



Факультет программной инженерии и компьютерной техники
Распределённые системы хранения данных

Лабораторная работа №3
Вариант №31297

Преподаватель: Шешуков Дмитрий Михайлович
Выполнил: Кульбако Артемий Юрьевич, Сараев Владислав Витальевич Р33112

Задание

Цель работы - настроить процедуру периодического резервного копирования базы данных, сконфигурированной в ходе выполнения лабораторной работы №2. В процессе конфигурации процедуры резервного копирования по-прежнему необходимо пользоваться только интерфейсом командной строки и утилитой SQLPlus; использовать графические утилиты нельзя.

В процессе выполнения работы необходимо создать резервную копию БД на узле db197, настроить процесс репликации, и осуществить процедуру восстановления БД с резервной копии.

Репликацию необходимо организовать посредством периодического применения на реплике изменений из журнала повторов "оригинала".

Требования к настройке резервного копирования:

- Вся логика сервиса, осуществляющего репликацию БД, должна быть реализована в виде shell-скриптов.
- Необходимо реализовать задачу для планировщика cron, осуществляющую периодический (например, раз в час) запуск скрипта репликации.
- Каталог, в котором будет создаваться резервная копия экземпляра Oracle, выбирается на усмотрение студента.
- Для того, чтобы можно было продемонстрировать корректность работы репликации, тестовая база не должна быть пустой. Т.е. предварительно в ней нужно создать тестовые таблицы и заполнить их тестовыми данными, осуществив несколько транзакций.

Выполнение

Создадим тестовые данные в исходной базе:

```
CREATE TABLE test(  
  id INT,  
  name VARCHAR2(20)  
);  
INSERT INTO test VALUES (1, 'lol');
```

Переведем базу в режим ARCHIVELOG:

```
SHUTDOWN;  
STARTUP MOUNT;  
ALTER DATABASE ARCHIVELOG;  
ALTER DATABASE OPEN;
```

Для создания дубликата на узле 197 необходимо создать похожий экземпляр Oracle, но не создавать саму базу данных. Для этого модифицируем скрипт из лабораторной №2 и запустим его в режиме EX_ONLY:

```
# скрипт запускать командой `./configurer.sh` или `source configurer.sh`  
чтобы импорт переменных сработал  
(https://stackoverflow.com/questions/10781824/export-not-working-in-my-shell-
```

```

script)

function drop_db {
# отступы не менять здесь, bash-y не нравится!
echo "УДАЛЕНИЕ БАЗЫ ДАННЫХ"
sqlplus / as sysdba <<EOF
SHUTDOWN ABORT;
EXIT;
EOF
    rm -rf $mount_dir
    rm $ORACLE_HOME/dbs/*
}

function set_envs {
    echo "УСТАНОВКА ПЕРЕМЕННЫХ ОКРУЖЕНИЯ"
    export ORACLE_BASE=/u01/app/oracle
    export ORACLE_HOME=$ORACLE_BASE/product/11.2.0/dbhome_1
    export ORACLE_SID=ku1bako_saraev_p33112
    export PATH=$PATH:$ORACLE_HOME/bin
    export LD_LIBRARY_PATH=$ORACLE_HOME/lib
    export NLS_LANG=American_America.UTF8
    export NLS_SORT=AMERICAN
    export NLS_DATE_LANGUAGE=AMERICAN
    export NLS_DATE_FORMAT="DD.MM.YYYY"
}

function create_dirs {
    echo "ПОДГОТОВКА НЕОБХОДИМЫХ КАТАЛОГОВ"
    mkdir -p $mount_dir # создание точки монтирования
    chown oracle:oinstall $mount_dir # задание прав на точку
    for (( i = 1; i <= 4; i++ ))
    do
        mkdir -p $mount_dir/$db_name/node0$i
    done
    mkdir $mount_dir/$db_name/logs
    recovery_dir=$mount_dir/fra
    mkdir $recovery_dir # создание директории для резервных копий
}

function auth {
    echo "ЗАДАНИЕ МЕТОДА АУТЕНТИФИКАЦИИ АДМИНИСТРАТОРА"
    cd $ORACLE_HOME/dbs # переход в стандартный каталог для конфигов
    orapwd file=ora$ORACLE_SID force=Y # создание файла аутентификации
}

function create_configs {
    echo "СОЗДАНИЕ КОНФИГУРАЦИОННЫХ ФАЙЛЫ, НЕОБХОДИМЫХ ДЛЯ ИНИЦИАЛИЗАЦИИ И
ЗАПУСКА ЭКЗЕМПЛЯРА ORACLE"
    # параметры DB_RECOVERY не нужны для лаб2 и их можно удалить, но пригодятся
в лаб3
    echo "
    DB_NAME=$db_name
    DB_BLOCK_SIZE=4096
    SGA_TARGET=440M
    DB_RECOVERY_FILE_DEST_SIZE=20G
    DB_RECOVERY_FILE_DEST='$recovery_dir'
    " >> init$ORACLE_SID.ora # создание файла инициализации экземпляра
    cd $script_dir
}

```

```

script_dir=$(pwd)
mount_dir=/u01/qvs94
db_name=leftfish
modes=("DROP" "EX_ONLY" "FULL")
set_envs
case ${1} in
    ${modes[0]})
        drop_db
        ;;
    ${modes[1]})
        create_dirs
        auth
        create_configs
        ;;
    ${modes[2]})
        create_dirs
        auth
        create_configs
        echo "ЗАПУСК ЭКЗЕМПЛЯРА ORACLE"
        exit | sqlplus /nolog @mounter.sql
        echo "СОЗДАНИЕ НОВОЙ БАЗЫ ДАННЫХ"
        exit | sqlplus /nolog @db_creator.sql
        echo "СОЗДАНИЕ ДОПОЛНИТЕЛЬНЫХ ТАБЛИЧНЫХ ПРОСТРАНСТВ"
        exit | sqlplus /nolog @tb_creator.sql
        echo "ФОРМИРОВАНИЕ ПРЕДСТАВЛЕНИЯ СЛОВАРЯ ДАННЫХ"
        exit | sqlplus /nolog @view_creator.sql
        ;;
*)
    echo "РЕЖИМ НЕ ЗАДАН, БЫЛИ УСТАНОВЛЕНЫ ПЕРЕМЕННЫЕ ОКРУЖЕНИЯ"
    echo "РЕЖИМЫ:"
    for t in ${modes[@]}; do
        echo $t
    done
    ;;
esac

```

После выполним создание резервной копии на исходном узле (194) и отправим на узел назначения (предварительно лучше создать ssh-ключ, чтобы команды выполнялись автоматически, не запрашивая пароль).

```

# предварительно нужно создать экземпляр oracle и базу данных на узле-
источнике и только экземпляр oracle на узле назначения (запустить
configurer.sh в режиме EX_ONLY)

echo "СОЗДАНИЕ РЕЗЕРВНОЙ КОПИИ"
rman target / << EOF                                     # подключение к rman с помощью
механизма аутентификации OC
SQL 'ALTER SYSTEM ARCHIVE LOG CURRENT'; # принудительно получить последнюю
версию журнала повторов
CONFIGURE CONTROLFILE AUTOBACKUP ON;                    # включить автоматическое копирование
контрольного файла
BACKUP DATABASE PLUS ARCHIVELOG;                          # создание полной резервной копии БД
EXIT;
EOF

echo "ОТПРАВКА БЭКАПА НА УЗЕЛ НАЗНАЧЕНИЯ"
dest=oracle@db197
backup_path=/u01/qvs94/fra

```

```
scp -r $backup_path/LEFTFISH/ $dest:$backup_path
scp restore.sh $dest:/u01/dss3/restore.sh
ssh $dest << EOF
bash
chmod +x ~/dss3/restore.sh
source ~/dss3/restore.sh
EOF
```

Следующий шаг зависит от задачи. При первоначальном дублировании нужно будет выполнить:

```
echo "ВОССТАНОВЛЕНИЕ ИЗ РЕПЛИКИ НА УЗЛЕ НАЗНАЧЕНИЯ ПЕРВЫЙ РАЗ"
# DBID - идентификатор исходной базы данных, нужен для восстановления.
# Получить его можно в выводе команды `rman target /` на исходной базе
rman target / << EOF
SHUTDOWN;
SET DBID 1388105612;           # установка id аналогичному id
# исходной БД
STARTUP NOMOUNT;
RESTORE CONTROLFILE FROM AUTOBACKUP; # восстановить контрольный файл
ALTER DATABASE MOUNT;
CROSSCHECK BACKUP;           # проверки данных для восстановления
# на целостность
CROSSCHECK COPY;
CROSSCHECK ARCHIVELOG ALL;
RESTORE DATABASE;           # восстановить БД
EXIT
EOF
sqlplus / as sysdba << EOF
RECOVER DATABASE UNTIL CANCEL; # откатить/докатить до состояния из
# резервной копии
ALTER DATABASE OPEN RESETLOGS; # сбросить лог, чтобы не возникало
# ошибок записи после восстановления
EXIT;
EOF
```

Все последующие разы выполнять:

```
echo "ВОССТАНОВЛЕНИЕ НА УЗЛЕ НАЗНАЧЕНИЯ ИЗ РЕЗЕРВНОЙ КОПИИ"
# DBID - идентификатор исходной базы данных, нужен для восстановления.
# Получить его можно в выводе команды `rman target /` на исходной базе
rman target / << EOF
SHUTDOWN;
SET DBID 1388105612;
STARTUP NOMOUNT;
RESTORE CONTROLFILE FROM AUTOBACKUP;
ALTER DATABASE MOUNT;
CROSSCHECK BACKUP;
CROSSCHECK COPY;
CROSSCHECK ARCHIVELOG ALL;
RESTORE DATABASE;
RECOVER DATABASE;
ALTER DATABASE OPEN RESETLOGS;
EXIT;
EOF
```

Для периодического копирования создадим задачу в планировщике:

```
crontab -e
0 * * 1 * * source /u01/qvs94/backup.sh
```

Все скрипты с комментариями можно найти на:
<https://github.com/testpassword/Distributed-storage-systems/tree/master/lab3-08.05.21>

Вывод

В процессе выполнения лабораторной работы мы научились создавать дубликат, а также резервные копии базы данных Oracle, что является важной задачей в области хранения данных.