

Национальный исследовательский университет информационных технологий,
механики и оптики.

Кафедра вычислительной техники.
Системы управления базами данных.

Лабораторная работа №2
Основы конфигурирования серверов Oracle SQL.
650 вариант

Работу выполнил студент группы Р3415
Халанский Дмитрий

1. Задание

Цель работы — сконфигурировать базу данных Oracle на выделенном сервере и настроить процедуру периодического резервного копирования сконфигурированной БД. В процессе конфигурации БД и процедуры резервного копирования необходимо пользоваться только интерфейсом командной строки и утилитой SQLPlus; использовать графический установщик нельзя.

1.1. Порядок конфигурации БД

1. Задать значения необходимых для конфигурации переменных окружения.
2. Задать метод аутентификации администратора (зависит от варианта).
3. Создать конфигурационные файлы, необходимые для инициализации и запуска экземпляра Oracle.
4. Запустить экземпляр Oracle.
5. Создать новую базу данных (параметры конфигурации зависят от варианта).
6. Создать дополнительные табличные пространства (определяются вариантом).
7. Сформировать представления словаря данных.

1.2. Параметры конфигурации Oracle

Имя узла db150.

Точка монтирования /u01/rex71.

SID s100000, где s100000 - ID студента.

Метод аутентификации администратора БД файл.

Имя БД bigmath.

Размер блока данных 16384 байт.

Размер SGA 780 МБ.

Кодировка UTF-8.

Файлы данных табличного пространства SYSTEM

- node02/cpiku34.dbf

Файлы данных табличного пространства SYSAUX

- node03/yel82.dbf
- node02/jic65.dbf

Файлы данных табличного пространства USERS

- node03/oyaqebu930.dbf

- node04/uvipura387.dbf

Файлы данных дополнительных табличных пространств

LONG_ORANGE_DISK

- node01/longorangedisk01.dbf
- node03/longorangedisk02.dbf

NICE_GRAY_USER

- node04/nicegrayuser01.dbf
- node02/nicegrayuser02.dbf
- node02/nicegrayuser03.dbf
- node01/nicegrayuser04.dbf
- node04/nicegrayuser05.dbf

По завершении конфигурации БД необходимо создать её резервную копию, настроить процесс репликации, и осуществить процедуру восстановления БД с резервной копии.

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

2. Ход работы

2.1. Общие функции для скриптов

```

1 check_oracle_home() {
2     local home="$1"
3     if [ ! -d "$home" ]; then
4         printf "%s' does not point at an Oracle SQL installation.\n" \
5             "$home" >&2
6         return 5
7     fi
8 }
9
10 check_required_programs() {
11     local m=""
12
13     local p
14     for p in "$@"; do type "$p" >/dev/null 2>&1 || m="{m:+$m_}$p"; done
15
16     if [ ! -z "$m" ]; then
17         printf "Missing_programs:_%s\n" "$m" 1>&2
18         printf 'Please ensure they are present in $PATH.\n' 1>&2
19         return 5
20     fi
21 }
22
23 check_db_name() {
24     local name="$1"
25     local name_len=$(printf "%s" "$name" | wc -c)
26     if [ $name_len -lt 1 -o "$name_len" -gt 8 ]; then
27         printf "The_database_name_should_consist_of_1-8_" >&2
28         printf "characters,_got_%d.\n" "$name_len" >&2

```

```

29         return 5
30     fi
31
32     if ! printf "%s" "$name" | grep '^[a-zA-Z_#]*$' > /dev/null; then
33         printf "The_database_name_should_only_contain_alphabetic_>&2
34         printf "characters,\n'_','_','#','_and_'$'_symbols,\n" >&2
35         printf "got_%s'" "$name" >&2
36         return 5
37     fi
38 }
39
40 get_mounted_fname() {
41     local base="$1"
42     local fname="$2"
43
44     if [ "$fname" = "${fname#/}" ]; then
45         printf "%s\n" "$base/$fname"
46     else
47         printf "%s\n" "$fname"
48     fi | sed 's;/*;g'
49 }
50
51 suggest_editing() {
52     local file_name="$1"
53     local prompt="${2:-Do_you_wish_to_edit_file_$file_name?}"
54
55     local answer
56     read -p "$prompt_[Y/n]_" answer
57
58     if printf "%s" "$answer" | egrep $(locale noexpr) > /dev/null; then
59         return 0
60     fi
61
62     local editors="${VISUAL:-}_${EDITOR:-}_vim_vi_nano_emacs_editor"
63     local ed=""
64
65     for ed in $editors; do
66         if type $ed > /dev/null 2>&1; then
67             break;
68         fi
69     done
70
71     if [ -z "$ed" ]; then
72         printf 'No editor was found on the system; set $EDITOR ' >&2
73         printf 'to point at an editor.\n' >&2
74         return 5
75     fi
76
77     $ed "$file_name"
78 }

```

2.2. shell-скрипт конфигурации БД

```

1  #!/bin/bash -efu --posix
2
3  export LANG=C
4
5  if [ -r common.sh ]; then
6      ./common.sh
7  else
8      printf "Please_call_the_script_from_the_directory_that_contains_it." \
9          >&2
10     exit 5
11 fi

```

```

12
13 check_sga_size() {
14     local size="$1"
15     if ! printf "%s" "$size" | grep '^[0-9][0-9]*[KMG]$\ ' > /dev/null; then
16         printf "%s' is an incorrect SGA_size_specification.\n" \
17             "$size" 1>&2
18         printf "Examples_of_correct_specifications_are_" >&2
19         printf "'768M' and '2G'.\n" >&2
20         return 5
21     fi
22 }
23
24 check_block_size() {
25     local bs="$1"
26     if ! printf "%s" "$bs" | grep '^[0-9][0-9]*$\ ' > /dev/null; then
27         printf "Block_size_must_be_a_number;_got_'%s'_instead.\n" \
28             "$bs" >&2
29         return 5
30     fi
31
32     if [ "$bs" -gt 32768 -o "$bs" -lt 2048 ]; then
33         printf "Block_size_must_be_in_[2048;_32768];_"
34         printf "got_%d.\n" "$bs" >&2
35         return 5
36     fi
37 }
38
39 check_auth_type() {
40     local atype="$1"
41     if [ ! "$atype" = sys -a ! "$atype" = file ]; then
42         printf "Unsupported_auth_type_%s._" "$atype" >&2
43         printf "Specify_either_'sys'_or_'file'_\n" >&2
44         return 5
45     fi
46 }
47
48 print_initial_config() {
49     local MOUNIPOINT="$1"
50     local db_name="$2"
51     local sga="$3"
52     local block_size="$4"
53     cat << EOF
54     db_name='$db_name'
55     sga_max_size=$sga
56     ${block_size:+db_block_size=$block_size}
57     db_recovery_file_dest=$MOUNIPOINT/$db_name/flash_recovery_area'
58     db_recovery_file_dest_size='3G'
59 EOF
60 }
61
62 create_password_file() {
63     local path="$1"
64     local pass="$2"
65     mkdir -p "${path%/*}"
66     orapwd file="$path" password="$pass"
67 }
68
69 generate_initial_sql() {
70     local name="$1"
71     local mpoint="$2"
72     local dbs_file="$3"
73     shift 3
74
75     local tbl_sys=""
76     local tbl_sysaux=""

```

```

77     local tbl_users=""
78     local tbl_undo=""
79
80     local tbls_extra=""
81     local tbl_cur=""
82
83     local in=""
84     local state=o
85     local t
86     for t in "$@"; do
87         case $state in
88             o)
89                 if [ "$t" = '{' ]; then
90                     state=b
91                 else
92                     printf 'Error: expected {\.\n' '>&2
93                     return 5
94                 fi
95                 ;;
96             b)
97                 in=$(printf '%s' "$t" | tr '[:upper:]' '[:lower:]')
98                 state=i
99                 tbl_cur=""
100                ;;
101             i)
102                 if [ "$t" = '}' ]; then
103                     case $in in
104                         sys)
105                             tbl_sys="$tbl_cur"
106                             ;;
107                         sysaux)
108                             tbl_sysaux="$tbl_cur"
109                             ;;
110                         users)
111                             tbl_users="$tbl_cur"
112                             ;;
113                         undo)
114                             tbl_undo="$tbl_cur"
115                             ;;
116                         *)
117                             local st
118                             st="create_tablespace_$in_datafile_$tbl_cur"
119                             tbls_extra=$(printf "$st;\n$tbls_extra")
120                             ;;
121                     esac;
122                     in=''
123                     state=o
124                     tbl_cur=''
125                 else
126                     local m
127                     m=$(get_mounted_fname "$mpoint/$name" "$t")
128                     tbl_cur="{tbl_cur:$tbl_cur,}_$m'_SIZE_100M"
129                     tbl_cur="$tbl_cur_REUSE_AUTOEXTEND_ON"
130                     tbl_cur="$tbl_cur_MAXSIZE_UNLIMITED"
131                 fi
132                 ;;
133             esac
134     done
135
136     if [ "$state" != 'o' ]; then
137         printf "Expected_closing_}'_which_never_came\n" '>&2
138         return 5
139     fi
140
141     echo "create_spfile_from_pfile_=$' $dbs_file ';"

```

```

142     echo "startup_nomount;"
143     echo "create_database_$name"
144     echo "character_set_utf8"
145     echo "archivelog"
146     [ -n "$tbl_sys" ] && echo datafile "$tbl_sys"
147     [ -n "$tbl_sysaux" ] && echo sysaux datafile "$tbl_sysaux"
148     [ -n "$tbl_users" ] && echo default tablespace users datafile \
149         "$tbl_users"
150     [ -n "$tbl_undo" ] && echo undo tablespace undotbs1 datafile \
151         "$tbl_undo"
152     echo ';';
153     printf "%s\n" "$tbls_extra"
154
155     echo "@?/rdms/admin/catalog.sql"
156     echo "@?/rdms/admin/catproc.sql"
157     echo "grant_sysdba_to_sys;"
158     echo "exit"
159 }
160
161 print_help() {
162 cat << EOF
163 Usage: $0 [OPTION]... ('{ ' TABLESPACE DATAFILE... '}' )...
164 Initialize a new Oracle database with the specified parameters.
165
166 There are a few mandatory parameters that must be passed to the script in order
167 for it to work. All of them can be specified in environment variables as well,
168 though if both a variable and the corresponding parameter is set, only the
169 parameter is considered.
170
171 The mandatory parameters and the corresponding variables are:
172 -H DIR          ORACLE_HOME      path to the Oracle installation directory
173 -s SID          ORACLE_SID       the instance identifier of the new instance
174 -M DIR          MOUNTPOINT       the directory to which the database files
175                                 shall be installed
176 -n NAME         DB_NAME          name of the database instance
177 -S MEM          SGA_MAX_SIZE     amount of memory to be used by the SGA,
178                                 a number with suffix G, M, or K for
179                                 gigabytes, megabytes, or kilobytes
180                                 respectively
181 -p PASS         PASSWORD         password for the SYS user
182
183 Additional parameters may also be specified:
184 -b BS          BLOCK_SIZE       size of data block (8192 by default)
185 -a TYPE        AUTH_TYPE       either 'file' or 'sys' for authentication
186                                 based on password files or system accounts
187                                 respectively. 'sys' by default.
188 -i            INTERACTIVE      if the parameter is specified or the variable
189                                 is non-empty, prompt user to manually edit
190                                 the configs before applying them.
191 EOF
192 }
193
194 ORACLE_SID="${ORACLE_SID:-}"
195 ORACLE_HOME="${ORACLE_HOME:-}"
196 MOUNTPOINT="${MOUNTPOINT:-}"
197 DB_NAME="${DB_NAME:-}"
198 SGA_MAX_SIZE="${SGA_MAX_SIZE:-}"
199 PASSWORD="${PASSWORD:-}"
200
201 AUTH_TYPE="${AUTH_TYPE:-sys}"
202 BLOCK_SIZE="${BLOCK_SIZE:-}"
203
204 INTERACTIVE="${INTERACTIVE:-}"
205
206 while getopts 'hH:s:M:a:n:S:b:p:i' name; do

```

```

207         case "$name" in
208             h)
209                 print_help
210                 exit 0
211                 ;;
212             H)
213                 ORACLE_HOME="$OPTARG"
214                 ;;
215             s)
216                 ORACLE_SID="$OPTARG"
217                 ;;
218             M)
219                 MOUNTPOINT="$OPTARG"
220                 ;;
221             a)
222                 AUTH_TYPE="$OPTARG"
223                 ;;
224             n)
225                 DB_NAME="$OPTARG"
226                 ;;
227             S)
228                 SGA_MAX_SIZE="$OPTARG"
229                 ;;
230             b)
231                 BLOCK_SIZE="$OPTARG"
232                 ;;
233             p)
234                 PASSWORD="$OPTARG"
235                 ;;
236             i)
237                 INTERACTIVE=true
238                 ;;
239             ?)
240                 print_help
241                 exit 1
242                 ;;
243         esac
244     done
245     shift $((OPTIND - 1))
246
247     check_sga_size "$SGA_MAX_SIZE"
248     check_oracle_home "$ORACLE_HOME"
249     check_db_name "$SDB_NAME"
250     check_auth_type "$AUTH_TYPE"
251     [ -z "$BLOCK_SIZE" ] || check_block_size "$BLOCK_SIZE"
252
253     PATH=${ORACLE_HOME}/bin${PATH:+:$PATH}
254     check_required_programs sqlplus orapwd
255
256     tmpdir=$(mktemp -d)
257     trap "rm -rf $tmpdir" INT TERM EXIT
258
259     dbs_file="$tmpdir/init.ora"
260     print_initial_config "$MOUNTPOINT" "$SDB_NAME" "$SGA_MAX_SIZE" "$BLOCK_SIZE" \
261     > "$dbs_file"
262     [ -z "$INTERACTIVE" ] ||
263     suggest_editing "$dbs_file" "Do_you_wish_to_edit_the_initial_config?"
264
265     sql_file="$tmpdir/create.sql"
266     generate_initial_sql "$SDB_NAME" "$MOUNTPOINT" "$dbs_file" "$@" > "$sql_file"
267     [ -z "$INTERACTIVE" ] ||
268     suggest_editing "$sql_file" "Edit_the_bootstrap_SQL_script?"
269
270     for i in 1 2 3 4; do mkdir -p "$MOUNTPOINT/$SDB_NAME/node0$i"; done
271     mkdir -p "$MOUNTPOINT/$SDB_NAME/flash_recovery_area"

```



```

272
273 export ORACLE_HOME ORACLE_SID
274
275 auth_file="$ORACLE_HOME/dbs/orapw$ORACLE_SID"
276 [ "$AUTH_TYPE" = sys ] ||
277     create_password_file "$auth_file" "$PASSWORD"
278
279 sqlplus / as sysdba @"$sql_file"

```

2.2.1. Параметры вызова

```

1 ./create.sh -H "$HOME/app/oracle/product/11.2.0/dbhome_1" -s s182349 \
2 -M $HOME/rex71 -n bigmath -S 780M -p test -b 16384 -a file \
3 '{ sys node02/upiku34.dbf }' \
4 '{ undo undo.dbf }' \
5 '{ sysaux node03/yel82.dbf node02/jic65.dbf }' \
6 '{ users node03/oyaqebu930.dbf node04/uvipura387.dbf }' \
7 '{ long_orange_disk node01/longorangedisk01.dbf \
8 node03/longorangedisk02.dbf }' \
9 '{ nice_gray_user node04/nicegrayuser01.dbf \
10 node02/nicegrayuser02.dbf node02/nicegrayuser03.dbf \
11 node01/nicegrayuser04.dbf node04/nicegrayuser05.dbf }'

```

2.3. shell-скрипт конфигурации standby

```

1 #!/bin/bash -exfu --posix
2
3 export LANG=C
4
5 if [ -r common.sh ]; then
6     . ./common.sh
7 else
8     printf "Please call the script from the directory that contains it." \
9         >&2
10    exit 5
11 fi
12
13 create_listener_configs() {
14     local ORACLE_HOME="$1"
15     local rep_sid="$2"
16     local rep_name="$3"
17     local key_handle="$4"
18
19     local tns="$ORACLE_HOME/network/admin/tnsnames.ora"
20     echo "${rep_name}S_(DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)" \
21         "(KEY=\"$key_handle\"))_ (CONNECT_DATA=(SID=${rep_sid})))" \
22         >> "$tns"
23
24     local lora="$ORACLE_HOME/network/admin/listener.ora"
25     echo "${rep_name}L_(ADDRESS_LIST=(ADDRESS=(PROTOCOL=ipc)" \
26         "(KEY=$key_handle)))" >> "$lora"
27     echo "sid_list ${rep_name}L_(SID_LIST=(SID_DESC=(SID_NAME=$rep_sid)" \
28         "(ORACLE_HOME=$ORACLE_HOME)))" >> "$lora"
29 }
30
31 get_replica_control() {
32 ORACLE_SID=$1 sqlplus / as sysdba << EOF
33 alter database create standby controlfile as '$2';
34 exit
35 EOF
36 }
37
38 get_old_pfile() {

```

```

39 ORACLE_SID=$2 sqlplus / as sysdba << EOF
40 create pfile='$1' from spfile;
41 exit
42 EOF
43 }
44
45 edit_pfile() {
46     pfile="$1"
47     ctrl_file="$2"
48     db_mnt="$3"
49     db_name="$4"
50     rep_name="$5"
51     rep_mnt="$6"
52     sed -i -e "s~control_files=.*$~control_files='$ctrl_file '~I" "$pfile"
53     echo >> "$pfile"
54     echo "db_file_name_convert='$db_mnt/$db_name', '$rep_mnt/$rep_name' " \
55         >> "$pfile"
56     echo "log_file_name_convert='$db_mnt/$db_name', '$rep_mnt/$rep_name' " \
57         >> "$pfile"
58     echo "lock_name_space='${rep_name}S' " >> "$pfile"
59 }
60
61 generate_replica_sql() {
62     pfile="$1"
63     echo "create_spfile_from_pfile='$pfile ';"
64     echo "startup_nomount;"
65     echo "alter_database_mount_standby_database;"
66     echo "alter_database_recover_managed_standby_database " \
67         "disconnect_from_session;"
68     echo "exit"
69 }
70
71 generate_primary_sql() {
72     rep_name="$1"
73     echo "alter_system_set_log_archive_dest_2_='SERVICE=${rep_name}S' \
74         'OPTIONAL_REOPEN=180';"
75     echo "alter_system_set_log_archive_dest_state_2_='ENABLE';"
76     echo "exit"
77 }
78
79 print_help() {
80     cat << EOF
81 Usage: $0 [OPTION]...
82 Create a redo standby of a specified Oracle database located on the same
83 machine.
84
85 There are a few mandatory parameters that must be passed to the script in order
86 for it to work. All of them can be specified in environment variables as well,
87 though if both a variable and the corresponding parameter is set, only the
88 parameter is considered.
89
90 The mandatory parameters and the corresponding variables are:
91 -H DIR      ORACLE_HOME    path to the Oracle installation directory
92 -s SID      ORACLE_SID     the instance identifier of the instance
93 -M DIR      MOUNTPOINT     the directory containing the database files
94 -n NAME     DB_NAME        name of the database instance
95 -R DIR      REP_MOUNTPOINT the directory containing the standby instance
96 -r NAME     REP_NAME       unique name of the standby instance
97
98 Additional parameters may also be specified:
99 -i          INTERACTIVE    if the parameter is specified or the variable
100                                is non-empty, prompt user to manually edit
101                                the configs before applying them.
102 EOF
103 }

```

```

104
105 ORACLE_SID="{ORACLE_SID:-}"
106 ORACLE_HOME="{ORACLE_HOME:-}"
107 MOUNTPOINT="{MOUNTPOINT:-}"
108 DB_NAME="{DB_NAME:-}"
109 REP_MOUNTPOINT="{REP_MOUNTPOINT:-}"
110 REP_NAME="{REP_NAME:-}"
111
112 INTERACTIVE="{INTERACTIVE:-}"
113
114 while getopts 'hH:s:M:n:R:r:i' name; do
115     case "$name" in
116         h)
117             print_help
118             exit 0
119             ;;
120         H)
121             ORACLE_HOME="$OPTARG"
122             ;;
123         s)
124             ORACLE_SID="$OPTARG"
125             ;;
126         M)
127             MOUNTPOINT="$OPTARG"
128             ;;
129         n)
130             DB_NAME="$OPTARG"
131             ;;
132         R)
133             REP_MOUNTPOINT="$OPTARG"
134             ;;
135         r)
136             REP_NAME="$OPTARG"
137             ;;
138         i)
139             INTERACTIVE=true
140             ;;
141         ?)
142             print_help
143             exit 1
144             ;;
145     esac
146 done
147 shift $((OPTIND - 1))
148
149 check_oracle_home "$ORACLE_HOME"
150 check_db_name     "$DB_NAME"
151 check_db_name     "$REP_NAME"
152
153 PATH="{ORACLE_HOME}/bin${PATH:+:$PATH}"
154 check_required_programs sqlplus lsnrctl
155
156 tmpdir=$(mktemp -d)
157 trap "rm_rf_$tmpdir" INT TERM EXIT
158
159 mkdir -p "$REP_MOUNTPOINT/$REP_NAME"
160 for i in undo.dbf node01 node02 node03 node04; do
161     o="$MOUNTPOINT/$DB_NAME/$i"
162     d="$REP_MOUNTPOINT/$REP_NAME/$i"
163     [ ! -e "$o" ] || cp -r "$o" "$d";
164 done
165
166 export ORACLE_HOME
167
168 ctrl_file="$REP_MOUNTPOINT/$REP_NAME/control.ctl"

```

```

169 get_replica_control "$ORACLE_SID" "$ctrl_file"
170
171 dbs_file="$tmpdir/init.ora"
172 get_old_pfile "$dbs_file" "$ORACLE_SID"
173 edit_pfile "$dbs_file" "$ctrl_file" "$MOUNTPOINT" "$SDB_NAME" \
174     "$REP_NAME" "$REP_MOUNTPOINT"
175 [ -z "$INTERACTIVE" ] ||
176     suggest_editing "$dbs_file" "Do_you_wish_to_edit_the_standby_config?"
177
178 sql_file="$tmpdir/init.sql"
179 sql_pr_file="$tmpdir/pr_mod.sql"
180
181 create_listener_configs "$ORACLE_HOME" "$ORACLE_SID$REP_NAME" "$REP_NAME" \
182     "${REP_NAME}h";
183
184 [ -z "$INTERACTIVE" ] ||
185     suggest_editing "$ORACLE_HOME/network/admin/listener.ora" \
186     "Edit_the_listener_config?"
187 [ -z "$INTERACTIVE" ] ||
188     suggest_editing "$ORACLE_HOME/network/admin/tnsnames.ora" \
189     "Edit_the_tnsnames_config?"
190
191 generate_replica_sql "$dbs_file" > "$sql_file"
192 [ -z "$INTERACTIVE" ] ||
193     suggest_editing "$sql_file" "Edit_the_standby_creation_query?"
194
195 generate_primary_sql "$REP_NAME" > "$sql_pr_file"
196 [ -z "$INTERACTIVE" ] ||
197     suggest_editing "$sql_pr_file" \
198     "Edit_the_primary_database_modification_query?"
199
200 echo start "${REP_NAME}L" | lsnrctl
201
202 ORACLE_SID="$ORACLE_SID$REP_NAME" sqlplus / as sysdba @"$sql_file"
203 ORACLE_SID="$ORACLE_SID" sqlplus / as sysdba @"$sql_pr_file"
204
205 ln -s "$ORACLE_HOME/dbs/orapw$ORACLE_SID" \
206     "$ORACLE_HOME/dbs/orapw$ORACLE_SID$REP_NAME"

```

2.3.1. Параметры вызова

```

1 ./replica.sh -H "$HOME/app/oracle/product/11.2.0/dbhome_1" -s s182349 \
2     -M $HOME/rex71 -n bigmath -R /u01/rep71 -r replica

```

3. Выводы

В результате проделанной работы было обнаружено, что у Oracle SQL отсутствует сообщество, в связи с чем поиск объяснений даже самых простых ошибок в сети Интернет не даёт положительных результатов.

Выяснилось, что существуют проблемы, которые не затрагиваются документацией Oracle и анализируются только на дремучих форумах любителей баз данных. Например, стандартный путь к файлу паролей в документации найти не удалось, как и в полноценных сторонних описаниях. Ответ был найден в посте на форуме от 2007 года.

С другой стороны, графические установщики решают большую часть этих проблем. Вероятно, именно так, по замыслу разработчиков, и надо использовать базу данных.