

**Специальность 10.05.01 «Компьютерная безопасность»,
Специализация «Математические методы защиты информации»
Уровень высшего образования – специалитет**

Дисциплина: Основы построения защищенных баз данных

Лабораторная работа №2.

Администрирование Oracle. Управление структурами хранения базы данных.

1. Учебные цели:

- Отработать вопросы управления табличными пространствами Oracle.
- Освоить приемы создания и удаления табличных пространств, добавления файлов, увеличения объема табличного пространства, отслеживание ошибок в работе табличных пространств в журналах Allert Log.

2. Требования к результатам обучения основной образовательной программы, достигаемые при проведении лабораторной работы:

- Уметь использовать возможности современных систем для решения задач администрирования и защиты баз данных.
- Владеть средствами приложений СУБД Oracle для создания и удаления табличных пространств, добавления файлов, увеличения объема табличного пространства, отслеживание ошибок в работе табличных пространств в журналах Allert Log.

3. Перечень материально-технического обеспечения

ПЭВМ с проигрывателем виртуальных машин, виртуальная машина с установленной СУБД Oracle.

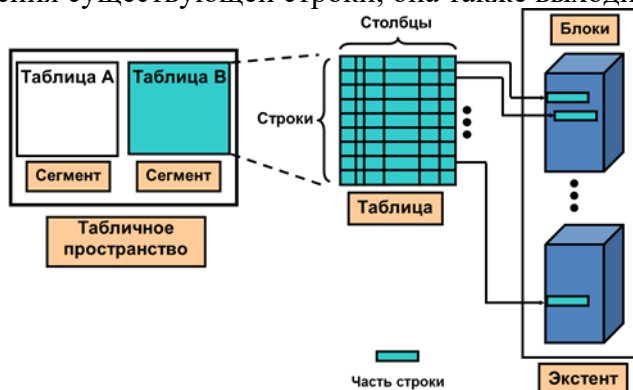
4. Краткие теоритические сведения и задания на исследование. Задания выделены рамками и синим шрифтом. Результаты лабораторной работы представляются в виде файла, содержащего копии экрана, показывающие этапы выполнения заданий.

1) Структуры хранения.

База данных разделена на логические единицы хранения, называемые *табличными пространствами (tablespaces)*. Каждое табличное пространство содержит множество логических блоков Oracle. Параметр DB_BLOCK_SIZE задает размер логического блока. Он может быть в диапазоне от 2 Кб до 32 Кб. Значение по умолчанию 8 Кб. Определенное количество смежных логических блоков формирует *экстент (extent)*. Множество экстентов, выделенных для конкретной логической структуры, образует *сегмент (segment)*. Блоки данных Oracle – наименьшие единицы логических операций ввода-вывода.

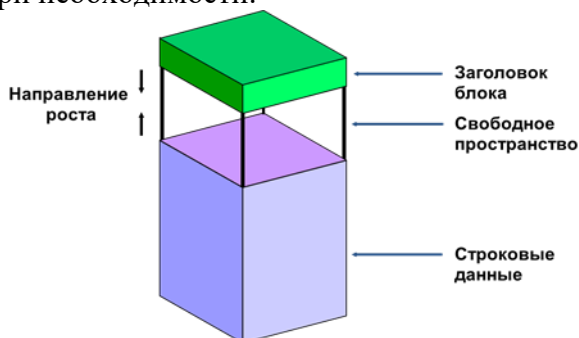


При создании таблицы для хранения ее данных создается сегмент. В табличном пространстве содержится множество сегментов. Логически таблица состоит из строк, в столбцах которых находятся значения. В конечном счете, строка располагается в блоке данных, и в общем случае в блоке хранится *часть строки* (row piece), так при определенных условиях вся строка может не поместиться в блок. Такое происходит, когда при вставке слишком большая строка не помещается в единственном блоке или когда в результате изменения существующей строки, она также выходит за пределы блока



Содержимое блока данных

- Заголовок блока (block header) содержит описатель типа блока (таблица индекс и т.д.), адрес блока данных, каталог таблиц (table directory), каталог строк (row directory), а также слоты транзакций размером по 23 байта каждый, используемые при выполнении изменений строк блока. Заголовок блока растет сверху вниз.
- Строковые данные (row data) – фактические данные, хранимые в блоке. Пространство строковых данных растет снизу вверх.
- Свободное пространство (free space) располагается посередине, что предоставляет возможность роста заголовку и данным строк. Свободное пространство занимается при вставке новых строк и при изменении строки, приводящем к увеличению размера располагаемых в ней значений. В качестве примера событий, вызывающих рост заголовка можно привести ситуации, когда требуются дополнительные входы (row entries) в каталоге строк или когда недостаточно первоначально выделенных слотов транзакций и требуется выделить дополнительные. Свободное пространство блока изначально непрерывно. Однако удаления и обновления могут фрагментировать свободное пространство блока. Сервер Oracle объединяет свободное пространство блока при необходимости.



2) Табличные пространства и файлы данных

Данные в Oracle логически хранятся в табличных пространствах, а физически располагаются в файлах данных.

- Табличные пространства:
 - могут принадлежать только одной базе данных в определенный момент времени
 - состоят из одного или более файлов данных
 - в дальнейшем разбиваются на логические единицы хранения
- Файлы данных:
 - Могут принадлежать только одному табличному пространству и одной БД.

- Являются репозиторием для данных объектов схем.

Базы данных, табличные пространства и файлы данных тесно связаны между собой, но имеют важные отличия:

- База данных Oracle состоит из одной или более логических структур, называемых табличными пространствами, которые в совокупности хранят всю информацию базы данных.
- Каждое табличное пространство базы данных Oracle состоит из одного или более файлов, называемых файлами данных, которые являются физическими структурами, соответствующими операционной системе, в которой выполняется Oracle.

Информация базы данных хранится в файлах данных, которые образуют определенное табличное пространство. Простейшая база данных Oracle может иметь два табличных пространства (SYSTEM и SYSAUX), в каждом из которых по одному файлу данных. Другая база данных может, например, состоять из трех табличных пространств, в каждом из которых по два файла (всего шесть файлов данных). Одна база данных может одержать до 65534 файлов данных.

3) Файлы, сопровождаемые Oracle (OMF)

Использование *файлов, сопровождаемых Oracle (Oracle Managed Files – OMF)* устраняет необходимость прямого управления администратором файлами операционной системы, входящими в базу данных Oracle. Операции задаются в терминах объектов базы данных, а не с использованием имен файлов. При необходимости база данных использует внутри себя стандартный интерфейс файловой системы, чтобы создавать и удалять файлы следующих компонентов БД:

- табличных пространства;
- журнальных файлов;
- управляющих файлов;
- архивных журналов;
- файлов отслеживания измененных блоков;
- журналов отката;
- резервов RMAN.

В базе данных могут быть как файлы, сопровождаемые, так и несопровождаемые сервером Oracle. Каталоги файловой системы, задаваемые в этих параметрах, должны существовать и база данных их не создает. Для каталога должны быть установлены права доступа, позволяющие базе данных создавать в нем файлы.

Пример:

```
SQL> ALTER SYSTEM SET DB_CREATE_FILE_DEST = '/u01/oradata';
SQL> CREATE TABLESPACE tbs_1;
```

В примере задан параметр DB_CREATE_FILE_DEST. Поэтому в команде CREATE TABLESPACE опущена фраза DATAFILE. Файл данных создается в месторасположении, определяемом параметром DB_CREATE_FILE_DEST.

Параметр	Описание
DB_CREATE_FILE_DEST	Указывает директорию по умолчанию в файловой системе для файлов данных и временных файлов
DB_CREATE_ONLINE_LOG_DEST_n	Определяет месторасположение по умолчанию для создаваемых журнальных и управляющих файлов
DB_RECOVERY_FILE_DEST	Определяет месторасположение резервных объектов, получаемых с помощью утилиты RMAN

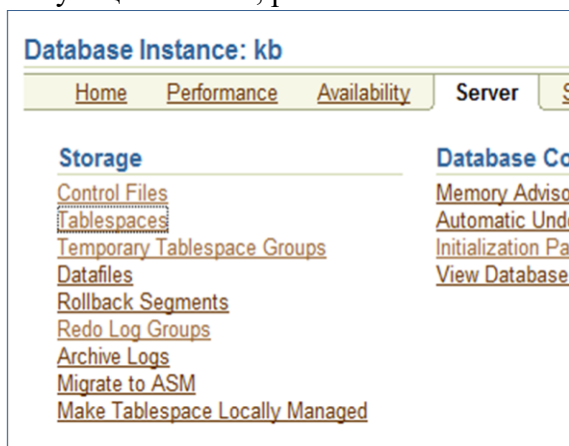
4) Управление пространством в табличном пространстве

Память в табличном пространстве выделяется экстендами. Во время создания табличного пространства выбирается один из двух методов отслеживания информации о свободном и используемом пространстве.

- Табличное пространство с локальным управлением:
 - Сопровождение сведений о свободных экстендах производится в табличном пространстве.
 - Информация о свободных экстендах хранится в битовой карте.
 - Каждый бит соответствует блоку или группе блоков.
 - Битовое значение показывает, свободен ли блок или он используется.
 - Рекомендуются использовать табличные пространства с локальным управлением.
- Табличное пространство, управляемое с помощью словаря данных:
 - Управление свободными экстендами производится с помощью словаря данных.
 - При выделении и освобождении экстендов вносятся изменения в соответствующие таблицы.
 - Такие табличные пространства поддерживаются только для обратной совместимости.

5) Исследование структур хранения

Логические структуры данных хранятся в физических файлах базы данных. С помощью Enterprise Manager можно без труда просмотреть информацию о логических структурах базы данных. Для получения подробных сведений о каждой структуре щелкните на соответствующей ссылке, расположенной в секции Storage на странице Server.



Создание нового табличного пространства.

Чтобы создать табличное пространство, выполните следующие шаги:

1. Выберите закладку Server, а затем щелкните на ссылке Tablespaces под заголовком Storage.
2. Щелкните на кнопке Create.
3. Введите имя табличного пространства: kb2023.
4. Выберите Locally Managed под заголовком Extent Management.

Сервер Oracle эффективно управляет экстендами внутри табличного пространства такого вида. В случае табличных пространств, управляемых с помощью словаря данных, администратор должен более активно участвовать в управлении экстендами, и для отслеживания изменений необходим доступ к словарю данных. Такие пространства больше не поддерживаются. Oracle не рекомендует их использовать.

5. Выберите Permanent под заголовком Type. Табличные пространства заданного типа хранят постоянные объекты БД, создаваемые системой и пользователями.

Create Tablespace

Show SQL Cancel OK

General Storage

* Name

Extent Management

Type

Permanent

Locally Managed

Dictionary Managed

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.

Select Name

Directory

Size (MB)

Add

No items found

General Storage

Show SQL Cancel OK

Database Instance: kb > Tablespaces >

Logged in As SYS

Add Datafile

Cancel Continue

* File Name

* File Directory C:\APP\IEUSER\ORADATA\KB\

Tablespace TR1

File Size 100 MB

Reuse Existing File

Storage

Automatically extend datafile when full (AUTOEXTEND)

Increment KB

Maximum File Size Unlimited

Value MB

TIP Changes made on this page will NOT take effect until you click "OK" button on the Tablespace page.

Cancel Continue

Extent Allocation

Automatic

Uniform

Size KB

Segment Space Management

Automatic

Objects in the tablespace automatically manage their free space. It offers high performance for free space management.

Manual

Objects in the tablespace will manage their free space using free lists. It is provided for backward compatibility.

Enable logging

Yes

Generate redo logs for creation of tables, indexes and partitions, and for subsequent inserts. Recoverable

No

Redo log entries are smaller, the above operations are not logged and not recoverable.

Block information

Block Size (B) 8192

6. Выберите Read Write под заголовком Status. Статус Read Write означает, что пользователи могут читать и писать в табличное пространство после того, как оно будет создано. Это значение по умолчанию.

7. Чтобы добавить файлы данных к табличному пространству, щелкните на кнопке Add в секции Datafiles. Табличное пространство должно содержать хотя бы один файл.

Табличные пространства вида Bigfile используются чрезвычайно большими базами данных. В них применяется предлагаемая Oracle возможность *автоматического управления хранением (Automatic Storage Management)*, другие инструменты управления логическими томами с поддержкой *расщепления (striping)* и дисковых массивов (redundant array of independent disks – RAID), а также динамическое расширение логических томов.

8. На странице Add Datafiles введите имя файла. Оставьте стандартные или введите новые значения в полях File Directory и File Size.

9. В секции Storage можно выбрать «Automatically extend datafile when full (AUTOEXTEND)» и задать в поле Increment размер, который будет добавляться к файлу каждый раз, когда он будет полностью заполнен. Однако при этом продолжает действовать ограничение физического размера носителя, на котором размещается файл. Оставьте в поле Maximum File Size значение Unlimited. Щелкните на кнопке «OK». После этого вы вернетесь на общую страницу создания табличного пространства.

10. Щелкните на закладке Storage. Появляется страница Create Tablespace Storage.

11. Оставьте все стандартные значения, выводимые на странице Storage.

Хранение данных в локально-управляемых табличных пространствах.

Экстенды внутри локально-управляемого табличного пространства могут быть выделены одним из двух способов:

- **Automatic.** Этот способ также называют автоматическим выделением. При его использовании предполагается, что размер экстенгов внутри табличного пространства определяется системой и нельзя задать размер экстенга. Вы не можете выбрать automatic для временного табличного пространства.
- **Uniform** определяет, что табличное пространство содержит экстенги одинакового и заданного вами размера. Стандартный размер 1 мегабайт. Все экстенги временного табличного пространства по умолчанию всегда имеют одинаковый размер. Нельзя задать параметр uniform для табличного пространства типа undo.

Управление пространством внутри (Segment Space Management) сегментов локально-управляемого табличного пространства может быть:

- **Automatic.** В этом случае Oracle использует битовые матрицы для управления свободным пространством внутри сегментов. Битовая матрица – это карта, в которой описан статус каждого блока сегмента. Статус зависит от размера пространства блока, доступного для вставки строк. Как только больше или меньше пространства блока становится доступным, это отражается в битовой карте. Такие карты позволяют Oracle автоматически управлять свободным пространством и поэтому этот метод называют *автоматическим управлением пространством сегмента (Automatic Segment Space Management – ASSM)*.
- **Manual** означает, что для управления свободным пространством внутри сегментов используются списки свободных блоков. В них попадают блоки, доступные для вставки строк. Такой вид управления называется *ручным управлением пространством сегментов (manual segment space management)*, поскольку необходимо задавать и настраивать параметры хранения созданных в табличных пространствах объектов схем: PCTUSED, FREELISTS, и FREELIST GROUPS. Этот метод поддерживается для обратной совместимости. Рекомендуется использовать ASSM.

Преимущества табличных пространств с локальным управлением

Табличные пространства с локальным управлением (Locally Managed) имеют следующие преимущества над табличными пространствами, управляемыми с помощью словаря данных:

- Локальное управление устраняет использование рекурсивных операций, возникающих, когда управление табличным пространством осуществляется с помощью словаря системы и выделение или освобождение экстенгов приводит к выполнению операции выделения или освобождения экстенгов в сегменте отмены или таблицах словаря данных.
- Так как информация о свободном пространстве не записывается в словарь данных, уменьшается конкуренция за его таблицы.
- Автоматически отслеживаются смежные свободные пространства и поэтому отпадает необходимость их объединения.
- Размер экстенгов может определяться автоматически системой или же все экстенги будут иметь один и тот же размер.
- Изменения в битовых картах не приводят к генерации информации отмены, так как не вносятся изменения в таблицы словаря данных (за исключением специальных случаев изменения информации о квотах табличного пространства).

Примечание: для перевода табличного пространства, управляемого с помощью словаря данных, в табличное пространство с локальным управлением можно использовать пакетную процедуру DBMS_SPACE_ADMIN.TABLESPACE_MIGRATE_TO_LOCAL.

Режим журналирования (Logging)

Изменения, вносимые в объекты табличного пространства, регистрируются в оперативном журнале (redo log). К примеру, когда режим журналирования отключен,

операции создания объектов, использующие прямую вставку (direct load path inserts), не отражаются в оперативном журнале, и такие объекты не восстановимы в случае потери данных. Поэтому, если объекты создаются без включения режима Logging, их необходимо резервировать для обеспечения последующего восстановления.

Размер блока (Block Information)

В этой секции показывается размер блока для создаваемого табличного пространства. Его нельзя изменить, если только не заданы параметры инициализации DB_nK_CACHE_SIZE. Тогда размер блока можно выбрать из списка этих альтернативных размеров.

Табличные пространства учебной базы данных

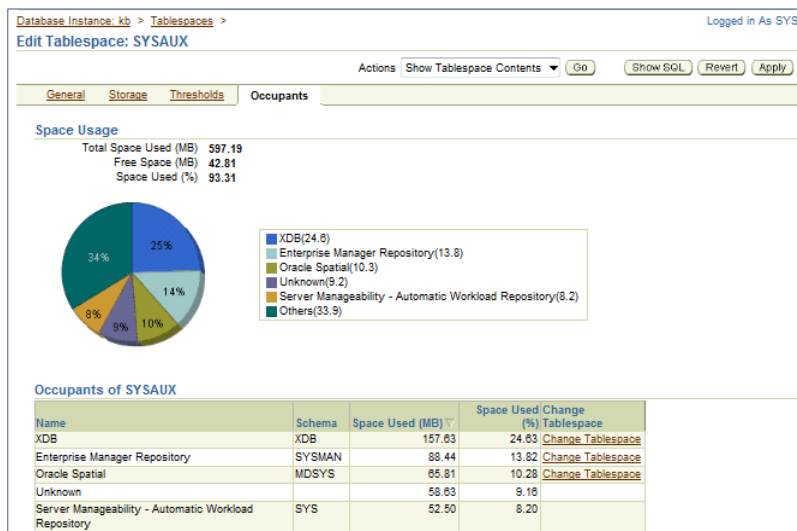
Edit View Delete Actions Add Datafile Go												
Select	Name	Allocated Size(MB)	Space Used (MB)	Allocated Space Used(%)	Auto Extend	Allocated Free Space(MB)	Status	Datafiles	Type	Extent Management	Segment Management	
<input checked="" type="radio"/>	EXAMPLE	345.6	309.8	<div><div></div></div> 89.6	YES	35.8	✓	1	PERMANENT	LOCAL	AUTO	
<input type="radio"/>	SYSAUX	630.0	588.4	<div><div></div></div> 93.4	YES	41.6	✓	1	PERMANENT	LOCAL	AUTO	
<input type="radio"/>	SYSTEM	730.0	721.2	<div><div></div></div> 98.8	YES	8.8	✓	1	PERMANENT	LOCAL	MANUAL	
<input type="radio"/>	TEMP	20.0	3.0	<div><div></div></div> 15.0	YES	17.0	✓	1	TEMPORARY	LOCAL	MANUAL	
<input type="radio"/>	UNDOTBS1	65.0	15.3	<div><div></div></div> 23.6	YES	49.7	✓	1	UNDO	LOCAL	MANUAL	
<input type="radio"/>	USERS	5.0	4.1	<div><div></div></div> 81.2	YES	0.9	✓	1	PERMANENT	LOCAL	AUTO	
Total Allocated Size (GB) 1.75 Total Used (GB) 1.60 Total Allocated Free Space (GB) 0.15												

В учебной базе данных созданы следующие табличные пространства:

- **SYSTEM.** Табличное пространство SYSTEM используется сервером Oracle для управления базой данных. Оно содержит словарь данных и таблицы с административной информацией о БД. Все эти объекты располагаются в схеме SYS и доступны только пользователю SYS или другим пользователям-администраторам, имеющим необходимые привилегии.
- **SYSAUX.** Вспомогательное к SYSTEM табличное пространство. Некоторые компоненты и продукты, которые использовали в предыдущих версиях Oracle табличное пространство SYSTEM или свои собственные табличные пространства, теперь применяют SYSAUX. Каждая база данных Oracle версии 10g и следующих версий должна иметь табличное пространство SYSAUX.

В Enterprise Manager выводится секторная диаграмма, отражающая содержимое этого табличного пространства. Чтобы просмотреть диаграмму, щелкните на ссылке [Tablespaces](#), расположенной на странице Server. Выберите SYSAUX и щелкните на кнопке Edit. Затем щелкните на закладке [Occupants](#).

С помощью ЕМ после создания компонента (occupant) можно выполнять мониторинг использования пространства в SYSAUX этим компонентом. При обнаружении чрезмерного использования пространства или приближении такой ситуации можно переместить компонент в другое табличное пространство. Для этого надо выбрать компонент и щелкнуть на кнопке Change Tablespace.



- **TEMP.** Табличное пространство, используемое при выполнении команд SQL для создания временных сегментов (например, с данными сортировки большого объема или данными создаваемого индекса). Каждому пользователю назначается как табличное пространство по умолчанию для хранения создаваемых объектов с данными, так и временное табличное пространство по умолчанию. Наилучший практический метод состоит в задании для базы данных временного табличного пространства, назначаемого новому создаваемому пользователю, если только при этом для него не указывается другое. В преконфигурированной базе данных TEMP – это *временное табличное пространство по умолчанию*. Это означает, что при создании пользователя оно назначается ему в качестве временного табличного пространства, если явно не указано другое табличное пространство.

- **UNDOTBS1.** Табличное пространство, в котором сервер БД хранит информацию отмены. Каждый экземпляр базы данных должен иметь только одно такое пространство в данный момент времени, если используется *автоматическое управление информацией отмены* (Automatic Undo Management). Табличное пространство типа UNDO создается вместе с базой данных.

- **USERS.** Табличное пространство для хранения постоянных объектов и данных пользователей. В учебной базе данных USERS определено в качестве табличного пространство по умолчанию для всех объектов, принадлежащих пользователям, кроме SYS и SYSTEM. Для двух этих пользователей в качестве *постоянного табличного пространства по умолчанию (default permanent tablespace)* остается SYSTEM.

- **EXAMPLE.** Табличное пространство, содержащее объекты схем с примерами, которые могут быть установлены в процессе создания базы. Эти схемы используются в документации Oracle и в курсах обучения.

Примечание: в общем случае для упрощения администрирования следует иметь отдельное табличное пространство для индексов.

Изменение табличного пространства

В созданное табличное пространство можно внести различные изменения.

Переименование

Просто введите новое имя для табличного пространства и щелкните на кнопке Apply.

Изменение статуса (Status)

Табличное пространство может иметь один из трех статусов или состояний. В зависимости от типа табличного пространства не все состояния могут быть доступны.

- **Read Write** означает, что табличное пространство в оперативном состоянии и доступно для чтения и записи.
- **Read Only** - режим, разрешающий в транзакциях одни лишь чтения. До перевода в этот режим все существующие транзакции должны завершиться (фиксацией или откатом). Дальнейшие операции DML, затрагивающие это табличное пространство, не разрешены, и

они будут завершаться аварийно. Табличное пространство, переведенное в режим «Read Only», находится в оперативном состоянии. Табличные пространства SYSTEM и SYSAUX нельзя перевести в режим «Read Only».

- **Offline** - автономный режим, сделав информацию в пространстве временно недоступна для общего использования. Оставшаяся часть БД открыта пользователям. Различные варианты перевода в автономное состояние:
 - **Normal**; табличное пространство может быть нормально переведено в автономное состояние, если нет ошибочных условий ни для одного файла данных табличного пространства. При переводе в автономное состояние Oracle гарантирует запись на диск всех данных табличного пространства, выполняя контрольную точку для всех его файлов данных.
 - **Temporary**; табличное пространство может быть временно переведено в автономное состояние, даже если есть ошибочные ситуации для одного или нескольких файлов табличного пространства. В автономное состояние переводятся оперативные файлы, которые еще не находятся в автономном состоянии. Для них выполняется контрольная точка. Если же все файлы в оперативном состоянии, но перевод табличного пространства в автономное состояние выполняется с опцией temporary, тогда перед обратным переводом пространства в оперативное состояние не потребуется восстановление. Однако, если один или несколько файлов табличного пространства находятся в автономном состоянии из-за ошибок записи, и вы перевели все пространство временно в автономный режим, тогда перед обратным переводом в оперативное состояние потребуется восстановить табличное пространство.
 - **Immediate**; табличное пространство можно незамедлительно перевести в автономное состояние без контрольной точки для всех его файлов данных. Перед обратным переводом в оперативное состояние необходимо выполнить восстановление табличного пространства. Нельзя перевести в автономное состояние пространство с параметром immediate, когда база данных в режиме NOARCHIVELOG.
 - **For Recover**; эта опция больше не поддерживается. Синтаксис команды содержит этот параметр для обратной совместимости с предыдущими версиями Oracle.

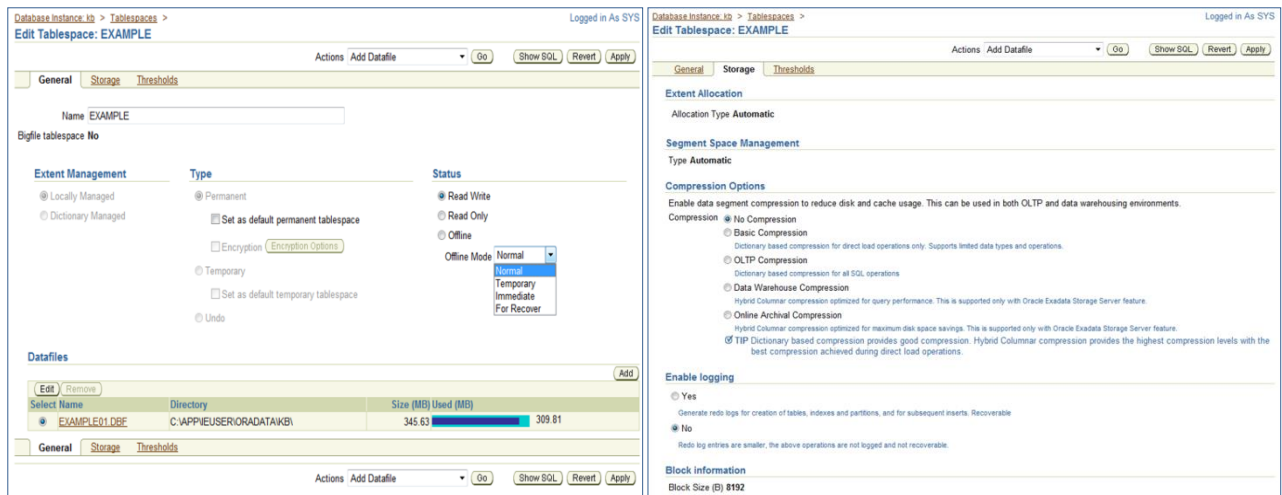
Изменить размер табличного пространства можно, добавив ему файлы данных или изменив размеры какого-либо существующего файла данных этого пространства.

- Чтобы добавить новый файл данных, щелкните на кнопке Add и введите на странице Add Data File информацию о новом файле.
- Чтобы изменить размер существующего файла данных, выберите файл в области Datafiles на странице Edit Tablespace, щелкнув на имени файла или выбрав файл и щелкнув на кнопке Edit. Затем на странице Edit Datafile измените размер файла. Табличное пространство можно таким образом сделать больше или меньше. Однако нельзя сделать размер файла данных меньше, чем размер используемой в нем памяти. Появится сообщение об ошибке: «ORA-03297: file contains used data beyond requested RESIZE value».

Закладка Thresholds для изменения *пороговых значений* использования памяти табличного пространства, превышение которых вызывает генерацию предупреждений и критических сообщений. Имеется три варианта выбора:

- Use Default Thresholds; можно использовать предустановленные стандартные пороговые значения, а также установить новые стандартные значения.
- Specify Thresholds; установка пороговых значений для конкретного табличного пространства.
- Disable Thresholds; отключение генерации сигнальных сообщений, связанных с использованием памяти данного табличного пространства.

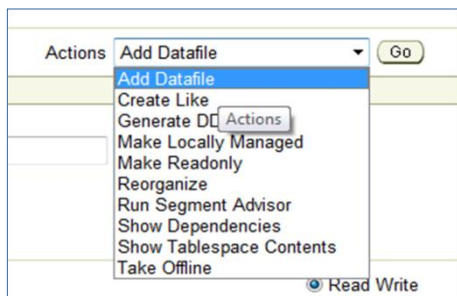
Примечание: может пройти несколько минут перед тем, как будет зарегистрировано сигнальное сообщение о превышении порогового значения.



Операции с табличными пространствами

С помощью меню Actions можно решать различные задачи, связанные с табличными пространствами. Выберите табличное пространство и действие, которое вы хотите выполнить:

- **Add Datafile** – добавление файла данных, делающее табличное пространство большим по размеру.
- **Create Like** – создание другого табличного пространства с использованием выбранного табличного пространства в качестве шаблона.
- **Generate DDL** – генерация команды DDL, создающей табличное пространство. Эта команда может быть затем скопирована и перенесена в текстовый файл для использования в скрипте или документации.
- **Make Locally Managed**; если управление выбранного пространства осуществляется с помощью словаря данных, оно будет преобразовано в табличное пространство с локальным управлением. Обратное преобразование невозможно
- **Make Readonly** – остановка всех записей в табличное пространство. Разрешено завершение текущих транзакций, однако запрещено начинать новые команды DML или другие операции записи в это табличное пространство. Действие доступно, если табличное пространство не находится уже в состоянии ‘только чтение’.
- **Reorganize** – запуск мастера реорганизации, который можно использовать для перемещения объектов внутри табличного пространства в целях освобождения памяти. Такие действия следует выполнять в часы, когда объекты табличного пространства мало используются.
- **Run Segment Advisor**; консультант Segment Advisor помогает определить, есть ли у объекта пространство, которое можно вернуть базе данных для использования, устранив фрагментацию внутри этого объекта. На уровне табличного пространства генерируются советы для каждого сегмента табличного пространства.
- **Show Dependencies**; выводятся объекты, зависящие от табличного пространства, и объекты, от которых зависит данное пространство.
- **Show Tablespace Contents**; выводится информация о всех сегментах табличного пространства, а также дополнительно графическая карта расположения этих сегментов.
- **Take Offline**; в результате оперативное табличное пространство становится недоступным. Оно не удаляется, оно только переводится в недоступное состояние.

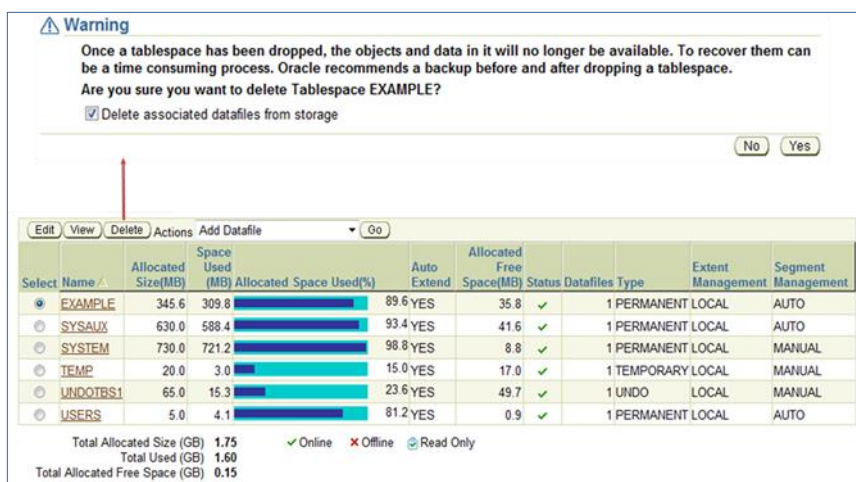


Удаление табличного пространства.

Табличное пространство и все его сегменты можно удалить из базы данных. Для этого пользователю требуется системная привилегия DROP TABLESPACE.

При удалении табличного пространства из управляющего файла базы данных вычеркиваются указатели на удаляемые файлы. Дополнительно можно задать удаление файлов операционной системы (файлов данных), принадлежащих удаляемому табличному пространству. Если этот параметр не указывается, файлы в операционной системе необходимо после удалить вручную.

Нельзя удалить табличное пространство, содержащее активные сегменты, например, используемые в настоящий момент таблицы. Нельзя удалить табличное пространство, содержащее данные отмены, которые необходимы для отката незафиксированных транзакций. Перед удалением табличное пространство может быть в оперативном или автономном состоянии. Однако лучше сначала перевести его в автономное состояние и затем удалить.



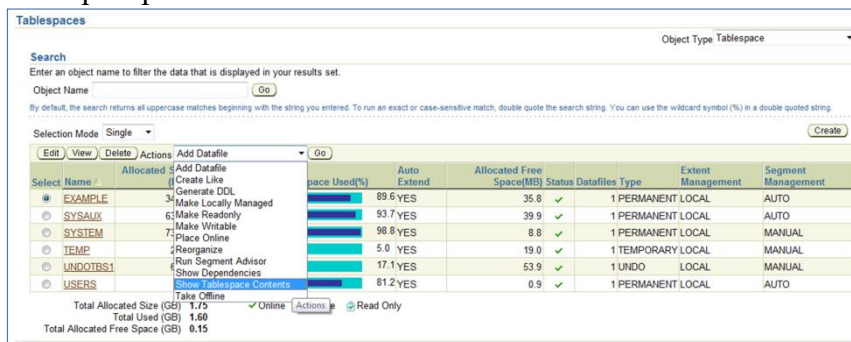
Просмотр сведений о табличных пространствах

Щелкните на кнопке View, чтобы вывести на экран информацию о выбранном табличном пространстве. Находясь на странице View Tablespace, можно также по кнопке Edit перейти на страницу изменения табличного пространства.

Выводимая информация о табличном пространстве и файлах данных может быть также получена путем выполнения запросов к следующим представлениям.

- Информация о табличном пространстве:
 - DBA_TABLESPACES
 - V\$TABLESPACE
- Информация о файлах данных:
 - DBA_DATA_FILES
 - V\$DATAFILE
- Информация о временных файлах:
 - DBA_TEMP_FILES
 - V\$TEMPFILE
- Сбор информации о структурах хранения данных

Чтобы в ЕМ просмотреть и изменить данные о табличных пространствах, выберите Server>Tablespaces. Для выбора необходимой операции используйте кнопки и действия из выпадающего списка поля Actions. На рисунке, к примеру, выбирается действие Show Tablespace Contents для вывода информации о структурах хранения, располагаемых в табличном пространстве.



Просмотр содержимого табличного пространства

Подробная информация о табличном пространстве выводится на странице Show Tablespace Contents. В выводимых сведениях приводится список сегментов табличного пространства и для каждого сегмента его тип, размер и количество экстенгов. По любому из этих четырех значений можно отсортировать выводимые данные, щелкнув для этого на соответствующий заголовок столбца. Кроме того, можно отфильтровать выводимый список, введя критерий выбора в секции Search. Для табличного пространства, управляемого с помощью словаря данных, дополнительно выводятся столбцы:

- Max Extents (максимально допустимое число экстенгов);
- Next (размер следующего экстенга);
- Percent Increase (процент роста последующего экстенга по отношению к предыдущему).

Для просмотра списка экстенгов щелкните на ссылке в столбце Extents.

Для просмотра экстенгов в графическом виде раскройте «Extent map» и перемещайте курсор над отдельными экстенгами. При этом отображается следующая информация:

- имя сегмента, которому принадлежит экстенг;
- номер экстенга (Extent ID);
- номер блока (Block ID);
- размер экстенга в блоках;
- файл данных, в котором располагается экстенг.

Show Tablespace Contents

Size (MB) 345.6 Used (MB) 309.8 Extent Mgmt LOCAL Auto Extend Yes Return

Block Size (KB) 8 Used (%) 89.6 Segment Mgmt AUTO Extents 532

Segments

Search

Segment Name Type Minimum Size (KB) Minimum Extents

All Types 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.

Highlight Extents	Segment Name	Type	Size (KB)	Extents
	SH.CUSTOMERS	TABLE	12,288	27
	SH.SALES.SALES_Q4_2001	TABLE PARTITION	8,192	1
	SH.SALES.SALES_Q3_2001	TABLE PARTITION	8,192	1
	SH.SALES.SALES_Q2_2001	TABLE PARTITION	8,192	1
	SH.SALES.SALES_Q1_2001	TABLE PARTITION	8,192	1
	SH.SALES.SALES_Q4_2000	TABLE PARTITION	8,192	1
	SH.SALES.SALES_Q3_2000	TABLE PARTITION	8,192	1
	SH.SALES.SALES_Q2_2000	TABLE PARTITION	8,192	1
	SH.SALES.SALES_Q1_2000	TABLE PARTITION	8,192	1
	SH.SALES.SALES_Q4_1999	TABLE PARTITION	8,192	1

Previous 1-10 of 313 Next 10

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.

Zoom 25%

Увеличение базы данных происходит, если:

- Создать новое табличное пространство
- Добавить файл данных в существующее табличное пространство
- Увеличить размер файла данных
- Включить возможность динамического расширения файла данных

Действия по увеличению базы данных можно выполнить, используя Enterprise Manager или с помощью команд SQL. В конечном счете, размер базы данных можно представить в виде суммы размеров всех табличных пространств.

Отобразите для табличного пространства SYSAUX: статус, размер блока данных, размеры файлов, компоненты этого табличного пространства и размер занимаемого компонентами дискового пространства, экстенды в графическом виде «Extent map».

Переименуйте созданное ранее табличное пространство «kb2023» в «kb-5-2023». Переведите его в режим «Read only». Добавьте в это пространство еще один файл. Верните режим «Read Write».

Создайте новое табличное пространство с локальным управлением (LMT) с именем INVENTORY размером 5 MB.
(Tablespace Name: INVENTORY Extent Management is Locally Managed. Type is Permanent. Status is Read Write. В блоке Datafiles: Убедитесь, что “Use bigfile tablespace” НЕ отмечено, Filename INVENTORY, File Size 5 MB)

Проверить, что INVENTORY не авторасширяемое

Выполните скрипт для создания и заполнения таблицы X в табличном пространстве INVENTORY. Обратите внимание на ошибки.

Скрипт:

```
CREATE TABLE x (a CHAR(1000)) TABLESPACE inventory;  
INSERT INTO x VALUES ('a');  
INSERT INTO x SELECT * FROM x;  
INSERT INTO x SELECT * FROM x;  
INSERT INTO x SELECT * FROM x;  
INSERT INTO x SELECT * FROM x;  
INSERT INTO x SELECT * FROM x;  
INSERT INTO x SELECT * FROM x;  
INSERT INTO x SELECT * FROM x;  
INSERT INTO x SELECT * FROM x;  
INSERT INTO x SELECT * FROM x;  
INSERT INTO x SELECT * FROM x;  
INSERT INTO x SELECT * FROM x;  
INSERT INTO x SELECT * FROM x;  
INSERT INTO x SELECT * FROM x;  
COMMIT;
```

Должна появиться ошибка «ORA-01653: unable to extend table SYS.X by 128 in tablespace INVENTORY»

Убедитесь, что в журнале Alert Log есть запись об ошибке (см. рис.)

Database Instance: kb > View Alert Log Contents > Logged in As SYS

Search Alert Log Contents Page Refreshed Feb 24, 2019 10:49:55 AM PST [Refresh](#) [Custom Log Location](#)

Log Location: C:\APP\USER\diag\rdbs\kb\kb1alrt
 Modified: Feb 24, 2019 10:26:34 AM (UTC-08:00)
 Size (MB): 1.16

Search Criteria

Date Range: ☒ Most Recent 8 Hours

Time Interval: Start Date: End Date:
 (Example: Feb 24, 2019) (Example: Feb 25, 2019)
 Start Time: 01:00 AM @ PM End Time: 01:00 AM @ PM

Message Types: ☒ Incident Error ☒ Warning ☒ Trace
☒ Error ☒ Notification ☒ Unknown

Message Text:
☐ Regular Expression

Maximum Entries Retrieved: 500
 Entries Per Page: 25

[Advanced Search Options: Filter By Log Message Fields](#)

Results

Timestamp	Type	Level	ID Group	Message ID	Message Text
Feb 24, 2019 10:26:34 AM PST	NOTIFICATION	16	admin_ddl: opeixe:3070:2802784106		Completed: ALTER TABLESPACE "INVENTORY" ADD DATAFILE 'C:\APP\USER\ORADATA\KB\INVENTORY'
Feb 24, 2019 10:28:30 AM PST	NOTIFICATION	16	admin_ddl: opeixe:2999:4222384190		ALTER TABLESPACE "INVENTORY" ADD DATAFILE 'C:\APP\USER\ORADATA\KB\INVENTORY02' SIZE 10M
Feb 24, 2019 10:19:31 AM PST	UNKNOWN	16			ORA-1053: unable to extend table SYS.X by 128 in tablespace INVENTORY

Увеличьте размер пространства для INVENTORY. Сделаем это двумя способами. Сначала увеличим размер файла данных до 40 MB. Затем добавим еще один файл размером 30 MB. В обоих случаях используйте «Show SQL» для ознакомления с выполняемыми командами. Убедитесь, что пространство увеличилось.

Удалите таблицу x командой `drop table x purge;`
 Повторите выполнение скрипта. Убедитесь, что ошибка больше не повторилась.
 Удалите таблицу x командой `drop table x purge;`

5) Автоматическое управление пространством хранения данных

Автоматическое управление пространством хранения данных (Automatic Storage Management – ASM) осуществляет вертикальную интеграцию файловой системы и менеджера томов. Эта функциональная возможность специально создана для файлов базы данных Oracle. Возможность ASM может быть сконфигурирована для одной машины с архитектурой SMP или на нескольких узлах кластера при использовании Oracle Real Application Clusters (RAC).

ASM распределяет загрузку ввода-вывода между всеми доступными ресурсами в целях достижения оптимальной производительности, устраняя необходимость ручной настройки ввода-вывода. ASM помогает администраторам сопровождать динамическую среду баз данных, позволяя увеличивать размер базы данных без ее обязательной остановки для регулировки распределения пространства хранения.

ASM позволяет сопровождать избыточные копии данных для достижения отказоустойчивости, а также может работать поверх механизмов надежного хранения, предоставляемых производителями оборудования. Управление данными осуществляется путем выбора характеристик надежности и производительности для соответствующих классов информации, а не путем ручной настройки работы с файлами.

Возможности ASM экономят время администраторов БД, автоматизируя ручные операции со структурами хранения. В результате появляется возможность управления базами данных большего размера с более высокой эффективностью.

Основные характеристики и преимущества ASM

- Расщепление файлов, отсутствие логических томов
- Оперативная реконфигурация и динамическая перебалансировка дисков
- Настраиваемая скорость перебалансировки
- Обеспечение избыточности на файловой основе
- ASM поддерживает только файлы базы данных Oracle
- Поддержка кластерной архитектуры
- Автоматическая инсталляция

В соответствии с ASM файлы делятся на экстененты (они отличаются от экстенентов файлов данных, рассмотренных ранее). Экстененты каждого файла равномерно распределяются по всем дискам. ASM использует индексный механизм для отслеживания расположения каждого экстенента. При изменении возможностей устройств хранения ASM не производит повторное расщепление всех данных. Производится перемещение данных такого объема, который пропорционален объему добавленной или удаленной дисковой памяти так, чтобы равномерно перераспределить файлы и обеспечить сбалансированную загрузку дисков. Эти действия производятся при активной базе данных.

Вы можете повысить скорость операции перебалансировки или же снизить ее, чтобы уменьшить влияние этой операции на подсистему ввода-вывода. ASM обеспечивает защиту на основе зеркалирования данных. При этом не требуется покупать Менеджер логических томов (Logical Volume Manager) сторонней фирмы. Одно из уникальных преимуществ ASM состоит в том, что зеркалирование применяется на файловой основе, а не на основе томов. Поэтому одна и та же группа дисков может хранить совокупность файлов, защищаемых путем зеркалирования или не защищаемых совсем.

ASM поддерживает файлы данных, журнальные файлы, управляющие файлы, архивные журналы, резервные наборы, полученные с помощью RMAN, и другие типы файлов базы данных Oracle. ASM поддерживает Real Application Clusters и устраняет необходимость использования менеджера логических томов кластера (Cluster Logical Volume Manager) или кластерной файловой системы (Cluster File System).

Основные характеристики и преимущества ASM

ASM не исключает использования прежде существовавших возможностей базы данных. Существующие БД могут продолжать работать по-прежнему. Новые файлы можно создавать в виде ASM-файлов. В то же время администрирование существующих файлов может осуществляться старыми методами, но возможна миграция этих файлов в среду ASM. На диаграмме описываются связи, существующие между различными компонентами хранения информации внутри базы данных Oracle. В левой части и средней части диаграммы показаны связи, которые существовали в предыдущих версиях Oracle. В правой части отражаются новые понятия, введенные вместе с ASM.

Файлы БД теперь могут храниться в виде ASM-файлов. На вершине новой иерархии находятся *дисковые группы ASM (ASM disk groups)*. Любой ASM-файл находится только в одной дисковой группе. Однако дисковая группа может содержать файлы, которые принадлежат нескольким базам данных, и одна база данных может использовать пространство нескольких дисковых групп. Как можно заметить, одна дисковая группа состоит из ASM-дисков и каждый ASM-диск принадлежит только к одной дисковой группе. ASM-файлы всегда распределяются по всем ASM-дискам в дисковой группе. ASM-диски разбиваются на части (partition), состоящие из *единичных секций (allocation units – AU)*, размером в 1 мегабайт. Единичные секции – наименьшие непрерывные участки дискового пространства, которые выделяет ASM. ASM запрещает расщепление блоков между несколькими единичными секциями.



Примечание. На рисунке отражен только один тип ASM-файла: файл данных. Однако ASM может использоваться для хранения других типов файлов базы данных.

Время на выполнение лабораторной работы – 2 часа.