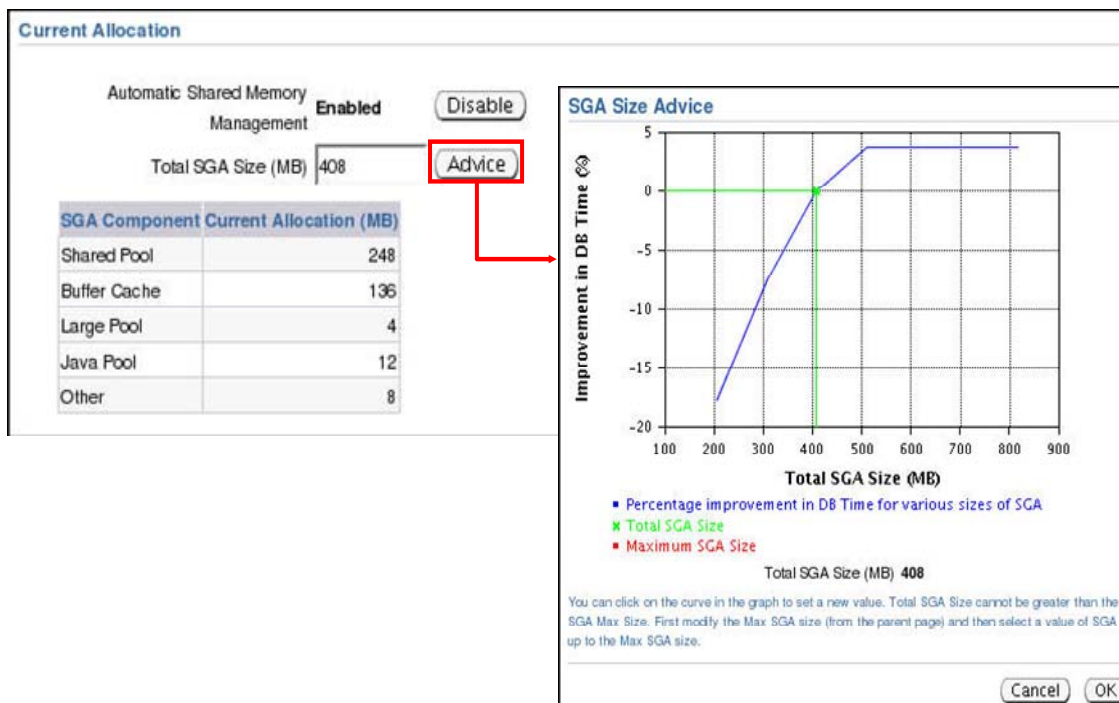
1. На домашней странице базы данных перейдите на вкладку «Server» (Сервер).
2. В области «Database Configuration» (Конфигурация базы данных) щелкните пункт «Memory Advisors» (Помощники по распределению памяти).
Откроется страница «Memory Advisors» (Помощники по распределению памяти).
3. Прокрутите страницу вниз до раздела SGA. Нажмите кнопку «Enable» (Включить) напротив пункта «Automatic Shared Memory Management» (Автоматическое управление разделяемой памятью).
Откроется страница включения автоматического управления разделяемой памятью.
4. Укажите общий размер SGA. Нажмите кнопку «OK».

Общий размер SGA можно увеличить позже, указав большее значение в поле «Total SGA Size» (Общий размер SGA) или для параметра инициализации `SGA_TARGET`. Однако нельзя указать значение, которое будет превышать значение поля «Maximum SGA Size» (Максимальный размер SGA) или параметра `SGA_MAX_SIZE`. Дополнительную информацию см. в руководстве *Oracle Database Administrator's Guide*.

Если функция AMM отключена, доступен помощник по PGA. Его рекомендуется использовать для настройки значения памяти PGA. Перейдите на вкладку PGA, чтобы открыть страницу свойств PGA. Нажмите кнопку «Advice» (Совет), чтобы вызвать помощник по PGA.

Примечание. Oracle рекомендует использовать функцию автоматического управления разделяемой памятью для упрощения выполнения задач по управлению памятью.

Помощник по автоматическому выделению разделяемой памяти



ORACLE

© Oracle, 2009. Все права защищены.

Помощник по автоматическому выделению разделяемой памяти

Если ASMM включена, не следует задавать параметры инициализации для определенных компонентов разделяемой памяти, которой управляет ASMM. После включения ASMM станет доступен помощник по размеру SGA, с помощью которого можно настроить оптимальное значение общего размера SGA.

Перед включением ASMM следует удалить индивидуальные параметры области памяти из SPFILE, поскольку при их задании возможно применение ограничений к ASMM. Если результаты выделения памяти с помощью ASMM для определенных компонентов требуют регулировки, значения для этих компонентов можно указать вручную. Если указанные значения меньше текущих значений, такие значения рассматриваются как минимальные **объемы памяти для соответствующих компонентов**. Если указанные значения больше текущих значений, размер компонентов памяти изменяется в соответствии с указанными значениями при условии наличия свободной памяти. При этом ограничивается объем памяти, доступной для автоматической настройки, но если для используемой среды требуется указать особое значение размера, которое не может быть настроено с помощью функции ASMM, то такая возможность будет предоставлена.

Необходимые параметры инициализации:

- SHARED_POOL_SIZE
- LARGE_POOL_SIZE
- JAVA_POOL_SIZE
- DB_CACHE_SIZE
- STREAMS_POOL_SIZE

Чтобы настроить эти параметры при включенной функции ASMM, используйте команду ALTER SYSTEM.

Динамическая статистика производительности

Для всей системы	Для сеанса	Для службы
V\$SYSSTAT • statistic# • name • class • value • stat_id	V\$SESSTAT • sid • statistic# • value	V\$SERVICE_STATS • service_name_hash • service_name • stat_id • stat_name • value
V\$SYSTEM_EVENT • event • total_waits • total_timeouts • time_waited • average_wait • time_waited_micro	V\$SESSION_EVENT • sid • event • total_waits • total_timeouts • time_waited • average_wait • max_wait • time_waited_micro • event_id	V\$SERVICE_EVENT • service_name • service_name_hash • event • event_id • total_waits • total_timeouts • time_waited • average_wait • time_waited_micro

Сводная статистика

События ожидания

ORACLE

© Oracle, 2009. Все права защищены.

Динамическая статистика производительности

Статистика служит для эффективной диагностики проблем производительности. Сервер Oracle создает множество типов статистики для различных уровней разрешения.

На уровне всей системы, сеанса и службы рассчитываются как события ожидания, так и сводная статистика. На слайде верхний ряд представлений — это сводная статистика, а нижний ряд — представления событий ожидания.

В процессе анализа проблемы производительности в любой из этих областей, как правило, обращается внимание на изменения в статистике (значение дельты) за конкретный период времени. Все возможные события ожидания заносятся в представление V\$EVENT_NAME. Все статистические данные собираются в представлении V\$STATNAME. В базе данных Oracle доступны около 480 типов статистики.

Динамическая статистика производительности (продолжение)

Динамическая статистика для всей системы

Пример

```
SQL> SELECT name, class, value FROM v$sysstat;  
NAME                                CLASS      VALUE  
-----
```

...		
table scans (short tables)	64	135116
table scans (long tables)	64	250
table scans (rowid ranges)	64	0
table scans (cache partitions)	64	3
table scans (direct read)	64	0
table scan rows gotten	64	14789836
table scan blocks gotten	64	558542
...		

Статистика для всей системы классифицируется по разделу настройки и цели отладки.

Существуют такие классы, как общая активность экземпляра, активность буфера журнала повторов, блокирование, активность буферного кэша базы данных и т. п.

Представления для устранения неполадок и настройки

<u>Экземпляр/база данных</u> V\$DATABASE V\$INSTANCE V\$PARAMETER V\$SPPARAMETER V\$SYSTEM_PARAMETER V\$PROCESS V\$BGPROCESS V\$PX_PROCESS_SYSSTAT V\$SYSTEM_EVENT	<u>Диск</u> V\$DATAFILE V\$FILESTAT V\$LOG V\$LOG_HISTORY V\$DBFILE V\$TEMPFILE V\$TEMPSEG_USAGE V\$SEGMENT_STATISTICS
<u>Память</u> V\$BUFFER_POOL_STATISTICS V\$LIBRARYCACHE V\$SGAINFO V\$PGASTAT	<u>Конфликты</u> V\$LOCK V\$UNDOSTAT V\$WAITSTAT V\$LATCH

ORACLE

© Oracle, 2009. Все права защищены.

Представления для устранения неполадок и настройки

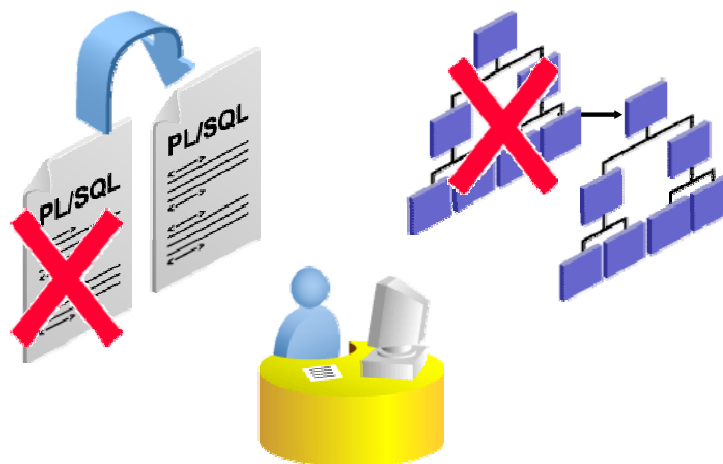
На данном слайде показаны некоторые представления, с помощью которых можно определить причину проблем производительности или проанализировать текущее состояние базы данных.

Полное описание этих представлений см. в руководстве *Oracle Database Reference*.

Недопустимые и неиспользуемые объекты

Влияние на производительность:

- объекты кода PL/SQL компилируются повторно;
- индексы восстанавливаются.



ORACLE

© Oracle, 2009. Все права защищены.

Недопустимые и неиспользуемые объекты

Недопустимые объекты PL/SQL и неиспользуемые индексы влияют на производительность. Недопустимые объекты PL/SQL необходимо скомпилировать повторно, прежде чем их можно будет использовать. Для этого время на компиляцию требуется добавить к первому действию, при котором выполняется попытка получить доступ к пакету, процедуре или функции PL/SQL. Если повторная компиляция PL/SQL не выполняется, операция завершается с ошибкой. Неиспользуемые индексы игнорируются оптимизатором. Если правильность выполнения оператора SQL зависит от индекса, который был помечен как неиспользуемый, производительность не улучшится до тех пор, пока индекс не будет восстановлен.

Недопустимые объекты PL/SQL: чтобы просмотреть текущее состояние объектов PL/SQL, выполните запрос словаря данных. Для поиска недопустимых объектов PL/SQL можно создать следующий запрос:

```
SELECT object_name, object_type FROM DBA_OBJECTS
WHERE status = 'INVALID';
```

По умолчанию метрика количества недопустимых объектов владельца проверяется каждые 24 часа. Если количество объектов для отдельного владельца превышает 2, выдается предупреждение.

Недопустимые и неиспользуемые объекты (продолжение)

При обнаружении объектов PL/SQL со статусом `INVALID` необходимо выяснить, был ли этот объект *когда-либо* допустимым (`VALID`). Зачастую разработчик приложения пренебрегает удалением кода, который не работает. Если недопустимость объекта PL/SQL вызвана ошибкой в коде, вы не сможете полноценно работать, пока эта ошибка не будет устранена. Если же процедура была допустимой в прошлом и стала недопустимой недавно, проблему можно устранить двумя описанными далее способами.

- Не предпринимайте никаких действий. Большая часть объектов PL/SQL автоматически компилируется повторно, если это необходимо при их вызове. При повторной компиляции таких объектов происходит небольшая задержка (в большинстве случаев она даже не заметна).

- Вручную перекомпилируйте недопустимый объект.

Недопустимые объекты PL/SQL можно перекомпилировать вручную с помощью Enterprise Manager или следующих команд SQL:

```
ALTER PROCEDURE HR.add_job_history COMPILE;
```

Перекомпиляция пакетов PL/SQL вручную требует выполнения двух действий:

```
ALTER PACKAGE HR.maintainemp COMPILE;
```

```
ALTER PACKAGE HR.maintainemp COMPILE BODY;
```

Неиспользуемые индексы: для поиска недопустимых индексов можно отправить запрос к представлению словаря данных `DBA_INDEXES`:

```
SELECT index_name, table_name FROM DBA_INDEXES  
WHERE status = 'UNUSABLE';
```

Состояние секционированных индексов отображается в представлении `DBA_IND_PARTITIONS`.

Неиспользуемые индексы преобразуются в допустимые путем их восстановления и пересчета указателей. При восстановлении неиспользуемого индекса происходит воссоздание индекса в новом месте и удаление неиспользуемого индекса. Это можно сделать как с помощью Enterprise Manager, так и с помощью следующих команд SQL:

```
ALTER INDEX HR.emp_empid_pk REBUILD;
```

```
ALTER INDEX HR.emp_empid_pk REBUILD ONLINE;
```

```
ALTER INDEX HR.email REBUILD TABLESPACE USERS;
```

Если фразу `TABLESPACE` опустить, индекс восстанавливается в том же табличном пространстве, в котором он уже существует. Фраза `REBUILD ONLINE` позволяет пользователям продолжать обновление таблицы индексов во время выполнения процедуры восстановления. (Если ключевое слово `ONLINE` не используется, пользователям придется ждать завершения восстановления, прежде чем выполнять команду DML для данной таблицы. Если индекс непригоден к использованию, он не задействуется во время восстановления, даже если ключевое слово `ONLINE` используется.)

В Enterprise Manager для восстановления индекса `UNUSABLE` используется реорганизация.

Примечания. Процедура восстановления индекса требует наличия свободного пространства, доступного для восстановления. Прежде чем выполнить восстановление, убедитесь в наличии достаточного места. В Enterprise Manager требования, касающиеся пространства, проверяются автоматически.

Экспресс-тест

Функция автоматического управления памятью позволяет экземпляру Oracle перераспределять память из _____ в SGA.

1. Большой пул
2. Буфер журнала
3. PGA
4. Пул потоков

ORACLE

© Oracle, 2009. Все права защищены.

Ответ: 3

Экспресс-тест

Размер SGA_TARGET не может превышать размер

_____.

1. LOG_BUFFER
2. SGA_MAX_SIZE
3. STREAMS_POOL_SIZE
4. PGA_AGGREGATE_TARGET

ORACLE

© Oracle, 2009. Все права защищены.

Ответ: 2

Заключение

В ходе прохождения урока вы научились:

- использовать Enterprise Manager для контроля производительности;
- **использовать функцию автоматического управления памятью (АММ);**
- использовать помощник по распределению памяти для определения размера буферов памяти;
- просматривать динамические представления данных **производительности;**
- устранять неполадки, связанные с недопустимыми и непригодными к использованию объектами.

ORACLE

© Oracle, 2009. Все права защищены.

Обзор упражнения 13: контроль и улучшение производительности

В данном упражнении рассматриваются следующие темы:

- выявление и восстановление неиспользуемых индексов;
- **использование страницы производительности** в Enterprise Manager.

ORACLE

© Oracle, 2009. Все права защищены.

