

windows 批处理

批处理文件 (batch file) 包含一系列 DOS 命令，通常用于自动执行重复性任务。用户只需双击批处理文件便可执行任务，而无需重复输入相同指令。编写批处理文件非常简单，但难点在于确保一切按顺序执行。编写严谨的批处理文件可以极大程度地节省时间，在应对重复性工作时尤其有效。

在Windows中善用批处理可以简化很多重复工作

什么是批处理

批处理 (Batch)，也称为批处理脚本。顾名思义，批处理就是对某对象进行批量的处理。批处理文件的扩展名为 bat。

目前比较常见的批处理包含两类：DOS批处理和PS批处理。PS批处理是基于强大的图片编辑软件Photoshop的，用来批量处理图片的脚本；而DOS批处理则是基于DOS命令的，用来自动地批量地执行DOS命令以实现特定操作的脚本。这里要讲的就是DOS批处理。

批处理是一种简化的脚本语言，它应用于DOS和Windows系统中，它是由DOS或者Windows系统内嵌的命令解释器（通常是COMMAND.COM或者CMD.EXE）解释运行。类似于Unix中的Shell脚本。批处理文件具有.bat或者.cmd的扩展名，其最简单的例子，是逐行书写在命令行中会用到的各种命令。更复杂的情况，需要使用if, for, goto等命令控制程序的运行过程，如同C, Basic等中高级语言一样。如果更需要实现更复杂的应用，利用外部程序是必要的，这包括系统本身提供的外部命令和第三方提供的工具或者软件。

批处理文件，或称为批处理程序，是由一条条的DOS命令组成的普通文本文件，可以用记事本直接编辑或用DOS命令创建，也可以用DOS下的文本编辑器Edit.exe来编辑。在“命令提示”下键入批处理文件的名称，或者双击该批处理文件，系统就会调用Cmd.exe运行该批处理程序。一般情况下，每条命令占据一行；当然也可以将多条命令用特定符号（如：&、&&、|、||等）分隔后写入同一行中；还有的情况就是像if、for等较高级的命令则要占据几行甚至几十几百行的空间。

系统在解释运行批处理程序时，首先扫描整个批处理程序，然后从第一行代码开始向下逐句执行所有的命令，直至程序结尾或遇见exit命令或出错意外退出。

批处理命令简介

- echo
- rem
- pause
- call
- start
- goto
- set

http://xstarcd.github.io/wiki/windows/windows_cmd_summary_commands.html

批处理符号简介

- 回显屏蔽 @
- 重定向1 >与>>
- 重定向2 <
- 管道符号 |
- 转义符 ^
- 逻辑命令符包括：&、&&、||

http://xstarcd.github.io/wiki/windows/windows_cmd_summary_symbols.html

常用DOS命令

文件夹管理

- cd 显示当前目录名或改变当前目录。
- md 创建目录。
- rd 删除一个目录。
- dir 显示目录中的文件和子目录列表。
- tree 以图形显示驱动器或路径的文件夹结构。
- path 为可执行文件显示或设置一个搜索路径。
- xcopy 复制文件和目录树。

文件管理

- type 显示文本文件的内容。
- copy 将一份或多份文件复制到另一个位置。
- del 删除一个或数个文件。
- move 移动文件并重命名文件和目录。(Windows XP Home Edition中没有)
- ren 重命名文件。
- replace 替换文件。
- attrib 显示或更改文件属性。
- find 搜索字符串。
- fc 比较两个文件或两个文件集并显示它们之间的不同

网络命令

- ping 进行网络连接测试、名称解析
- ftp 文件传输
- net 网络命令集及用户管理
- telnet 远程登陆
- ipconfig显示、修改TCP/IP设置
- msg 给用户发送消息
- arp 显示、修改局域网的IP地址-物理地址映射列表

系统管理

- at 安排在特定日期和时间运行命令和程序
- shutdown立即或定时关机或重启
- taskkill 结束进程
- taskkill结束进程(比taskkill高级, 但WinXPHome版中无该命令)
- tasklist显示进程列表(Windows XP Home Edition中没有)
- sc 系统服务设置与控制
- reg 注册表控制台工具
- powercfg控制系统上的电源设置

对于以上列出的所有命令, 在cmd中输入命令+/?即可查看该命令的帮助信息。如find /?

Windows Batch 常用命令

1	<pre>! echo 和 @ 回显命令 @ #关闭单行回显 echo off #从下一行开始关闭回显 @echo off #从本行开始关闭回显 一般批处理第一行都是这个</pre>
---	-----------------------------------------------------------------------------------------------------------

```

1 @echo on          #从下一行开始打开回显
2 echo on           #显示当前是 echo off 状态还是 echo on 状态
3 echo              #输出一个“回车换行”，空白行
4 echo.             #(同echo, echo; echo+ echo[ echo] echo/ echo)
5
6
7
8 2 errorlevel
9 echo %errorlevel%
10 每个命令运行结束，可以用这个命令行格式查看返回码
11 默认值为0，一般命令执行出错会设 errorlevel 为1
12
13 3 dir
14 显示文件夹内容
15 dir              #显示当前目录中的文件和子目录
16 dir /a           #显示当前目录中的文件和子目录，包括隐藏文件和系统文件
17 dir c:/a:d       #显示 C 盘当前目录中的目录
18 dir c:/a:-d      #显示 C 盘根目录中的文件
19 dir c:/b/p       #/b只显示文件名，/p分页显示
20 dir *.exe /s     #显示当前目录和子目录里所有的.exe文件
21
22 4 cd
23 切换目录
24 cd              #进入根目录
25 cd              #显示当前目录
26 cd /d d:sdsk    #可以同时更改盘符和目录
27
28 5 md
29 创建目录
30 md d:abc         #如果 d:a 不存在，将会自动创建中级目录
31 #如果命令扩展名被停用，则需要键入 mkdir abc。
32
33 6 rd
34 删除目录
35 rd abc           #删除当前目录里的 abc 子目录，要求为空目录
36 rd /s/q d:temp   #删除 d:temp 文件夹及其子文件夹和文件，/q安静模式
37
38 7 del
39 删除文件
40 del d:test.txt   #删除指定文件，不能是隐藏、系统、只读文件
41 del /q/a/f d:temp*. *
42 删除 d:temp 文件夹里面的所有文件，包括隐藏、只读、系统文件，不包括子目录
43 del /q/a/f/s d:temp*. *
44 删除 d:temp 及子文件夹里面的所有文件，包括隐藏、只读、系统文件，不包括子目录
45
46 8 ren
47 重命名命令
48 ren d:temp tmp   #支持对文件夹的重命名
49
50 9 cls
51 清屏
52
53 10 type
54 显示文件内容
55 type c:boot.ini  #显示指定文件的内容，程序文件一般会显示乱码
56 type *.txt        #显示当前目录里所有.txt文件的内容
57
58 11 copy
59 拷贝文件
60 copy c:test.txt d:test.bak
61 复制 c:test.txt 文件到 d:，并重命名为 test.bak
62 copy con test.txt
63 从屏幕上等待输入，按 Ctrl+Z 结束输入，输入内容存为test.txt文件
64 con代表屏幕，prn代表打印机，nul代表空设备
65 copy 1.txt + 2.txt 3.txt
66 合并 1.txt 和 2.txt 的内容，保存为 3.txt 文件
67 如果不指定 3.txt，则保存到 1.txt
68 copy test.txt +
69 复制文件到自己，实际上是修改了文件日期
70
71 12 title
72 设置cmd窗口的标题
73 title 新标题     #可以看到cmd窗口的标题栏变了
74
75 13 ver
76 显示系统版本
77
78 14 label 和 vol
79 设置卷标
80
81

```

```

82 vol          #显示卷标
83 label        #显示卷标，同时提示输入新卷标
84 label c:system #设置C盘的卷标为 system
85
86 15 pause
87 暂停命令
88
89 16 rem 和 ::
90 注释命令
91 注释行不执行操作
92
93 17 date 和 time
94 日期和时间
95 date          #显示当前日期，并提示输入新日期，按"回车"略过输入
96 date/t        #只显示当前日期，不提示输入新日期
97 time          #显示当前时间，并提示输入新时间，按"回车"略过输入
98 time/t        #只显示当前时间，不提示输入新时间
99
100 18 goto 和 :
101 跳转命令
102 :label        #行首为:表示该行是标签行，标签行不执行操作
103 goto label    #跳转到指定的标签那一行
104
105 19 find (外部命令)
106 查找命令
107 find "abc" c:test.txt
108 在 c:test.txt 文件里查找含 abc 字符串的行
109 如果找不到，将设 errorlevel 返回码为1
110 find /i "abc" c:test.txt
111 查找含 abc 的行，忽略大小写
112 find /c "abc" c:test.txt
113 显示含 abc 的行的行数
114
115 20 more (外部命令)
116 逐屏显示
117 more c:test.txt #逐屏显示 c:test.txt 的文件内容
118
119 21 tree
120 显示目录结构
121 tree d:         #显示D盘的文件目录结构
122
123
124 22 &
125 顺序执行多条命令，而不管命令是否执行成功
126
127 23 &&
128 顺序执行多条命令，当碰到执行出错的命令后将不执行后面的命令
129 find "ok" c:test.txt && echo 成功
130 如果找到了"ok"字样，就显示"成功"，找不到就不显示
131
132 24 ||
133 顺序执行多条命令，当碰到执行正确的命令后将不执行后面的命令
134 find "ok" c:test.txt || echo 不成功
135 如果找不到"ok"字样，就显示"不成功"，找到了就不显示
136
137 25 |
138 管道命令
139 dir *.* /s/a | find /c ".exe"
140 管道命令表示先执行 dir 命令，对其输出的结果执行后面的 find 命令
141 该命令行结果：输出当前文件夹及所有子文件夹里的.exe文件的个数
142 type c:test.txt|more
143 这个和 more c:test.txt 的效果是一样的
144
145 26 > 和 >>
146 输出重定向命令
147 > 清除文件中原有的内容后再写入
148 >> 追加内容到文件末尾，而不会清除原有的内容
149 主要将本来显示在屏幕上的内容输出到指定文件中
150 指定文件如果不存在，则自动生成该文件
151 type c:test.txt >prn
152 屏幕上不显示文件内容，转向输出到打印机
153 echo hello world>con
154 在屏幕上显示hello world，实际上所有输出都是默认 >con 的
155 copy c:test.txt f: >nul
156 拷贝文件，并且不显示"文件复制成功"的提示信息，但如果f盘不存在，还是会显示出错信息
157 copy c:test.txt f: >nul 2>nul
158 不显示"文件复制成功"的提示信息，并且f盘不存在的话，也不显示错误提示信息
159 echo ^W ^> ^W>c:test.txt
160 生成的文件内容为 ^W > W
161

```

```

162 ^和>是控制命令，要把它们输出到文件，必须在前面加个^符号
163
164 27 <
165 从文件中获得输入信息，而不是从屏幕上
166 一般用于 date time label 等需要等待输入的命令
167 @echo off
168 echo 2005-05-01>temp.txt
169 date <temp.txt
170 del temp.txt
171 这样就可以不等待输入直接修改当前日期
172
173 28 %0 %1 %2 %3 %4 %5 %6 %7 %8 %9 %*
174 命令行传递给批处理的参数
175 %0 批处理文件本身
176 %1 第一个参数
177 %9 第九个参数
178 %* 从第一个参数开始的所有参数
179
180 批参数(%n)的替代已被增强。您可以使用以下语法:
181
182 %~1 - 删除引号(")，扩充 %1
183 %~f1 - 将 %1 扩充到一个完全合格的路径名
184 %~d1 - 仅将 %1 扩充到一个驱动器号
185 %~p1 - 仅将 %1 扩充到一个路径
186 %~n1 - 仅将 %1 扩充到一个文件名
187 %~x1 - 仅将 %1 扩充到一个文件扩展名
188 %~s1 - 扩充的路径指含有短名
189 %~a1 - 将 %1 扩充到文件属性
190 %~t1 - 将 %1 扩充到文件的日期/时间
191 %~z1 - 将 %1 扩充到文件的大小
192 %~$PATH:1 - 查找列在 PATH 环境变量的目录，并将 %1
193 扩充到找到的第一个完全合格的名称。如果环境
194 变量名未被定义，或者没有找到文件，此组合键会
195 扩充到空字符串
196
197 可以组合修定符来取得多重结果:
198
199 %~dp1 - 只将 %1 扩展到驱动器号和路径
200 %~nx1 - 只将 %1 扩展到文件名和扩展名
201 %~dp$PATH:1 - 在列在 PATH 环境变量中的目录里查找 %1，
202 并扩展到找到的第一个文件的驱动器号和路径。
203 %~ftza1 - 将 %1 扩展到类似 DIR 的输出行。
204 可以参照 call/? 或 for/? 看出每个参数的含意
205 echo load "%~%1" "%~%2">c:test.txt
206 生成的文件内容为 load "%1" "%2"
207 批处理文件里，用这个格式把命令行参数输出到文件
208
209 29 if
210 判断命令
211 if "%1"=="a" echo 第一个参数是/a
212 if /i "%1" equ "a" echo 第一个参数是/a
213 /i 表示不区分大小写，equ 和 == 是一样的，其它运算符参见 if/?
214 if exist c:test.bat echo 存在c:test.bat文件
215 if not exist c:windows (
216 echo 不存在c:windows文件夹
217 )
218 if exist c:test.bat (
219 echo 存在c:test.bat
220 ) else (
221 echo 不存在c:test.bat
222 )
223
224 30 setlocal 和 endlocal
225 设置"命令扩展名"和"延缓环境变量扩充"
226 SETLOCAL ENABLEEXTENSIONS #启用"命令扩展名"
227 SETLOCAL DISABLEEXTENSIONS #停用"命令扩展名"
228 SETLOCAL ENABLEDELAYEDEXPANSION #启用"延缓环境变量扩充"
229 SETLOCAL DISABLEDELAYEDEXPANSION #停用"延缓环境变量扩充"
230 ENDLOCAL #恢复到使用SETLOCAL语句以前的状态
231 "命令扩展名"默认为启用
232 "延缓环境变量扩充"默认为停用
233 批处理结束系统会自动恢复默认值
234 可以修改注册表以禁用"命令扩展名"，详见 cmd /?。所以用到"命令扩展名"的程
235 序，建议在开头和结尾加上 SETLOCAL ENABLEEXTENSIONS 和 ENDLOCAL 语句，以确
236 保程序能在其它系统上正确运行
237 "延缓环境变量扩充"主要用于 if 和 for 的符合语句，在 set 的说明里有其实用例程
238
239 31 set
240 设置变量
241

```

```

242  引用变量可在变量名前后加 %，即 %变量名%
243  set          #显示目前所有可用的变量，包括系统变量和自定义的变量
244  echo %SystemDrive%  #显示系统盘盘符。系统变量可以直接引用
245  set p          #显示所有以p开头的变量，要是有一个也没有就设errorlevel=1
246  set p=aa1bb1aa2bb2  #设置变量p，并赋值为 = 后面的字符串，即aa1bb1aa2bb2
247  echo %p%        #显示变量p代表的字符串，即aa1bb1aa2bb2
248  echo %p:~6%      #显示变量p中第6个字符以后的所有字符，即aa2bb2
249  echo %p:~6,3%     #显示第6个字符以后的3个字符，即aa2
250  echo %p:~0,3%     #显示前3个字符，即aa1
251  echo %p:~2%       #显示最后面的2个字符，即b2
252  echo %p:~0,-2%    #显示除了最后2个字符以外的其它字符，即aa1bb1aa2b
253  echo %p:aa=c%     #用c替换变量p中所有的aa，即显示c1bb1c2bb2
254  echo %p:aa=%       #将变量p中的所有aa字符串置换为空，即显示1bb12bb2
255  echo %p:*bb=c%     #第一个bb及其之前的所有字符被替换为c，即显示c1aa2bb2
256  set p=%p:*bb=c%    #设置变量p，赋值为 %p:*bb=c%，即c1aa2bb2
257  set /a p=39        #设置p为数值型变量，值为39
258  set /a p=39/10     #支持运算符，有小数时用去尾法，39/10=3.9，去尾得3，p=3
259  set /a p=p/10      #用 /a 参数时，在 = 后面的变量可以不加%直接引用
260  set /a p="1&0"     #"与"运算，要加引号。其它支持的运算符参见set/?
261  set p=             #取消p变量
262  set /p p=请输入
263  屏幕上显示"请输入"，并将输入的字符串赋值给变量p
264  注意这条可以用来取代 choice 命令
265  注意变量在 if 和 for 的复合语句里是一次性全部替换的，如
266  @echo off
267  set p=aaa
268  if %p%==aaa (
269      echo %p%
270      set p=bbb
271      echo %p%
272  )
273  结果将显示
274  aaa
275  aaa
276  因为在读取 if 语句时已经将所有 %p% 替换为aaa
277  这里的"替换"，在 /? 帮助里就是指"扩充"、"环境变量扩充"
278  可以启用"延缓环境变量扩充"，用 ! 来引用变量，即 !变量名!
279  @echo off
280  SETLOCAL ENABLEDELAYEDEXPANSION
281  set p=aaa
282  if %p%==aaa (
283      echo %p%
284      set p=bbb
285      echo !p!
286  )
287  ENDLOCAL
288  结果将显示
289  aaa
290  bbb
291  还有几个动态变量，运行 set 看不到
292  %CD%          #代表当前目录的字符串
293  %DATE%        #当前日期
294  %TIME%        #当前时间
295  %RANDOM%       #随机整数，介于0~32767
296  %ERRORLEVEL%  #当前 ERRORLEVEL 值
297  %CMDEXTVERSION% #当前命令处理器扩展名版本号
298  %CMDCMDLINE%  #调用命令处理器的原始命令行
299  可以用echo命令查看每个变量值，如 echo %time%
300  注意 %time% 精确到毫秒，在批处理需要延时处理时可以用到
301
302
303  32 start
304  批处理中调用外部程序的命令，否则等外部程序完成后才继续执行剩下的指令
305
306  33 call
307  批处理中调用另外一个批处理的命令，否则剩下的批处理指令将不会被执行
308  有时有的应用程序用start调用出错的，也可以call调用
309
310  34 choice (外部命令)
311  选择命令
312  让用户输入一个字符，从而选择运行不同的命令，返回码errorlevel为1234.....
313  win98里是choice.com
314  win2000pro里没有，可以从win98里拷过来
315  win2003里是choice.exe
316  choice /N /C y /T 5 /D y >nul
317  延时5秒
318
319  35 assoc 和 ftype
320  文件关联
321  assoc 设置'文件扩展名'关联，关联到'文件类型'

```

```

ftype 设置'文件类型'关联, 关联到'执行程序 and 参数'
当你双击一个.txt文件时, windows并不是根据.txt直接判断用 notepad.exe 打开
而是先判断.txt属于 txtfile '文件类型'
再调用 txtfile 关联的命令行 txtfile=%SystemRoot%\system32\notepad.exe %1
可以在"文件夹选项"→"文件类型"里修改这两种关联

assoc          #显示所有'文件扩展名'关联
assoc .txt     #显示.txt代表的'文件类型', 结果显示 .txt=txtfile
assoc .doc     #显示.doc代表的'文件类型', 结果显示 .doc=Word.Document.8
assoc .exe     #显示.exe代表的'文件类型', 结果显示 .exe=exefile
ftype          #显示所有'文件类型'关联
ftype exefile  #显示exefile类型关联的命令行, 结果显示 exefile="%1" %*
assoc .txt=Word.Document.8
设置.txt为word类型的文档, 可以看到.txt文件的图标都变了
assoc .txt=txtfile
恢复.txt的正确关联
ftype exefile="%1" %*
恢复 exefile 的正确关联
如果该关联已经被破坏, 可以运行 command.com , 再输入这条命令

36 pushd 和 popd
切换当前目录
@echo off
c: & cd & md mp3      #在 C: 建立 mp3 文件夹
md d:mp4              #在 D: 建立 mp4 文件夹
cd /d d:mp4           #更改当前目录为 d:mp4
pushd c:mp3           #保存当前目录, 并切换当前目录为 c:mp3
popd                  #恢复当前目录为刚才保存的 d:mp4

37 for
循环命令
这个比较复杂, 请对照 for/? 来看
for %%i in (c: d: e: f:) do echo %%i
依次调用小括号里的每个字符串, 执行 do 后面的命令
注意%%i, 在批处理中 for 语句调用参数用2个%
默认的字符串分隔符是"空格键", "Tab键", "回车键"
for %%i in (*.txt) do find "abc" %%i
对当前目录里所有的txt文件执行 find 命令
for /r . %%i in (*.txt) do find "abc" %%i
在当前目录和子目录里所有的.txt文件中搜索包含 abc 字符串的行
for /r . %%i in (.) do echo %%~pni
显示当前目录名和所有子目录名, 包括路径, 不包括盘符
for /r d:mp3 %%i in (*.mp3) do echo %%i>d:mp3.txt
把 d:mp3 及其子目录里的mp3文件的文件名都存到 d:mp3.txt 里去
for /l %%i in (2,1,8) do echo %%i
生成2345678的一串数字, 2是数字序列的开头, 8是结尾, 1表示每次加1
for /f %%i in ('set') do echo %%i
对 set 命令的输出结果循环调用, 每行一个
for /f "eol=P" %%i in ('set') do echo %%i
取 set 命令的输出结果, 忽略以 P 开头的那几行
for /f %%i in (d:mp3.txt) do echo %%i
显示 d:mp3.txt 里的每个文件名, 每行一个, 不支持带空格的名称
for /f "delims=" %%i in (d:mp3.txt) do echo %%i
显示 d:mp3.txt 里的每个文件名, 每行一个, 支持带空格的名称
for /f "skip=5 tokens=4" %%a in ('dir') do echo %%a
对 dir 命令的结果, 跳过前面5行, 余下的每行取第4列
每列之间的分隔符为默认的"空格"
可以注意到 dir 命令输出的前5行是没有文件名的
for /f "tokens=1,2,3 delims=-" %%a in ('date /t') do (
    echo %%a
    echo %%b
    echo %%c
)
对 date /t 的输出结果, 每行取1、2、3列
第一列对应指定的 %%a, 后面的 %%b 和 %%c 是派生出来的, 对应其它列
分隔符指定为 - 和"空格", 注意 delims=- 后面有个"空格"
其中 tokens=1,2,3 若用 tokens=1-3 替换, 效果是一样的
for /f "tokens=2* delims=-" %%a in ('date /t') do echo %%b
取第2列给 %%a, 其后的列都给 %%b

38 subst (外部命令)
映射磁盘。
subst z: serverd     #这样输入z:就可以访问serverd了
subst z: /d          #取消该映射
subst                #显示目前所有的映射

39 xcopy (外部命令)
文件拷贝
xcopy d:mp3 e:mp3 /s/e/y
复制 d:mp3 文件夹、所有子文件夹和文件到 e:, 覆盖已有文件
加 /i 表示如果 e: 没有 mp3 文件夹就自动新建一个。否则会有询问

```

