# 脚本高级技巧

#### 上一行命令执行不成功时退出脚本

if [[ $? -ne 0 ]]; then echo "Download ${UniAgentOBSUrl} failed and exit(-1)"; exit -1; fi

#### 上一行命令执行成功时进行下一步

isProcessRunning ${UNIAGENT\_PROC\_NAME} > /dev/null  
status=$?  
if [[ ${status} == 0 ]]; then  
echo "Upgrading agent to the latest version, please wait..."  
bash /usr/local/uniagent/script/uninstall.sh > /dev/null  
if [[ $? -ne 0 ]]; then echo "${UNIAGENT\_PROC\_NAME} is running, reinstall ${UNIAGENT\_PROC\_NAME} failed and exit(-1)"; exit -1; fi  
fi

#### 根据上一条指令成功与否来决定下一步动作

if test $? = 0  
then  
 echo "Database Successfully Uploaded to the Ftp Server!"  
 echo -e "Database Successfully created and uploaded to the FTP Server!" | mail -s "Backup from $DATE" your\_email@email.com  
  
else  
 echo "Error in database Upload to Ftp Server" > $LOG\_FILE  
fi

### wait命令

wait [作业指示或进程号]

wait [作业指示或进程号]

1.等待作业号或者进程号制定的进程退出，返回最后一个作业或进程的退出状态状态。如果没有制定参数，则等待所有子进程的退出，其退出状态为0.

2.如果是shell中等待使用wait，则不会等待调用函数中子任务。在函数中使用wait，则只等待函数中启动的后台子任务。

3.在shell中使用wait命令，相当于高级语言里的多线程同步。

**示例**

<https://developer.aliyun.com/article/1095951>

### type 显示命令类型

<https://developer.aliyun.com/article/1095951>

### 判断参数个数

if [ $# != 1 ] ; then  
echo "USAGE: $0 TABNAME"  
echo " e.g.: $0 CDR\_CALL\_20040701"  
exit 1;  
fi

### main函数

main () {  
 clean #clean是个函数  
}  
  
main "$@"