

```

1  #!/bin/bash
2  # get options
3  while getopts "c:" opt
4  do
5      case $opt in
6          c) OPT_CONFIG="$OPTARG";;
7          \?)
8              echo "Invalid option: -$OPTARG" >&2
9              exit 1;;
10         :)
11             echo "Option -$OPTARG requires an argument." >&2
12             exit 1;;
13         esac
14     done
15     shift $((OPTIND - 1))
16
17     autorestart=0
18     command=$1
19
20     if [ "$command" = "autorestart" -o "$command" = "autorestart64" ]; then
21         autorestart=1
22
23         polltime=2
24         if [ -n "$2" ]; then
25             if [ "$2" -gt 0 ]; then
26                 polltime=$2
27             else
28                 echo "Poll time must be greater than 0 seconds."
29                 exit 1
30             fi
31         fi
32
33         restarttime=120
34         if [ -n "$3" ]; then
35             if [ "$3" -gt 0 ]; then
36                 restarttime=$3
37             else
38                 echo "Autorestart wait time must be greater than 0 seconds."
39                 exit 1
40             fi
41         fi
42
43         if [ "$command" = "autorestart64" ]; then
44             command=start64
45         else
46             command=start
47         fi
48     fi
49
50
51     if [ "$command" = "start64" ]; then
52         ANA_WEB_64=1
53         export ANA_WEB_64
54         command=start
55     fi
56
57     . common.sh
58
59     SawDataConfigDir=${SADATADIR}/web/config
60     SawJavaHostDir=${SAWROOTDIR}/javahost
61     CONFIG_XML=${OPT_CONFIG:=instanceconfig.xml}
62     BASE_CONFIG_XML=$(basename $CONFIG_XML)
63     CONFIG=${SawDataConfigDir}/${BASE_CONFIG_XML}
64

```

```

65 if [ "$ANA_WEB_64" = "1" ]; then
66     SAWBINDIR=${SAWROOTDIR}/bin64
67     SASAWSERVER=${SAWBINDIR}/sawserver64
68 else
69     SAWBINDIR=${SAWROOTDIR}/bin
70     SASAWSERVER=${SAWBINDIR}/sawserver
71 fi
72
73 SASAWSERVER="$SASAWSERVER -c $CONFIG "
74
75 #files
76 SASYSINITSCRIPT=${SAROOTDIR}/setup/sysenvinit.sh
77 SASRVTUNESCRIPRT=${SAROOTDIR}/setup/systunesrv.sh
78
79
80 if [[ ! -f $CONFIG ]]
81 then
82     echo "Config File $CONFIG doesn't exist"
83     echo "Exit now ..."
84     exit 1
85 fi
86
87 bin_ps="/bin/ps -o pid,args"
88 bin_sed="/bin/sed"
89 bin_grep="/bin/grep"
90 bin_awk="/usr/bin/awk"
91 bin_wc="/usr/bin/wc"
92 bin_id="/usr/bin/id"
93 bin_tty="/usr/bin/tty"
94
95 ATTACHED_TTY=$(($bin_tty|$bin_grep dev)
96 if test $ATTACHED_TTY; then
97     bold=$(tput bold)
98     offbold=$(tput sgr0)
99 else
100     bold=""
101     offbold=""
102 fi
103
104 ana_username=$(($bin_id -u -n)
105
106 get_pid ()
107 {
108     echo `${bin_ps} -u${ana_username} | ${bin_grep} $1 | ${bin_awk}
109     '{print $1}'`
110 }
111 get_num_proc()
112 {
113     echo `${bin_ps} -u${ana_username} | ${bin_grep} $1 | ${bin_wc} -l`
114 }
115
116 javahost_pid_pattern=
117 "[/]java.*com.siebel.analytics.javahost.standalone.Main"
118 javahost_pid=$(get_pid $javahost_pid_pattern)
119 sawserver_pid_pattern="[s]awserver.*$CONFIG"
120 sawserver_pid=$(get_pid $sawserver_pid_pattern)
121 instance_pattern="[s]awserver"
122 running_instances=$(get_num_proc $instance_pattern)
123
124 log_dir="${SADATADIR}/web/log"
125 javahost_logfile="${log_dir}/javahost.out.log"
126 logfile="${log_dir}/sawserver_${BASE_CONFIG_XML}.out.log"

```

```

127 {
128 if [ "$ANA_VARIANT" = "Linux" ];
129 then
130     echo "Usage: run-config-saw.sh { [ -c config_file ] start | stop |
forcestop | forcestopall | \
131 autorestart [<poll every n seconds> <restart if down for n seconds> ]}"
132 elif [ "$ANA_VARIANT" = "Solaris" ];
133 then
134     echo "Usage: run-config-saw.sh { [ -c config_file ] start | start64
| stop | forcestop | forcestopall | \
135 autorestart [<poll every n seconds> <restart if down for n seconds>] | \
136 autorestart64 [<poll every n seconds> <restart if down for n seconds>] }"
137 elif [ "$ANA_VARIANT" = "AIX" ];
138 then
139     echo "Usage: run-config-saw.sh { [ -c config_file ] start | start64
| stop | forcestop | forcestopall | \
140 autorestart [<poll every n seconds> <restart if down for n seconds>] | \
141 autorestart64 [<poll every n seconds> <restart if down for n seconds>] }"
142 elif [ "$ANA_VARIANT" = "HPUX" ];
143 then
144     if [ `uname -m` = "ia64" ];
145     then
146         echo "Usage: run-config-saw.sh { [ -c config_file ] start | start64
| stop | forcestop | forcestopall | \
147 autorestart [<poll every n seconds> <restart if down for n seconds>] | \
148 autorestart64 [<poll every n seconds> <restart if down for n seconds>] }"
149     else
150         echo "Usage: run-config-saw.sh { [ -c config_file ] start | stop |
forcestop | forcestopall | \
151 autorestart [<poll every n seconds> <restart if down for n seconds>] }"
152     fi
153 else
154     echo "Unsupported platform."
155 fi
156 }
157
158 start_javahost()
159 {
160     cd ${SawJavaHostDir}/bin && ./run.sh -service >> ${javahost_logfile}
161 } 2>&1 &
162 }
163
164 start_saw()
165 {
166     . $SASYSINITSCRIPT
167     . $SASRVTUNESCRIP
168     cd ${SAROOTDIR}/server/Log
169     ${SASAWSERVER} >> ${logfile} 2>&1 &
170 }
171
172 case "$command" in
173     'start')
174         if [[ `is_dev_random_readable` != "1" ]]; then
175             echo "Cannot start Oracle BI Presentation Services:"
176             echo "    /dev/random or /dev/urandom cannot be read"
177             exit 0
178         elif [[ "${ANA_WEB_64}" = "1" && `is_supported_64bit_kernel` !=
"1" ]]; then
179             echo "Oracle BI Presentation Services cannot be run in
64-bit mode on this machine"
180             exit 0
181             # elif test "$pid" ; then
182             # 1) Case: Sawserver is running and Javahost is not running

```

```

183     elif [[ -n "$sawserver_pid" && -z "$javahost_pid" ]]; then
184         echo "Oracle BI Presentation Services is already running."
185         echo "Oracle BI Java Host startup initiated."
186         echo "Please check '${javahost_logfile}' for Oracle BI Java
Host startup details."
187     echo "Please go to the '${log_dir}/javahost' directory for
Oracle BI Java Host log files."
188     start_javahost
189     # elif test "$javahost_pid" ; then
190     # 2) Case: Sawserver is not running and Javahost is running
191     elif [[ -n "$javahost_pid" && -z "$sawserver_pid" ]]; then
192         echo "Starting Oracle BI Presentation Services with config
file $BASE_CONFIG_XML"
193         start_saw
194         echo "Oracle BI Presentation Services startup initiated."
195     elif [[ -z "$javahost_pid" && -z "$sawserver_pid" ]]; then
196     # 3) Case: Sawserver is not running and Javahost is not running
197         echo "Oracle BI Java Host startup initiated."
198         echo "Please check '${javahost_logfile}' for Oracle BI Java
Host startup details."
199     echo "Please go to the '${log_dir}/javahost' directory for
Oracle BI Java Host log files after starting
200 up."
201     start_javahost
202     (
203         start_saw
204         echo "Oracle BI Presentation Services startup initiated."
205     )
206     echo "Please go to the '${log_dir}' directory for Oracle BI
Presentation Services log files."
207     elif [[ -n "$javahost_pid" && -n "$sawserver_pid" ]]; then
208         echo "Oracle BI Presentation Services with process id "$
sawserver_pid" and config `${BASE_CONFIG_XML}` is
already running."
209         echo "Oracle BI Java Host with process id "$javahost_pid"
is already running"
210         exit 0
211     fi
212     ;;
213
214     'stop')
215     # if test "$pid" ; then
216     if [[ -n "$sawserver_pid" ]]; then
217     (
218         echo "Stopping Oracle BI Presentation Services with
process id `${sawserver_pid}` and config
`${BASE_CONFIG_XML}`"
219         kill $sawserver_pid
220     )
221     else
222         echo "Oracle BI Presentation Services with config
`${BASE_CONFIG_XML}` is not running."
223     fi
224     if [[ $running_instances -eq 1 ]] ; then
225         echo "Stopping Oracle BI Java Host with process id
`${javahost_pid}` ."
226         echo "Please check '${javahost_logfile}' for details."
227         cd ${SawJavaHostDir}/bin && ./shutdown.sh -service >> ${
javahost_logfile} 2>&1 &
228     fi
229     ;;
230
231     'forcestop')
232     if [[ $running_instances -eq 1 && -n $javahost_pid ]] ; then

```

```

233     echo "Stopping Oracle BI Java Host with process id
${bold}${javahost_pid}${offbold} ."
234     echo "Please check '${javahost_logfile}' for details."
235     cd ${SawJavaHostDir}/bin && ./shutdown.sh -service >> ${
javahost_logfile} 2>&1 &
236     fi
237     # if test "$pid" ; then
238     if [ -n "$sawserver_pid" ]; then
239         echo "Stopping Oracle BI Presentation Services with process
id ${bold}${sawserver_pid}${offbold} and config
${bold}${BASE_CONFIG_XML}${offbold}."
240         kill -9 $sawserver_pid
241         echo "Oracle BI Presentation Services forcibly stopped."
242     fi
243     ;;
244
245     'forcestopall')
246     if [[ -n $javahost_pid ]] ; then
247         echo "Stopping Oracle BI Java Host with process id
${bold}${javahost_pid}${offbold} ."
248         echo "Please check '${javahost_logfile}' for details."
249         cd ${SawJavaHostDir}/bin && ./shutdown.sh -service >> ${
javahost_logfile} 2>&1 &
250     fi
251     # if test "$pid" ; then
252     for pid in $(get_pid [s]awserver)
253     do
254         echo "Stopping Oracle BI Presentation Services with process
id ${bold}${pid}${offbold} ."
255         kill -9 $pid
256         echo "Oracle BI Presentation Services forcibly stopped."
257     done
258     ;;
259 *)
260     usage
261     exit 1
262     ;;
263 esac
264
265 if [ "$autorestart" = 1 ]; then
266     while
267     true
268     do
269         sleep $polltime
270         sawserver_pid=$(get_pid $sawserver_pid_pattern)
271         # pid=`get_pid $processname`
272         if [[ -z "$sawserver_pid" ]] ; then
273             sleep $restarttime
274             sawserver_pid=$(get_pid $sawserver_pid_pattern)
275             # pid=`get_pid $processname`
276             if [[ -z "$sawserver_pid" && -z "$javahost_pid" ]] ; then
277                 start_javahost
278                 (
279                     echo "Restarting Oracle BI Presentation Services."
280                     start_saw
281                 )
282             elif [[ -z "$sawserver_pid" && -n "$javahost_pid" ]] ; then
283                 (
284                     echo "Restarting Oracle BI Presentation Services."
285                     start_saw
286                 )
287             fi
288         fi
289     done

```

```
290 fi
291
292 exit 0
293
294
```