



**PDM STEP Suite**

**Архивация и  
восстановление  
данных в Oracle  
вер. 2.1**

**руководство администратора**

# 1. Содержание.

1.	СОДЕРЖАНИЕ.....	2
2.	ИСПОЛЬЗУЕМЫЕ СОКРАЩЕНИЯ.....	3
3.	РЕЖИМЫ РАБОТЫ БД.....	3
4.	СОСТОЯНИЯ ЭКЗЕМПЛЯРА БД.....	3
5.	ПЕРЕВОД БД В АРХИВНЫЙ РЕЖИМ РАБОТЫ.....	4
5.1.	Перевод БД в архивный режим средствами DBA STUDIO.....	4
5.2.	Перевод БД в архивный режим с помощью SERVER MANAGER.....	5
6.	ЧТО НУЖНО АРХИВИРОВАТЬ? .....	6
7.	СТРАТЕГИЯ АРХИВАЦИИ.....	9
8.	СРАВНЕНИЕ СПОСОБОВ АРХИВАЦИИ.....	10
9.	АРХИВАЦИЯ СРЕДСТВАМИ ОС.....	10
9.1.	Архивация «на холодную».....	10
9.2.	Архивация «на горячую».....	11
10.	ВОССТАНОВЛЕНИЕ СРЕДСТВАМИ ОС.....	12
10.1.	Полное восстановление БД.....	12
10.2.	Частичное восстановление БД.....	13
11.	АРХИВАЦИЯ И ВОССТАНОВЛЕНИЕ ПРИ ПОМОЩИ ЭКСПОРТА И ИМПОРТА ДАННЫХ.....	14
12.	АРХИВАЦИЯ И ВОССТАНОВЛЕНИЕ СРЕДСТВАМИ RECOVERY MANAGER(RMAN) .....	15
12.1.	Настройка переменных окружения для работы RMAN (ORACLE 8.1.7).....	15
12.2.	Основные понятия RMAN.....	15
12.3.	Режимы работы RMAN.....	16
12.4.	Создание БД каталога для RMAN.....	16
12.5.	Стратегия архивации с помощью RMAN.....	18
12.6.	Полная архивация БД и журналов транзакций.....	19
12.7.	Добавочная архивация БД.....	22
12.8.	Восстановление БД.....	22
12.9.	Управление архивом.....	25

## 2. Используемые сокращения.

ОС – операционная система.

БД – база данных.

**SID (Oracle System Identifier)** – уникальный идентификатор экземпляра БД Oracle. Задается при создании экземпляра БД.

**RMAN (Recovery Manger)** - специализированная утилита Oracle, предназначенная для архивации и восстановления файлов экземпляра БД Oracle.

## 3. Режимы работы БД.

БД Oracle изначально может работать в двух различных режимах: обычном и архивном. В обычном режиме все изменения данных сразу заносятся в файлы табличных пространств. А в архивном режиме дополнительно ведутся журналы транзакций, которые позволяют в случае сбоя провести все операции БД заново тем самым восстановить её текущее состояние. Когда текущий файл журнала транзакций заполняется, БД сохраняет его в специальную директорию и присваивает уникальное имя и номер.

## 4. Состояния экземпляра БД.

1. **Открытая БД** – состояние, когда экземпляр БД запущен и доступен для пользователей.
2. **Остановленная БД** – состояние, когда экземпляр БД находится в неактивном состоянии и работа пользователей с ним невозможна.
3. **Смонтированная БД** - состояние, когда экземпляр БД запущен, но не доступен для пользователей. Используется для проведения сервисных операций с БД. Например, перевод БД в архивный режим.
4. **Не смонтированная** – состояние, когда ядро БД работает, но при этом сам экземпляр не смонтирован. Такой режим используется при создании БД.

### Команды управления состояниями.

Исходное состояние БД	Желаемое состояние БД			
	Открытая	Остановленная	Смонтированная	Не смонтированная
Открытая.		shutdown immediate	shutdown immediate startup mount	startup nomount
Остановленная	startup		strtup mount	statup nomount
Смонтированная	alter database open	shutdown immediate		shutdown immediate startup nomount
Не смонтированная	startup	shutdown immediate	shutdown immediate startup mount	

## 5. Перевод БД в архивный режим работы.

В архивном режиме работы БД позволяет архивировать свои рабочие файлы, не останавливая работу с пользователями. Бесперывный режим работы обеспечивается с помощью использования журналов транзакций и архивации их по мере заполнения. Есть два способа перевода БД в архивный режим работы: с помощью DBA Studio и с помощью Server Manager.

### 5.1. Перевод БД в архивный режим средствами DBA Studio.

Чтобы перевести БД в архивный режим средствами DBA Studio необходимо выполнить следующую последовательность действий:

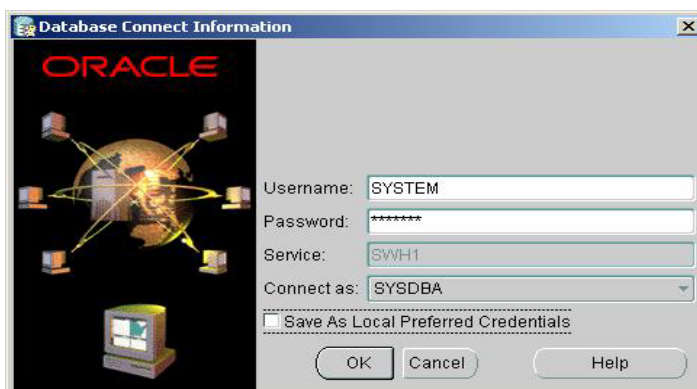
1. Запустить DBA Studio в самостоятельном режиме работы.



2. Присоединиться к БД с правами SYSDBA.

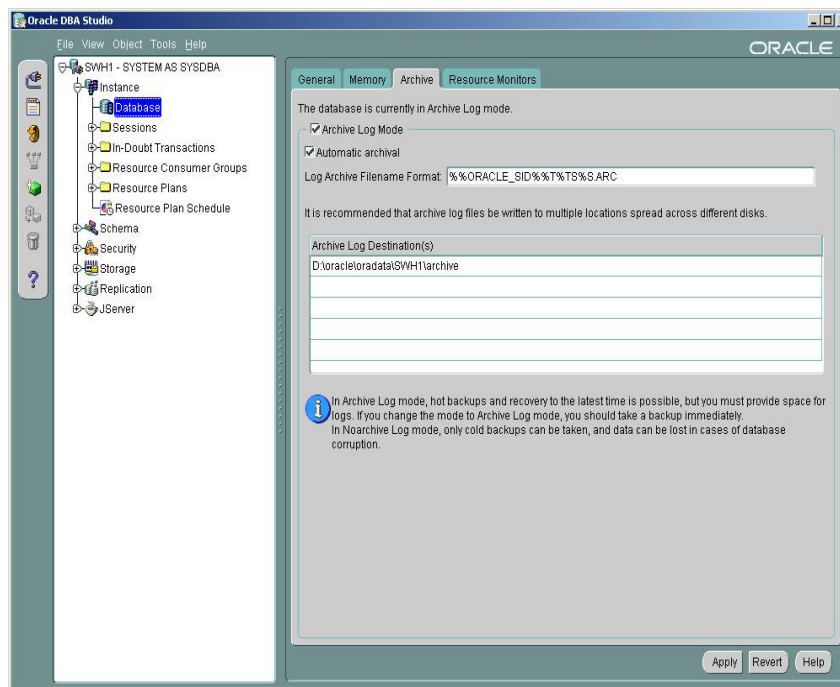
Для этого выбрать в контекстном меню для необходимой БД пункт **Connect...** и в появившемся диалоговом окне задать имя пользователя и пароль с правами SYSDBA.

Таковыми правами обладает пользователь **system** с паролем по умолчанию **manager**.



3. В дереве объектов БД в узле **Instance** выбрать элемент **Database**. В правой панели выбрать закладку **Archive** и поставить галки в пунктах **Archive log mode** и **Automatic archival**. Для повышения надежности в списке **Archive Log Directory** можно задать несколько каталогов для дублирования файлов журналов транзакций. Желательно чтобы эти директории физически находились на разных дисках, а диски обладали достаточным запасом свободного места.

4. Чтобы применить изменения необходимо нажать кнопку **Apply**.



## 5.2. Перевод БД в архивный режим с помощью Server Manager.

Чтобы перевести базу в архивный режим работы необходимо произвести две операции:

1. Настроить архивацию журналов транзакций.
2. Перевести БД в режим ARCHIVELOG.

### 5.2.1. Настройка процесса архивации журналов транзакций.

Чтобы запустить процесс архивации необходимо добавить в файл параметров инициализации INIT.ORA следующие строки:

```
log_archive_start = true
```

*// Параметр включает процесс архивации журналов транзакций*

```
log_archive_dest_1 = "location=D:\oracle\oradata\SWH1\archive"
```

*// Параметр указывает директорию, в которую будут складываться архивные журналы транзакций*

```
log_archive_format = %%ORACLE_SID%%T%%TS%%S.ARC
```

*// Параметр определяет формат имен файлов, в которые будут помещаться архивные копии журналов транзакций*

Обычно эти строки уже присутствуют в файле параметров и их необходимо только соответствующим образом отредактировать.

### 5.2.2. Перевод БД в режим ARCHIVELOG.

Чтобы перевести БД в режим **ARCHIVELOG** необходимо выполнить следующую последовательность операций:

1. В окне консоли запустить Server Manager командой **svrmgrl**.
2. Подключиться к локальному экземпляру Oracle с помощью команды **connect internal**.
3. Остановить экземпляр БД командой **shutdown immediate**.
4. Выполнить команду **startup mount**.
5. Командой **alter database archivelog;** перевести БД в архивный режим.
6. Перевести БД в оперативный режим командой **alter database open;**

С этого момента база данных становится доступной для пользователей и работает в архивном режиме.

Для проверки установленных режимов архивации журналов транзакций необходимо воспользоваться командой **archive log list**.

## 6. Что нужно архивировать?

В процессе работы экземпляр БД использует несколько групп файлов, которые следует архивировать для последующего восстановления БД.

1. Файлы конфигурации экземпляра БД.
2. Файлы, отвечающие за настройки начальных параметров инициализации экземпляра БД, сетевые настройки, настройки безопасности.
3. Управляющие файлы.
4. Файлы, содержащие информацию о схеме данных конкретного экземпляра БД.
5. Файлы журналов транзакций.
6. Файлы, содержащие все изменения или транзакции, произведенные в БД.
  - **Онлайновые (реального времени)** – журналы транзакций, с которыми идет работа в данный момент.
  - **Архивные** – заполненные файлы онлайн-журналов транзакций, которые хранятся в специальной директории и используются для восстановления целостности данных в режиме ARCHIVELOG.
7. Файлы табличных пространств.

Следующая таблица показывает, как называются и где располагаются файлы различных групп.

Группа Файлов	Наименования и местонахождение
Файлы конфигурации.	<ul style="list-style-type: none"> <li>• Параметры инициализации.</li> <li>• Файл: Init.ora</li> <li>• Путь: disk:\Oracle\admin\%SID%\pfile\init.ora</li> <li>• Сетевые настройки.</li> <li>• Файлы: listener.ora, tnsnames.ora, sqlnet.ora.</li> <li>• Путь: disk:\%HOME_DIR%\NETWORK\ADMIN\</li> <li>• Файлы паролей.</li> <li>• Как правило, это файлы с расширением *.ora, имя которых начинается с символов PWD. Например: PWDswh1.ora.</li> <li>• Путь: disk:\%HOME_DIR%\DATABASE\</li> </ul> <p><b><u>В описании путей:</u></b></p> <p><b>disk</b> – буква диска, на котором располагается файл</p> <p><b>%SID%</b> – идентификатор экземпляра БД.</p> <p><b>%HOME_DIR%</b>- домашняя директория БД.</p> <p><b><u>Примечание:</u></b></p> <p>В принципе можно архивировать все файлы с расширением *.ora, которые найдутся в корневой директории БД disk:\Oracle\</p>
Управляющие файлы.	<p>Самый простой способ определить путь и названия управляющих файлов, это найти в файле INIT.ORA строку <b>control_files</b>, в которой будут перечислены используемые этим экземпляром управляющие файлы. На самом деле это копии одного и того же файла.</p> <p><b><u>Например:</u></b> control_files= ("e:\oracle\oradata\sw1\control01.ctl", "e:\oracle\oradata\sw1\control02.ctl", "e:\oracle\oradata\sw1\control03.ctl")</p> <p>Далее будет приведен еще один способ.</p>
Журналы транзакций	<ul style="list-style-type: none"> <li>• Онлайнные (реального времени)</li> <li>• Файлы с именем REDOxx.log</li> <li>• Путь: disk:\%BASE_DIR%\ORADATA\%SID%\</li> <li>• Архивные.</li> <li>• Путь: disk:\%BASE_DIR%\ORADATA\%SID%\archive</li> </ul>
Табличные пространства	<p>Путь: disk:\%BASE_DIR%\ORADATA\%SID%\</p>

### 6.1.1. Дополнительные способы определения имен и путей к файлам различных групп.

#### 1. Управляющие файлы.

2. Для того чтобы определить названия и пути к управляющим файлам необходимо в SQL Plus выполнить следующий запрос:

```
SELECT value FROM v$parameter
WHERE name = 'control_files';
```

В результате работы которого получится следующий отчет:

```
VALUE
```

```
-----
-----
/oracle/dbs/cf1.f, /oracle/dbs/cf2.f
```

#### 3. Файлы табличных пространств.

Название табличных пространств и пути к ним можно посмотреть с помощью “Администратора БД PSS для Oracle” или в SQL Plus выполнить следующий запрос:

```
SELECT t.name "Tablespace", f.name "Datafile"
FROM v$tablespace t, v$datafile f
WHERE t.ts# = f.ts#
ORDER BY t.name;
```

В результате работы этого запроса получится подобный отчет:

Tablespace	Datafile
-----	-----
SYSTEM	/oracle/dbs/tbs_01.f
SYSTEM	/oracle/dbs/tbs_02.f
SYSTEM	/oracle/dbs/tbs_03.f
TBS_1	/oracle/dbs/tbs_11.f
TBS_1	/oracle/dbs/tbs_12.f
TBS_2	/oracle/dbs/tbs_21.f
TBS_2	/oracle/dbs/tbs_22.f
TBS_2	/oracle/dbs/tbs_23.f
TBS_2	/oracle/dbs/tbs_24.f

#### 4. Файлы онлайн-журналов транзакций.

5. Чтобы узнать имена онлайн-журналов транзакций и пути к ним, необходимо в SQL Plus выполнить следующий запрос:

6. **SELECT member FROM v\$logfile;**



7. В результате работы этого запроса получится подобный отчет:

MEMBER

-----

E:\ORACLE\ORADATA\SWH1\REDO03.LOG

E:\ORACLE\ORADATA\SWH1\REDO02.LOG

E:\ORACLE\ORADATA\SWH1\REDO01.LOG

8. Пути к файлам архивных журналов транзакций.

9. Чтобы определить пути к папкам , где хранятся архивные журналы транзакций, необходимо в SQL Plus выполнить следующий запрос:

10. **SELECT destination FROM v\$archive\_dest**

11. **where status='VALID';**

В результате работы этого запроса получится отчет вида:

DESTINATION

-----

e:\oracle\oradata\swh1\archive

## 7. Стратегия архивации.

Под стратегией архивации подразумевается комплекс мероприятий по архивации файлов экземпляра БД с использованием тех или иных инструментальных средств БД Oracle. При разработке стратегии стоит обратить внимание на следующие пункты:

1. Какой способ архивации выбрать.
2. Какие файлы БД архивировать.
3. Куда архивировать файлы (на отдельный диск, на удаленный сервер).
4. Как часто производить архивацию.
5. Рассмотреть возможность автоматизации процесса архивации.

## 8. Сравнение способов архивации.

Способ архивации	Достоинства	Недостатки
Архивация средствами и ОС.	При архивации и восстановлении используются стандартные возможности Oracle и ОС.	При архивации и восстановлении необходимо точно знать, какие файлы конкретно необходимо сохранять и восстанавливать и за что они отвечают.
Импорт / Экспорт.	<ol style="list-style-type: none"> <li>1. Полная архивация и восстановление всех данных и схемы БД.</li> <li>2. Возможность выполнять импорт/экспорт средствами «Администратора БД PSS для Oracle»</li> </ol>	<ol style="list-style-type: none"> <li>1. Длительный процесс.</li> <li>2. Каждый архив содержит все данные БД и поэтому имеет большой размер.</li> <li>3. Надо вручную архивировать файлы конфигурации.</li> </ol>
Архивация средствами и RMAN.	<ol style="list-style-type: none"> <li>1. Автоматический учет архивной информации.</li> <li>2. Богатые возможности по архивации и восстановлению различных объектов БД.</li> <li>3. Возможность хранения архивной информации в отдельной БД.</li> </ol>	<ol style="list-style-type: none"> <li>1. Взаимодействие пользователя с RMAN возможно только посредством специального командного языка.</li> <li>2. Отсутствие встроенной возможности настройки процедуры автоматической архивации данных.</li> <li>3. Надо вручную архивировать файлы конфигурации.</li> </ol>

## 9. Архивация средствами ОС.

Архивация средствами ОС подразумевает «ручное» резервное копирование всех рабочих файлов БД, таких как:

1. Файлы табличных пространств.
2. Управляющие файлы.
3. Файлы журналов транзакций.
4. Файлы конфигурации.

### 9.1. Архивация «на холодную».

Архивация БД «на холодную» производится при остановленном экземпляре БД, при этом работа пользователей с ней не возможна.

Процесс архивации заключается в простом копировании управляющих файлов, файлов табличных пространств, конфигурации, архивных журналов транзакций в резервную директорию или на резервный сервер.

## 9.2. Архивация «на горячую».

### 9.2.1. Архивация файлов табличных пространств.

Операция архивации файла табличного пространства «на горячую» возможна только тогда, когда БД работает в архивном режиме. Она состоит из следующих этапов:

1. Перевод табличного пространства в режим резервного копирования.
2. Создание резервной копии файла табличного пространства.
3. Вывод табличного пространства из режима резервного копирования.
4. Принудительный сброс контрольной точки для синхронизации процесса регистрации резервной копии внутренними средствами БД.

Выполнение этих этапов реализует следующая последовательность команд SQL и операционной системы:

1. **ALTER TABLESPACE tab\_name BEGIN BACKUP;** - перевод табличного пространства tab\_name в режим резервного копирования. В этом режиме БД остается доступной для пользователей, а все изменения данных заносятся в журнал транзакций.
2. Средствами операционной системы копируем файлы табличного пространства в директорию, где мы их собираемся хранить. Для повышения надежности рекомендуется создавать дополнительную копию файлов на удаленном сервере.
3. **ALTER TABLESPACE tab\_name END BACKUP;** - вывод табличного пространства tab\_name из режима резервного копирования.
4. **ALTER SYSTEM CHECKPOINT;** - принудительный сброс контрольной точки.

Последовательность таких команд необходимо выполнить для каждого архивируемого табличного пространства.

### 9.2.2. Архивация управляющего файла.

Для архивации управляющего файла необходимо выполнить команду

```
ALTER DATABASE BACKUP CONTROLFILE TO 'control_filename' REUSE;
```

где control\_filename – путь к резервной копии управляющего файла.

### 9.2.3. Архивация файлов журналов транзакций.

Для архивации файлов журналов транзакций необходимо скопировать каталог, в который БД помещает архивные копии журналов транзакций в процессе своей работы. Перед тем как копировать файлы, необходимо сбросить в архив текущие онлайн-журналы транзакций. Для этого необходимо в SQLPlus или в Server Manager выполнить команду **ALTER SYSTEM ARCHIVE LOG CURRENT;**

### 9.2.4. Архивация файлов конфигурации.

Архивация конфигурационных файлов БД производится вручную. Для этого можно использовать любой из имеющихся архиваторов или просто хранить копии конфигурационных файлов в отдельном каталоге. Какие конфигурационные файлы нужно архивировать и где они находятся можно узнать из раздела «Что нужно архивировать?».

## 10. Восстановление средствами ОС.

Для восстановления поврежденной при сбое БД необходимо остановить экземпляр БД и переписать резервные копии рабочих файлов и журналов транзакций на прежнее место, после чего производить полное или частичное восстановление.

### 10.1. Полное восстановление БД.

Для полного восстановления БД необходимо произвести следующую последовательность действий:

1. Запустите в консоли Oracle Server Manager командой **svrmgrl**
2. Присоединитесь к экземпляру БД командой **connect internal** и **startup mount**
3. Командами **shutdown immediate** и **startup mount** смонтируйте БД. Подробнее о состояниях БД читайте в разделе «Состояния экземпляра БД».
4. Уточните текущее состояние БД командой **alter database open;**
5. Введите команду **recover database;**
6. После этого Oracle начнет восстановление БД с использованием журналов транзакций и выдаст подобный диалог, где жирным шрифтом обозначена команда, введенная пользователем:

```
SVRMGR> recover database
```

```
ORA-00279: change 306948 generated at 06/26/2003 20:39:54 needed for thread 1
```

```
ORA-00289: suggestion:
```

```
D:\ORACLE\ORADATA\SWH1\ARCHIVE1\SWH1T001S00239.ARC
```

```
ORA-00280: change 306948 for thread 1 is in sequence #239
```

```
Specify log: {<RET>=suggested | filename | AUTO | CANCEL}
```

```
auto
```

```
Log applied.
```

ORA-00279: change 307078 generated at 06/26/2003 20:42:38 needed for thread 1  
ORA-00289:suggestion:  
D:\ORACLE\ORADATA\SWH1\ARCHIVE1\SWH1T001S00240.ARC  
ORA-00280: change 307078 for thread 1 is in sequence #240  
ORA-00278:log file D:\ORACLE\ORADATA\SWH1\ARCHIVE1\SWH1T001S00239.ARC'  
no long  
er needed for this recovery  
Log applied.

В процессе диалога БД будет предлагать для восстановления очередной файл журнала транзакций. Чтобы не подтверждать каждый файл в каталоге, если их много, на очередной запрос нужно ответить AUTO . Тогда восстановление базы пойдет дальше в автоматическом режиме.

## 10.2. Частичное восстановление БД.

В процессе восстановления БД с помощью журналов транзакций может возникнуть ситуация, когда есть не все файлы, необходимые для полного восстановления (часть файлов потеряна). В таком случае можно восстановить состояние БД на момент записи архивного журнала транзакций с возможным наибольшим номером. Однако целостность данных в БД при этом теряется.

Существует несколько способов неполного восстановления БД. Рассмотрим два из них: по номеру изменения, и с помощью команды CANCEL.

### 10.2.1. Частичное восстановление по номеру изменения.

Частичное восстановление по номеру изменения применяется тогда, когда в нашем распоряжении имеются не все файлы архивных журналов транзакций, необходимые для полного восстановления целостности БД. В этом случае из имеющихся файлов необходимо выбрать журнал транзакций с наибольшим номером в непрерывной последовательности. Далее необходимо определить номер последнего изменения записанного в этот журнал. Для того чтобы это выяснить, необходимо воспользоваться представлением **v\$log\_history**, в столбце **next\_change#** которого содержится номер первого изменения, которое будет записано в следующий по номеру журнал транзакций. Уменьшив этот номер на единицу, мы получим номер последнего изменения записанного в нужный нам файл транзакций.

Чтобы определить номер последнего изменения необходимо в Server Manager или SqlPlus выполнить следующий запрос:

```
select next_change#-1 from v$log_history where sequence#=<seq_number>;
```

где **seq\_number**- максимальный номер журнала транзакций в непрерывной последовательности.

Дальнейший процесс восстановления в основном аналогичен процессу полного восстановления:

1. Запустить в консоли Server Manager командой **svrmgrl**.
2. Введите команды **connect internal** и **startup mount**.
3. Ввести команду **recover database until change scn\_number**; где **scn\_number** – номер последнего изменения.
4. Получив запрос Oracle на подтверждение первого архивного журнала транзакций, вводим **AUTO**, после чего процесс восстановления продолжается автоматически до указанного изменения.
5. После частичного восстановления БД её необходимо открыть с помощью команды **alter database open resetlogs**;
6. **Внимание:** После открытия БД командой **resetlog** все её резервные копии, созданные до момента её повторного запуска, оказываются несовместимыми с текущим состоянием. Поэтому после восстановления необходимо сделать новую резервную копию БД.

#### 10.2.2. Частичное восстановление с помощью команды **CANCEL**.

При этом способе восстановление БД продолжается до тех пор, пока на приглашение ввести имя файла очередного журнала транзакций будет введено слово **CANCEL**.

Команда, запускающая процесс восстановления, выглядит так:

**recover database until cancel**; Открытие БД производится в режиме **resetlogs** (как и в предыдущем случае).

## 11. Архивация и восстановление при помощи экспорта и импорта данных.

Архивацию и восстановления БД можно производить с помощью стандартных механизмов экспорта и импорта в Oracle. Для повышения надежности сохранности данных необходимо периодически, в зависимости от интенсивности работы с БД, производить полный экспорт. При достаточно интенсивном внесении изменений в данные, необходимо делать экспорт каждые 2 недели. Наличие экспортного файла БД поможет при её восстановлении в случае утери части архивных копий файлов БД.

Экспорт и импорт производится с помощью стандартных возможностей «Администратора БД PSS для Oracle».

## 12. Архивация и восстановление средствами Recovery Manager(RMAN) .

### 12.1. Настройка переменных окружения для работы RMAN (Oracle 8.1.7).

Перед началом работы с RMAN необходимо настроить переменные окружения Oracle NLS\_LANG и NLS\_DATE\_FORMAT устанавливающие правильную интерпретацию формата записи даты. Для этого необходимо в разделе реестра

`\HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\HOME_N`

где **HOME\_N** – раздел с настройками для экземпляра Oracle, с которым вы работаете, создать или отредактировать следующие переменные таким образом:

`NLS_LANG = AMERICAN_AMERICA.CL8MSWIN1251`

`NLS_DATE_FORMAT = 'DD/MM/YYYY'`

Для удобства занесения этих параметров в реестр можно создать текстовый файл с расширением REG и поместить в него следующий текст:

#### Windows Registry Editor Version 5.00

[HKEY\_LOCAL\_MACHINE\SOFTWARE\ORACLE\HOME\_N]

"NLS\_LANG"="AMERICAN\_AMERICA.CL8MSWIN1251"

"NLS\_DATA\_FORNAT"="DD/MM/YYYY"

Теперь, чтобы внести эти параметры в реестр, достаточно кликнуть двойным щелчком мыши по этому файлу.

### 12.2. Основные понятия RMAN.

В число основных понятий RMAN входит следующее:

- **Целевая БД (target database)** – БД, для которой снимается резервная копия, или которая восстанавливается по ранее снятой копии.
- **Резервный набор (backup set)** – Набор файлов, сформированный во время резервного копирования.
- **Резервный файл (backup piece)** – Двоичный файл с резервной информацией.

- **Канал (channel)** - сервисный процесс, возникающий при установлении связи с устройством ввода-вывода (диск или магнитная лента) для записи или чтения файлов резервирования.

### 12.3. Режимы работы RMAN.

Recovery Manager может работать в двух различных конфигурациях отличающихся местом хранения служебной информации об архивируемых объектах:

1. С использованием внешнего хранилища (каталога).
2. С использованием управляющего файла архивируемой БД.

В первой конфигурации в качестве хранилища используется табличное пространство отдельной БД. Такая конфигурация повышает надежность хранения данных об архивах, но требует предварительной настройки хранилища и архивации БД каталога. Вторая конфигурация не требует предварительной настройки и поэтому более удобна для использования. Однако в этом случае необходимо применять дополнительные меры повышения сохранности управляющего файла, так как в случае его потери будет невозможно воспользоваться RMAN для восстановления БД.

### 12.4. Создание БД каталога для RMAN.

Табличное пространство каталога в принципе может храниться в любом экземпляре БД. Однако использование для этого отдельной БД, которая может располагаться на другом сервере, повышает надежность хранения информации об архивных файлах целевой БД.

#### 12.4.1. Создание БД каталога.

Для того чтобы создать БД каталога необходимо следующее:

1. В базе данных, в которой планируется хранить каталог (например, с именем CATALOG), необходимо создать табличное пространство (например, с именем CATTBS). Это можно сделать средствами DBA Studio.
2. Так же эта БД должна иметь временное табличное пространство с именем TEMP, которое, как правило, уже создано по умолчанию.
3. Средствами DBA Studio необходимо создать в этой БД пользователя владельца каталога и указать для него имя и пароль (например, имя и пароль rman ), основное табличное пространство (в нашем примере CATTBS) и временное табличное пространство TEMP. В SQLPlus для этого нужно выполнить следующую команду:

```
CREATE USER rman IDENTIFIED BY rman  
TEMPORARY TABLESPACE temp  
DEFAULT TABLESPACE cattbs  
QUOTA UNLIMITED ON cattbs;
```



4. Назначить пользователю **rman** роли RECOVERY\_CATALOG\_OWNER, CONNECT, RESOURCE. В SQLPlus это делается с помощью команды:

**GRANT recovery\_catalog\_owner, connect, resource TO rman;**

#### 12.4.2. Инициализация каталога и регистрация целевой БД.

Для инициализации каталога в его БД необходимо выполнить следующую последовательность действий:

1. Присоединиться с помощью RMAN к БД каталога, используя пользователя **rman**. Для этого нужно запустить RMAN со следующей командной строкой:

**rman catalog user/pass@catalog\_alias**

или ввести в командной строке RMAN команду:

**connect catalog user/pass@catalog\_db\_alias**

где:

*user* – имя пользователя владельца каталога

*pass*-пароль пользователя владельца каталога

*catalog\_db\_alias*-идентификатор БД каталога, используемый для подключения к ней.

Для нашего примера команда подключения к каталогу будет выглядеть так:

**connect catalog rman/rman@catalog**

2. Создать каталог в табличном пространстве CATTBS командой:

**create catalog tablespace cattbs;**

3. Теперь для регистрации в каталоге целевой БД, необходимо присоединиться к ней с помощью команды:

**connect target system/manager@my\_base**

где *my\_base* – идентификатор целевой БД.

4. Смонтировать целевую БД, если она не смонтирована, командой:

**startup mount;**

5. Зарегистрировать целевую БД в каталоге командой:

**register database;**

#### 12.4.3. Подключение RMAN к каталогу.

1. Чтобы запустить RMAN в режиме с использованием каталога необходимо в консоли выполнить команду: **rman catalog rman/rman@catalog**

2. где *rman/rman@catalog* - логин/пароль пользователя владельца каталога и идентификатор БД каталога.

3. Затем необходимо присоединиться к архивируемой БД. Для этого надо в командной строке RMAN набрать команду **connect target user/pass@db\_alias**
4. где user – имя пользователя с правами DBA
5. pass – пароль пользователя
6. db\_alias – идентификатор БД, используемый для подключения к ней.
7. По умолчанию для подключения к целевой БД можно использовать пользователя **system** с паролем **manager** .
8. После этого можно приступить к работе с RMAN.

#### 12.4.4. Работа с RMAN без каталога.

1. Чтобы запустить RMAN в режиме без каталога необходимо в консоли выполнить команду: **rman nocatalog**.
2. После этого необходимо присоединиться к архивируемой БД. Для этого надо в командной строке RMAN набрать команду **connect target user/pass@db\_alias**  
где user – имя пользователя с правами DBA  
pass – пароль пользователя  
db\_alias – идентификатор БД, используемый для подключения к ней.  
По умолчанию для подключения к целевой БД можно взять пользователя **system** с паролем **manager** .
3. После этого можно приступить к работе с RMAN.

### 12.5. Стратегия архивации с помощью RMAN.

RMAN позволяет делать архивные копии следующих объектов БД:

- Базы данных полностью (всех файлов табличных пространств и текущего управляющего файла)
- Отдельных табличных пространств.
- Отдельно файлов табличных пространств.
- Файлов архивных журналов транзакций (когда БД работает в архивном режиме)
- Управляющего файла.

Хотя RMAN позволяет архивировать и восстанавливать файлы табличных пространств, журналов транзакций и управляющие файлы, но работоспособность БД зависит также и от других файлов, например файлов параметров инициализации, файлов паролей,

файлов сетевой конфигурации БД и т.д. Поэтому необходимо создавать архивные копии этих файлов независимо от RMAN.

С помощью RMAN можно делать архивные копии как **открытой** (доступной для пользователей) БД, так и **закрытой** (смонтированной и не доступной для пользователей) БД. Когда БД закрыта, архив может быть **непротиворечивым** или **противоречивым**, в зависимости от того, как БД была остановлена. Архивы сделанные с открытой БД, когда она работает в архивном режиме, всегда противоречивые и при восстановлении требуют дополнительной корректировки целостности данных с помощью архивных журналов транзакций. Непротиворечивые архивы получаются в том случае, когда БД была корректно остановлена и затем смонтирована без ошибок, такой архив называется *холодным*, и не требует дополнительной корректировки целостности данных после восстановления файлов БД.

Также RMAN позволяет делать **полные** и **добавочные** архивы. В полный архив помещаются все блоки архивируемого файла, а в добавочный только те блоки файла, которые изменились с момента проведения предыдущего добавочного архива.

Рассмотрим наиболее интересные стратегии архивации:

1. Полная архивация БД, работающей в архивном режиме, и ежедневная архивация новых журналов транзакций.
2. Добавочная архивация БД.

## 12.6. Полная архивация БД и журналов транзакций.

Одним из основных критериев при выборе той или иной стратегии архивации БД, является минимизация размера получаемых архивов. Для БД работающей в архивном режиме (с копированием заполненных журналов транзакций в специальный каталог) стратегия архивации должна включать следующие действия:

1. Предварительная архивация БД целиком (все табличные пространства и управляющий файл).
2. Периодическая архивация накопившихся журналов транзакций.

### 12.6.1. Архивация вспомогательных файлов.

Для того чтобы можно было восстановить БД после сбоя диска, когда придется переустанавливать сам Oracle или создавать БД заново, необходимо дополнительно архивировать вручную конфигурационные файлы БД. Где они находятся, следует читать в разделе “Что нужно архивировать?”.

### 12.6.2. Архивация БД полностью.

Чтобы заархивировать БД полностью (файлы табличных пространств и управляющий файл) необходимо выполнить следующую последовательность действий:

1. Запустить RMAN и присоединиться к архивируемой БД.
2. Для того чтобы получить непротиворечивый архив, необходимо чтобы база данных была остановлена без ошибок (командой **shutdown immediate**, а не **shutdown abort**), а потом смонтирована командой **startup mount**. Однако если БД работает в архивном режиме, то можно делать архив «на горячую» и он будет противоречивым, но при восстановлении будет скорректирован с помощью архивных журналов транзакций. Чтобы можно было так сделать, необходимо также производить резервное копирование журналов транзакций.
3. Выделить необходимое количество каналов типа **disk** для работы с диском командой **allocate channel t1 type disk;**
4. Задать с помощью опции **format** путь, по которому следует сохранить полученный архив и формат имен файлов архива. Если не задана опция **format**, то RMAN назначит архивным файлам уникальные имена и положит в каталог по умолчанию.
5. Для повышения надежности архивные копии желательно хранить либо на отдельном диске, либо на удаленном сервере. Можно сразу настроить архивацию на удаленный сервер по сети. Для этого необходимо на удаленном сервере создать сетевую папку с правами на запись и указать эту папку в опции **format**. Например: **format \\my\_server\backup\%U;** - архивировать в сетевую папку backup на сервере my\_server. Ключ %U заставляет RMAN генерировать уникальные имена файлов для полученных архивов.

Эти действия реализует следующая последовательность команд:

```
run {  
    allocate channel ch1 type disk;  
    backup database;  
}
```

или при использовании опции **format**:

```
run {  
    allocate channel ch1 type disk;  
    backup database  
    format '\oracle\backup\%U'; # %U – генерировать уникальные имена  
}
```

Информацию о полученном архиве можно посмотреть с помощью команды:

```
list backup;
```

### 12.6.3. Архивация файлов журналов транзакций.

Для успешного восстановления БД после сбоя необходимо иметь архив журналов транзакций. Для архивации журналов транзакций необходимо выполнить следующую последовательность действий:

1. Запустить RMAN в режиме **nocatalog** и присоединиться к архивируемой БД.
2. Смонтировать или открыть БД.
3. Синхронизировать архивацию текущего журнала транзакций.
4. Выделить канал типа **disk** для работы с диском.
5. Чтобы удалить архивированные журналы транзакций необходимо использовать опцию **delete input** команды **backup**.

Эти действия реализует следующая последовательность команд:

```
run {  
    allocate channel t1 type disk;  
    sql "ALTER SYSTEM ARCHIVE LOG CURRENT";  
    backup archivelog all  
    delete input;  
}
```

При интенсивной работе с БД необходимо ежедневно архивировать журналы транзакций. Чтобы автоматизировать этот процесс необходимо написать командный файл для RMAN и настроить планировщик задач Windows для ежедневного его выполнения.

Например, текст командного файла **backup\_arc\_log.rcm** для архивации журналов транзакций выглядит следующим образом:

```
connect target system/manager@testbase  
run {  
    allocate channel ch1 type disk;  
    backup archivelog all;  
}  
exit
```

где

**testbase** - идентификатор целевой БД( alias ), используемый для подключения к ней.

Вид командной строки RMAN, для выполнения этого командного файла:

```
RMAN @backup_arc_log.rcm nocatalog >> backup.log
```

Лог работы команды будет записываться в файл **backup.log**.

## 12.7. Добавочная архивация БД.

Добавочная архивация может выполняться как в обычном, так и в архивном режиме работы БД. При использовании добавочной архивации используется понятие уровня архивации (от 0 до 4). В начале использования добавочной архивации производится полное резервное копирование с уровнем 0. При проведении последующих архиваций, в зависимости от уровня, в архив попадают только блоки файлов, в которые изменились относительно архивов с меньшим уровнем. Существует два типа добавочных архивов: разностный и накапливающий. В разностный архив уровня N попадают блоки, которые изменились относительно ранних архивов с уровнем  $\leq N$ . В накапливающий архив попадают измененные блоки из более ранних архивов с уровнем  $< N$ .

Для создания добавочного архива, необходимо произвести такую последовательность действий:

1. Запустить RMAN и присоединиться к БД.
2. Если БД работает не в архивном режиме остановить и смонтировать её командами **shutdown immediate** и **startup mount**, иначе можно не монтировать.
3. При первой архивации сделать полный архив БД с уровнем 0. В последующие архивации создавать добавочные разностные архивы с необходимым уровнем (допустим с первым).

Последовательность команд для добавочной архивации с уровнем 0:

```
run {  
    allocate channel ch1 type disk;  
    backup  
    incremental level = 0  
    database;  
}
```

Для создания архива с большим уровнем необходимо задать нужное значение параметра **level**.

## 12.8. Восстановление БД.

Процесс восстановления БД сильно отличается в зависимости от повреждений файлов БД. Рассмотрим основные случаи.

### 12.8.1. Восстановление при отказе диска.

В случае отказа диска и потере рабочих исполняемых файлов БД, файлов конфигурации, файлов табличных пространств и управляющих файлов восстановление БД производится так:

1. Устанавливаем Oracle заново (если это требуется) и заново создаем экземпляр БД с тем же именем и SID –ом, с теми же путями что были до сбоя. Файлы архивов должны находиться в той же папке что и до сбоя.
2. Переписываем файлы INIT.ORA и файл паролей в исходные папки. Чтобы была возможность работать без временного табличного пространства APL\_USER\_TABLESPACE\_TMP, необходимо в файле **init.ora** увеличить значения параметров **sort\_area\_size** и **sort\_area\_retained\_size** примерно в 10 раз (после восстановления временного табличного пространства параметрам можно вернуть прежние значения).
3. Запускаем RMAN.
  - a. Если мы работаем с БД каталога, то запускаем RMAN и подключаемся к БД каталога и целевой БД:
  - b. **RMAN target sys/change\_on\_install@my\_db catalog rman/rman@catalog**
  - c. Если мы работаем без каталога, то сначала необходимо восстановить управляющие файлы от исходной БД. Количество файлов и пути к ним определяются из файла INIT.ORA исходной БД (раздел «Что нужно архивировать?»). Эти файлы дублируются ,так что для восстановления достаточно скопировать управляющий файл нужное количество раз и переименовать соответствующим образом.
  - d. Теперь запускаем RMAN и присоединяемся к целевой БД
  - e. **RMAN target sys/change\_on\_install@my\_db nocatalog.**
4. Запускаем целевую БД, не монтируя её:
5. **shutdown immediate** и **startup nomount** или **startup force nomount**;
6. При работе с каталогом восстанавливаем управляющий файл и монтируем с ним целевую БД:

```
run {  
  allocate channel ch1 type disk;  
  restore controlfile;  
  alter database mount;  
}
```
7. Так как рабочие файлы журналов транзакций тоже потеряны , то необходимо выполнять восстановление по последнему доступному файлу журналов. Чтобы узнать последний доступный журнал необходимо командой **list backup of archivelog all**;

посмотреть все доступные архивы журналов транзакций и найти там журнал с максимальным номером.

8. Выполняем восстановление БД до последнего доступного журнала транзакций со сбросом последовательности журналов транзакций.

```
run {  
    allocate channel ch1 type disk;  
    restore database;  
    recover database until logseq <max_log> thread 1;  
    alter database open resetlogs;  
}
```

9. где <max\_log> - номер максимального доступного журнала транзакций.

10. Для восстановления временного табличного пространства APL\_USER\_TABLESPACE\_TMP, необходимо сначала открыть БД командой **alter database open**, затем удалить это табличное пространство из схемы БД командой ServerManager

```
DROP TABLESPACE APL_USER_TABLESPACE_TMP;
```

и пересоздать его заново командой

```
CREATE TABLESPACE "APL_USER_TABLESPACE_TMP"
```

```
LOGGING
```

```
DATAFILE 'D:\ORACLE\ORADATA\SWHTEST1\APL_USER_TABLESPACE_TMP' SIZE  
10M TEMPORARY;
```

### 12.8.2. Полное восстановление при потере файлов табличных пространств.

Для полного восстановления БД необходимо произвести следующую последовательность действий:

1. Запустить RMAN.
2. Присоединиться к целевой БД.
3. Остановить и смонтировать БД, команды **shutdown immediate** и **startup mount**.
4. Выделить канал типа **disk** для работы с диском.
5. Восстановить потерянные файлы БД.
6. Произвести корректировку целостности данных с помощью журналов транзакций.
7. Удалить вновь восстановленные журналы транзакций.
8. Произвести повторную архивацию БД.
9. Сделать БД открытой, командой **alter database open**;



В результате получается следующая последовательность команд:

```
run {
  allocate channel ch1 type disk; #выделяем канал
  restore database; #восстанавливаем файлы данных
  recover database; #восстанавливаем данные
}

allocate channel for maintenance type disk; #выделяем канал для удаления
change archivelog all delete; #удаляем файлы журналов транзакций
release channel; #освобождаем канал
```

Если при восстановлении данных RMANу потребуются архивные журналы транзакций, то он автоматически разархивирует необходимые файлы в директорию, где они хранятся по умолчанию и использует их..

Для ручного восстановления журналов транзакций нужно использовать следующую последовательность команд:

```
run {
  allocate channel ch1 type disk;
  restore archivelog all;
}
```

## 12.9. Управление архивом.

- **Просмотр доступных архивных наборов БД** выполняется командой: **list backup;**
- **Просмотр доступных архивных наборов журналов транзакций** выполняется командой: **list backup of archivelog all;**
- **Проверка валидности архивного набора** выполняется следующей последовательностью команд:

```
run {
  allocate channel c1 type disk;
  validate backupset <n>;
}
```

<n> - номер архивного набора.

- **Удаление архивного набора** выполняется следующей последовательностью команд:

**allocate channel for maintenance type disk;**

**change backupset <n> delete;**

**release channel;**

где <n> - номер удаляемого набора или список номеров разделенных запятыми.