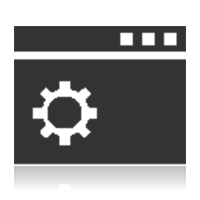
**Пакетные файлы или BAT файлы - примеры**

Если рассматривать исполняемые приложения, то среди них особую роль будут играть пакетные файлы с расширением BAT. Возможности ОС позволяют выполнить запись некоторой последовательности команд непосредственно в документ данного типа. Таким образом, получается файл, содержащий наборы команд. Это позволяет упростить функционирование ОС. Предусматривается наличие некоторого списка команд, которые могут быть запущены исключительно из пакетного документа.

Название BAT файлы было получено за счет наличия указанного расширения. Запуск пакетных документов является аналогичным приложениям .EXE и .COM. Следует отметить главное отличие от классических программ. BAT документы не содержат внутри код. Предусматривается наличие исключительно текстовой информация. Именно её считывает и выполняет командный процессор DOS. Существует несколько доступных вариантов:

* Особые команды для вывода определенных сведений или приложений на экран.
* Осуществление вызова различных пакетных документов.
* Работа с другими приложениями.
* Особые команды, обеспечивающие создание ветвлений, а также циклов.

Обеспечивается способность выполнения довольно сложных и многоуровневых последовательностей управления процессами. Наиболее значимым качеством пакетных документов требуется отметить автоматизацию работы над заданиями. Деятельность происходит без каких-либо вмешательств со стороны пользователей системы. Разрешается выполнение файлов BAT почти на всех этапах установки ОС Windows.

Допускается помещение внутрь пакетных документов всех существующих команд. Предполагается усложнение поставленных задач и даже создание условий. В этом деле оказывают поддержку команды goto, if и for. Требуется отметить autoexec.bat – файл необходим при осуществлении начальных настроек ОС в ходе запуска.

Следует рассмотреть простейший пример самостоятельного создания пакетного файла. Для выполнения поставленной задачи будет достаточно открыть новый документ с расширением TXT и ввести туда предложенный текст:

**title Batch File Testing**

**echo Hello World**

**echo.**

**echo Starting Notepad**

**start notepad**

**echo.**

**echo Starting Wordpad**

**start Wordpad**

**echo.**

**pause**

**exit**

После этого осуществляется сохранение документа и установка для него любого названия. Основным моментом требуется предусмотреть задание расширения CMD или BAT. После этого требуется выполнить запуск. Происходит назначение окна с названием "Batch File Testing". Работа указанного документа будет заключаться в том, чтобы открыть блокнот и выполнить запрос на нажатие любой клавиши. Когда это сделано, окно закроется.

Пакетные файлы применяются постоянно и сильно упрощают жизнь при осуществлении повседневных задач для программистов. Наиболее распространенными сферами использования BAT документов можно отметить резервное копирование данных, переименование, удаление файлов или документов, а также многое другое. Большинство действий в системе разрешается проводить только в том случае, если имеются администраторские права. Подобный фактор требуется принимать во внимание.

**Задание паузы в файле BAT перед исполнение последующей команды**

Разрешается осуществление паузы в результате применения большого числа команд. Часть из них предусматривают подобную функцию. Ниже будет рассмотрено выполнение поставленной задачи за счет команды **ping**:

**ping -n 1 -w 50000 123.123.123.254 >nul**

Следует перечислить и расшифровать параметры:

**n 1**

Запрос будет исполняться только один раз

**w 50000**

Для ожидания ответа выбран интервал в 50.000 мс (50 секунд)

**123.123.123.254**

В качестве этого параметра прописывается IP адрес, которого точно не существует внутри локальной сети. Это позволяет утверждать о невозможности ответа на запрос

**>nul**

Производится перенаправление для выдачи в несуществующее место

**Зацикливание в файле BAT выполнения команды**

Довольно распространенной задачей требуется отметить периодический вызов некоторой команды через равные промежутки времени. Подобные мероприятия особенно часто нужны в случае обновления некоторых программ или системы. Оптимальным решением поставленной задачи является создание зацикленного в файла BAT с некоторой паузой. Ниже рассмотрен пример реализации такого способа:

**:loop**

**ping -n 1 -w 60000 123.123.123.254 >nul**

**start notepad.exe**

**goto loop**

Происходит открытие блокнота каждую минуту.

**Создание папки, имя которой устанавливается текущей датой**

Папки данного вида популярны в качестве мест сохранения бэкапов. Необходимо рассмотреть основные методы создания.

**MD %SYSTEMDRIVE%\%date:~-10%**

Применение команды позволит создать новую папку и присвоить её название, соответствующее текущей дате. Недостатком указанного подхода считается невозможность получения результата в определённых ситуациях. Сюда можно отнести неподходящие установки отображения даты на компьютере или специфические региональные настройки. Чтобы подобные факторы не оказали своего влияния, применяется следующий вид:

**md %systemdrive%\%date:~0,2%\_%date:~3,2%\_%date:~6,4%**

или

**md %systemdrive%\%date:~6,4%\_%date:~3,2%\_%date:~0,2%**

Стоит предложить вариант по дальнейшему улучшению результата. Например, осуществление прибавления к названию документов текущей даты:

**@echo off**

**for /F "tokens=1-4 usebackq delims=. " %%1 in (`date /t`) do set mydate=%%4.%%3.%%2**

**ren test.txt test%mydate%.txt**

Это позволит изменить название документа с test.txt на test(гггг.мм.дд).txt

**Запрет повторного запуска программ**

Когда запуск определенной программы происходит с применением файла BAT, имеется возможность не допускать повторного исполнения такой процедуры.

**pslist.exe -e "операция"**

**if %errorlevel%==1 start "" "приложение"**

Здесь упоминается специальная программа pslist. Она является бесплатной и может быть свободно загружена из интернета.

**Выполнение резервных копий в автоматическом режиме**

Ниже будет представлен скрипт, который отвечает за создание архивных копий документов с данными пользователей, а также некоторыми другими документами. Не предусматривается проведение подобных мероприятий с медиа файлами. Причиной является их значительный объем. Дополнительно, их потеря редко считается критичной.

rem \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

rem задаём жесткий диск, где лежат папки для архивирования и винт куда будем бэкапить

rem \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**SET DRIVE1ST=d:**

**SET DRIVE2ND=e:**

rem \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

rem пароль на архив

rem \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**SET PASSWD=qmsdhgsdfgsdfhdsfhsdhf**

rem \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

rem если копировать для надёжности куда-нибудь на сетку

rem \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**rem SET COPYTO=\\shpakk\c$\!backup**

rem \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

rem \*\*\* Проверим путь назначения архива

rem \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**%DRIVE1ST%**

**if not exist \!backup md \!backup**

**if not exist \!backup\Departments md \!backup\Departments**

**if not exist \!backup\Exchange md \!backup\Exchange**

**if not exist \!backup\MEDEA\_rdisk md \!backup\MEDEA\_rdisk**

**cd \!backup**

rem \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

rem \*\*\* Backup \\SERVER\e$\Exchange

rem \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**SET DIRNAME=Exchange**

**SET FILENAME=SERVER\_Exchange**

**SET SOURCEDIR=e:\Exchange**

**rar a -r -m5 -dh -x\*.mp3 -x\*.fcd -x\*.avi -x\*.zip -x\*.rar -x\*.exe -w%DRIVE2ND%\ -p%PASSWD% %DIRNAME%\%FILENAME%\_new %SOURCEDIR%\\*.\***

**if not exist %DIRNAME%\%FILENAME%\_new.rar goto next2**

**del %DIRNAME%\%FILENAME%\_old.rar**

**ren %DIRNAME%\%FILENAME%.rar %FILENAME%\_old.rar**

**ren %DIRNAME%\%FILENAME%\_new.rar %FILENAME%.rar**

**copy %DIRNAME%\%FILENAME%.rar %COPYTO%\%DIRNAME%**

**:next2**

rem \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

rem \*\*\* Backup \\SERVER\e$\Users

rem \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**SET DIRNAME=Departments**

**SET FILENAME=SERVER\_Departments**

**SET SOURCEDIR=e:\Departments**

**if not exist %DIRNAME%\%FILENAME%\_new.rar goto next3**

**del %DIRNAME%\%FILENAME%\_old.rar**

**ren %DIRNAME%\%FILENAME%\_4.rar %FILENAME%\_old.rar**

**ren %DIRNAME%\%FILENAME%\_3.rar %FILENAME%\_4.rar**

**ren %DIRNAME%\%FILENAME%\_2.rar %FILENAME%\_3.rar**

**ren %DIRNAME%\%FILENAME%\_new.rar %FILENAME%\_2.rar**

**:next3**

rem \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

rem это если архив будет большим - кусками бить -v357888k

rem \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**rem rar a -r -m5 -v357888k -dh -x\*.mp3 -x\*.fcd -x\*.avi -x\*.exe -w%DRIVE2ND%\ -p%PASSWD% %DIRNAME%\%FILENAME%\_new %SOURCEDIR%**

rem \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

rem архивирование пошло, не берём мультимедиа файлы

rem \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**rar a -r -dh -x\*.mp3 -x\*.fcd -x\*.avi -x\*.exe -x\*.jpg -x\*.tif -p%PASSWD% %DIRNAME%\%FILENAME%\_new %SOURCEDIR%**

**goto next4**

**:next4**

rem \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

rem \*\*\* Create repair disk on \\SERVER

rem \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**SET DIRNAME=MEDEA\_rdisk**

**SET FILENAME=MEDEA\_rdisk**

**SET SOURCEDIR=%windir%\repair**

**%windir%\system32\rdisk.exe /s-**

**rar a -r -m5 -dh -w%DRIVE2ND%\ -p%PASSWD% %DIRNAME%\%FILENAME%\_new %SOURCEDIR%\\*.\***

**if not exist %DIRNAME%\%FILENAME%\_new.rar goto next5**

**del %DIRNAME%\%FILENAME%\_old.rar**

**ren %DIRNAME%\%FILENAME%.rar %FILENAME%\_old.rar**

**ren %DIRNAME%\%FILENAME%\_new.rar %FILENAME%.rar**

**copy %DIRNAME%\%FILENAME%.rar %COPYTO%\%DIRNAME%**

**:next5**

В итоге имеем 3 папки: Departments, в ней:

SERVER\_Departments\_2.rar - 2х дневной давности

SERVER\_Departments\_3.rar - 3х дневной давности

SERVER\_Departments\_4.rar - 4х дневной давности

SERVER\_Departments\_new.rar - НОВЫЙ

SERVER\_Departments\_old.rar - СТАРЫЙ Exchange

Так как архивируем обменник - он нам сильно то и не нужен храним только сегодняшний и вчерашний архивы

SERVER\_Exchange.rar

SERVER\_Exchange\_old.rar ну и систему на всякий в папке

SERVER\_rdisk

SERVER\_rdisk.rar

SERVER\_rdisk\_old.rar

**Использование комментариев в файлах BAT**

Чтобы разбираться в том, что именно осуществляет заданный BAT документ, рекомендуется применять комментарии. Использование их для главных блоков особенно важно в случае значительного объема текстовой информации. Просматривая файл через некоторое время, у Вас не возникнет проблем с пониманием особенностей его работы.

Написание комментариев можно осуществить несколькими способами. Когда приходится иметь дело с довольно большим количеством текста, рекомендуется следующий подход:

**goto start**

--------------------------------------

Этот пакетный файл предназначен

для автоматизации рутинных операций,

выполняемых ночью для синхронизации

содержимого корпоративного ftp-сервера

с ftp-серверами филиалов

--------------------------------------

Пакетный файл написан 01/01/2016

Последнее исправление внесено 10/02/2016

--------------------------------------

И т.д.

**:start**

Подобный вариант обеспечивает запуск пакетного документа, после чего выполняется переход контроля непосредственно команде, которая помещается после start. В некоторых ситуациях будет выгодным использовать иной подход. Например, хорошо подходит применение команды rem или два двоеточия. Они оптимально подходят для небольших по своему размеру комментариев с длиной в одну строчку.

**rem** Комментарий

**::** Комментарий

Если Вы хотите обеспечить для себя удобство последующего разбора своих записей, рекомендуется не забывать про возможности комментирования. Не стоит описывать каждый шаг, но основные блоки должны иметь информацию о себе. Комментарии считаются правилом хорошего тона в программировании.

**Ожидание появления заданного документа в файле BAT**

В некоторых ситуациях возникает потребность осуществить заданные действия при возникновении определенного файла. Осуществление такой задачи в пакетном документе предусматривается следующий подход:

**:test**

**if exist c:\1.txt goto go**

**ping -n 20 127.0.0.1 >nul**

**goto test**

**:go**

**start notepad.exe**

Приведенный текст задает необходимость выполнения проверки каждые 20 секунд того, существует ли в корне диска С документа с названием 1.txt. Если подобное событие произойдет, выполняются действия после **goto**. В конкретной ситуации производится запуск программы блокнота, но можно использовать любые другие варианты.

Периодически возникает ситуация, когда искомый файл обладает крупным объемом и проводится его копирование. Таким образом, поиск выдаст результат, но какие-либо действия повлекут за собой появление ошибки. Чтобы этого не возникло, нужно несколько улучшить содержимое пакетного файла.

**:test**

**if exist c:\1.txt goto go**

**ping -n 10 127.0.0.1 >nul**

**goto test**

**:go**

**rename c:\1.txt 1.txt**

**if not %errorlevel% == 0 goto go**

**del c:\1.txt**

Как и раньше, попытка каких-либо действий с не полностью скопированным документом вызовет ошибку, но она не приведет к завершению. Произойдет повторение цикла и так до тех пор, пока ошибка не исчезнет.

**Способы обхода процедуры проверки даты из файла BAT**

Некоторые программы в процессе своей работы осуществляют проверку текущей даты. В большинстве случаев это создано для контроля временных рамок использования. Достаточно часто владелец устанавливает возможность бесплатной эксплуатации на протяжении установленного периода времени. Когда сроки истекают, расширенный или даже весь функционал программы перестает работать. Другой вариант – это регулярное обновление до последней версии.

Если Вы не хотите, чтобы программа использовала временную проверку, можно изменить свои настройки даты и времени. Этот вариант нельзя назвать приемлемым по многим причинам. Более эффективным подходом требуется назвать использование пакетного файла со специально прописанной информацией. Предполагается запуск приложения и последующий возврат даты к тому значению, что присутствовало ранее.

**set tempdate=%date:~-10%**

**date 01-01-04**

**start notepad.exe**

**date %tempdate%**

Что именно делают прописанные строки? Прежде всего, производится сохранение текущей даты в системе. Для этого применяется особая переменная. Во второй строке происходит установка системной даты на новое значение. В данном случае, это 1 января 2004 года. Сразу после этого предполагается проведение вызова программы, отвечающей за проверку системной даты. Мы использовали блокнот, но в Вашем случае требуется предусмотреть подобный момент. Пока Блокнот является открытым, пакетный документ ожидает, а сама дата не предусматривает изменение обратно. После закрытия Блокнота происходит возврат системной даты из переменной.

Важно уделить внимание определенному моменту. Когда в пути до конкретного документа используются символы из кириллицы, рекомендуется применять текстовый редактор с поддержкой кодировки 866 (ОЕМ, русская) или аналогичного варианта. Когда это не предусматривается, происходит выдача ошибки и работа пакетного файла не осуществляется.

Большинство программ упрощают систему проверки – она выполняется исключительно в процессе запуска. Это позволяет модифицировать пакетный документ в соответствии с подобными запросами.

**set tempdate=%date:~-10%**

**date 01-01-04**

**start notepad.exe**

**date %tempdate%**

Процесс работы довольно прост. В первую очередь выполняется изменение системной даты и последующий запуск программы. Пакетный документ не станет ждать, пока она завершит своё функционирование и осуществит возврат даты на прежнее значение.

**Импорт переменных реестра из пакетного файла**

Когда возникает потребность выполнения импорта в реестр определенных переменных, осуществление поставленной задачи предполагает использование следующей команды:

**regedit.exe -s C:\environment.reg**

Производится импорт в реестр сведений из документа environment.reg. Поскольку применяется специальный ключ –s, то не предусматривается вывода сообщений для подтверждения таких действий.

**Экспорт переменных реестра из пакетного файла**

В некоторых случаях поставлена задача, противоположная рассмотренной в предыдущем абзаце. Чтобы сохранить в пакетном файле некоторую часть реестра с полным перечнем параметров, необходимо использовать следующий подход:

**regedit.exe -ea C:\environment.reg "HKEY\_CURRENT\_USER\Environment"**

Происходит сохранение ветки HKEY\_CURRENT\_USER\Environment в документ, расположенный по адресу C:\environment.reg. При запросе на восстановление значений HKEY\_CURRENT\_USER\Environment, предполагается необходимым только выполнить запуск environment.reg. Примером использования такой команды следует назвать осуществление регулярного бэкапа настроек ПО или всей системы. Подобная информация находится в реестре и может быть выгружена оттуда. Полный адрес может включать в себя проблемы. Чтобы они не вызвали ошибку, надо позаботиться о заключении подобного пути в кавычки.

**Применение внутри пакетного файла символа %**

Символ процента относится к категории спецсимволов. Если требуется применить его внутри пакетного документа, то следует прописывать дважды.

**echo 50%%**

Если ввести указанную команду, то предполагается выдача на экран в BAT документе 50%. Без использования двойного символа %%, результатом станет только число 50.

**Смена имен по маске из файла BAT**

В некоторых ситуациях поставлена задача сменить названия всех файлов в папке с применением определенного шаблона из пакетного документа. Предполагается использование следующей команды:

**for /f "tokens=\*" %%a in ('dir /b PATH\\*.\*') do ren PATH\%%a Prefix%%a**

Требуется выполнить некоторые замены в зависимости от конкретной ситуации. Вместо PATH\ устанавливается расположение документов, название которых должно быть изменено. Prefix требуется заменить на символы, добавляемые к файлу в процессе смены имени. Важным условием является расположение самого файла BAT. Он не должен находиться в папке, где происходит переименование, иначе название документа будет изменено. Другой особенностью является работа с подпапками. Они не смогут избежать общей процедуры смены имени и к названию также добавится префикс.

Имеется возможность осуществить процедуру исключительно по отношению к некоторым типам файлов. Для этого в прописанной ранее строке надо изменить \*.\* на другой вариант – после звездочки указывается интересующее расширение (\*.txt или иное). Это позволит переименовать только документы определенного вида. Пример с несколько другим подходом написан ниже:

**set thePATH=C:\test**

**for %%I in (\*.txt) do ren "%thePATH%\%%~nxI" "%%~nI.dat"**

**Определение имени компьютера из пакетного файла**

Непосредственно из файла BAT можно узнать название компьютера. Для этого необходимо прописать в документе

**hostname**

Команда предусматривает возврат имени устройства, на котором произошел её запуск.

**Изменение IP адреса устройства из пакетного файла**

Чтобы изменить IP адрес компьютера, следует применять команду netsh. Начать подобные мероприятия можно только после того, как станет определены текущие настройки. Требуется получить IP адрес на данный момент:

**netsh interface ip show address**

Происходит выдача текущей конфигурации. Нам необходимо узнать название применяемого интерфейса. В качестве примера это может быть FASTNET. Далее будет предполагаться необходимость изменения IP адреса на 172.22.0.33 (шлюз 172.22.0.1, маска 255.255.255.0). Предусматривается следующая строка в пакетном файле для выполнения поставленной задачи:

**netsh interface ip set address name="FASTNET" static 172.22.0.33 255.255.255.0 172.22.0.1 1**

Обработка команды повлечет изменение IP адреса у интерфейса FASTNET на 172.22.0.33. Это и является искомой задачей. Указанная команда предлагает весьма широкие возможности в плане работы с сетевыми настройками.

**Автоматизация удаления документов по типу**

Задача очистки жесткого диска от временных файлов может быть решена с помощью следующей строки:

**del /f /s /q C:\\*.tmp**

Следует разъяснить используемые параметры

**/f**

Происходит удаление всех документов, даже при наличии у них атрибута «только чтение»

**/s**

Выполняется удаление файлов из всех подкаталогов.

**/q**

Не всплывает окна, просящего подтверждения действиям удаления.

**C:**

Прописывается диск, для которого будет производиться удаление документов. В данном случае, разрешается указывать более конкретный путь, например, до некоторой папки.

**\*.tmp**

Тип удаляемых документов.

Следует аккуратно использовать параметр **/q**. С ним удаление будет происходить без каких-либо запросов и это может привести к весьма неприятным последствиям.

**Отправка сообщения непосредственно из файла BAT**

Следует рассмотреть ситуации, когда осуществляется работа пакетного файла на одном из устройств в сети. В подобной ситуации будет особенно удобным, когда после завершения установленных мероприятий произойдет пересылка соответствующего сообщения для администратора. Это довольно просто сделать с применением следующей строки:

**net send name Message text**

Здесь параметр **name** означает название некоторого устройства, куда будет отправлено сообщение. В качестве самого сообщения следует прописать информацию **Message text**. Важно учитывать применяемый алфавит. Когда осуществляется работа с кириллицей, нужно применять кодировку MS-DOS (866 кодовая страница) или любую аналогичную ей. В ином случае, следует приготовиться к выдаче набора ничего не значащих символов. Таким образом, рекомендуется применять соответствующие типы текстовых редакторов.

В качестве примера подобной программы можно отметить FAR. Следует выполнить открытие пакетного документа через этот текстовый редактор за счет нажатия клавиши F4 или выбора соответствующего пункта. Указывается кодировка DOS – она исключает проблему невозможности распознания символов системой.

**Запуск BAT файла из другого BAT документа**

Чаще будет проще использовать возможность запуска из одного BAT файла другого, чем прописывание дополнительных строк действий. Для этого могут предусматриваться и более серьёзные рекомендации, чем удобство пользователя. Например, при необходимости приостановить осуществление основного пакетного документа во время выполнения другого или задать их параллельную деятельность.

Следует рассмотреть подобный момент более подробно. Выполним создание пакетного файла 1.bat, который будет включать следующее содержимое:

**call 2.bat**

То есть, происходит вызов второго пакетного файла 2.bat, где пропишем одну команду:

**pause**

Посмотрим, что произойдет при запуске 1.bat. Если это сделать, то на экран выдастся сообщение с предложением нажатия любой клавиши. Когда это будет сделано, произойдет закрытие окна. Из этого можно сделать довольно простой вывод. Когда один пакетный файл вызывается из другого с применением call, предусматривается остановка его исполнения до той поры, пока выполнение второго не будет завершено.

Стоит рассказать об иной ситуации, которая предусматривает необходимость параллельного выполнения пакетных документов. Подобный запрос возникает довольно часто и для его реализации можно использовать команду start, как оптимальный вариант в сложившихся условиях. Вернемся к созданным документам и в 1.bat введем некоторые корректировки:

start 2.bat

Теперь предполагается открытие окна с предложением нажать любую кнопку при запуске 1.bat. При этом, окно главного пакетного документа закроется после выполнения своей задачи. Теперь Вы понимаете особенности применения команд start и call. Следует отметить тот факт, что они используются не только для того, чтобы запускать пакетные файлы. Команды вполне способны осуществить начало работы любого приложения или документа.

**Отключение выдачи на экран команд в процессе осуществления пакетного файла**

Когда в пакетном файле выполняются различные команды, предусматривается выдача результата работы на экран. Далеко не всегда имеется запрос в подобных действиях и требуется осуществить исполнение с отключенным выводом. Для этого следует применять символ @, который прописывается перед самой командой.

Требуется рассмотреть это на примере. Допустим, в пакетном файле имеется строка

**echo Testing**

На экран предусматривается выдача echo Testing, а потом результат деятельности команды – слово Testing. Можно написать внутри пакетного документа следующие сведения:

**@echo Testing**

Результатом работы станет только выдача слова Testing. Можно пойти ещё дальше и установить работу таким образом, чтобы отключился вывод команд на протяжении всего файла BAT. В таком случае, требуется задать

**@echo off**

**Автоматический ответ на запросы о подтверждении**

При работе некоторых команд может возникнуть ситуация, когда они должны совершить потенциально опасное действие в системе. В качестве примера рекомендуется назвать format или del. Подобный момент предусмотрен и выдается соответствующее предупреждение. Когда производится выполнение внутри пакетного файла, то его деятельность остановится. Она будет возобновлена только после подтверждения и это способно стать причиной потери драгоценного времени.

Существует способ осуществления задачи автоматизации. Предполагается использование перенаправления необходимого ответа в команду. Далее будет на примере рассмотрено подобное действие. Требуется создать на диске С обычную папку Folder и внутрь неё осуществим помещение любых 2х документов.

Запускаем интерпретатор команд и вводим в нем

**del C:\Folder**

Это предполагает необходимость осуществить удаление вместе со всеми файлами внутри. Поскольку подобное действие относится к числу потенциально опасных, поступит запрос на подтверждение следующего вида:

*C:\Folder\\*, продолжить [Y(да)/N(нет)]?*

Команда не станет продолжаться до тех пор, пока пользователь не ответит. Если работа производится в автономном режиме, ожидание может быть крайне длительным, а сам процесс останется невыполненным. Для выхода из сложившейся ситуации рекомендуется применение специального символа (|). Он предусматривает, что должна произойти передача следующей сразу после команде. Теперь модифицируем строку в пакетном файле:

**echo Y|del C:\Folder**

На экран происходит выдача подтверждения с уже положительным решением. Таким образом, из указанной папки произойдет удаление всех документов. Следует аккуратно использовать подобную возможность.

**Перенаправление результата выполнения команды в документ**

В случае деятельности BAT файлов сложного типа способны возникнуть проблемы в плане того, насколько правильно функционирует система. Именно по этой причине применяется специальное средство – запись в специальный лог-файл. Это предусматривает возможность ознакомиться с документом в дальнейшем и провести анализ того, насколько правильно осуществлялось выполнение поставленных задач.

Чтобы выполнить перенаправление результата работы в специально предназначенный для этого документ, требуется создать файл BAT со следующими строками:

@echo off

echo Start %time%

echo Create test.txt

echo test>C:\test.txt

echo Copy Test.txt to Old\_test.txt

copy C:\test.txt C:\Old\_test.txt

echo Stop %time%

Необходимо подробнее разобрать, что именно выполняет указанный набор команд.

Первой строкой задается отключение вывода команд. Это позволит получить в лог-документе исключительно результаты осуществления.

Второй строкой указывается время, когда начал исполняться пакетный файл.

Третьей строкой указывается на то, что следующей командой будет создаваться текстовый документ test.txt.

Четвертая строка предназначена для того, чтобы непосредственно осуществить поставленную задачу. Следующим этапом производится запись в него одного слова – test.

Пятая строка проводит выдачу в лог-документ пояснение о том, что следующим этапом станет происходить копирование в иное место.

В шестой строке осуществляется непосредственное копирование C:\test.txt в файл C:\Old\_test.txt. Можно говорить о том, что происходит создание копии документа, но с присвоением для него нового названия.

В заключительной строке указывается команда, которая предусматривается для указания времени действия пакетного документа. Аналогичные сведения можно получить в лог-файле.

Требуется рассмотреть дальнейшие возможности по модификации предложенного набора команд. Часто возникает запрос на выполнение хранения подобных документов в специально предназначенной для них папке. Каждый день должна производиться запись отчета с присвоением для него нового названия. Оптимальный вариант – это использование имени, состоящего из некоторого слова и автоматически добавляемой к нему текущей даты. Сам файл можно назвать 1.bat и сохранить.

Теперь создадим на диске папку LOG и полный путь к ней станет иметь указанный вид: C:\LOG. Следующим этапом происходит запуск команду установленного типа:

**1.bat>C:\LOG\%date~-10%.txt**

Когда в расположении документа предусматриваются пробелы, рекомендуется предусмотреть указанный момент и применять кавычки или специальный формат записи. Ниже указано два возможных варианта:

**"C:\Program Files\1.bat">C:\LOG\%date~-10%.txt**

**C:\Progra~1\1.bat>C:\LOG\%date~-10%.txt**

Предполагается, что сразу после начала работы 1.bat в папке C:\LOG произойдет создание документа с названием, которое включает текущую дату. Именно указанный файл и требуется установить в качестве отчета о деятельности пакетного файла 1.bat. Если запустить рассмотренный BAT файл, то предполагается предоставление указанного далее результата:

Start 19:03:27,20

Create test.txt

Copy Test.txt to Old\_test.txt

Скопировано файлов: 1.

Stop 19:03:27,21

Чтобы произвести перенаправление результатов работы bat-документа в лог-документа, следует применять специальный символ (>). Написание команды будет иметь представленный ниже вид:

**Путь\ИмяФайла.bat>Путь\ИмяЛогФайла.txt**

Некоторые вопросы касаются расширения, которое должно даваться лог-файлу. Оно может быть любым, в том числе являться страницей html. Последний вариант особенно удобен в том случае, когда выгрузка значений должна происходить на какой-либо сайт.

**Создание папки из BAT файла**

Чтобы выполнить поставленную задачу, предполагается несколько доступных вариантов действий. Наиболее простой способ включает в себя применение команды MKDIR или специального сокращенного аналога – MD. Предполагается написание указанной ниже строки в документе:

**MD FolderName**

Когда такая команда будет осуществлена, предполагается образование новой папки FolderName. Она помещается туда же, где располагается файл BAT. Вовсе не обязательно обеспечивать подобную привязку. Допускается создавать папку в любом указанном пользователем месте, что и внесено в следующую команду:

**MD D:\FolderName**

Разрешается применять системы переменных в тех случаях, когда выполняется создание папок. Пример заключается в следующем – требуется создать папку в корне жесткого диска D с названием в виде имени текущего юзера. Задается вот так:

**MD D:\%USERNAME%**

Дальнейшее усложнение команды предписывает создание папки с именем юзера на системном диске устройства:

**MD %SYSTEMDRIVE%\%USERNAME%**

Ниже рассматривается вариант, когда в качестве названия для папки применяется дата на момент выполнения пакетного файла.

**set datetemp=%date:~-10%**

**MD %SYSTEMDRIVE%\%datetemp%**

Нужно выполнить рассмотрение того, каким именно образом происходит функционирование набора команд. Первой строкой создается специальная переменная datetemp. Когда работы завершатся, она удаляется, но до этого момента играет важную роль. Предусматривается корректировка значения переменной на протяжении всего времени её существования. В строке осуществляется присвоение значения, представляющего собой десять символов с правой части от системной переменной DATE. Используется формат представленного типа: Дн ДД.ММ.ГГГГ. Сначала слева идет указание недель и оно не представляет интереса. При создании папок открывается большое количество путей, которыми разрешается следовать. Это открывает большие возможности перед пользователями. Если Вы хотите увидеть перечень всех переменных, следует применить команду SET.

**Создание документа с произвольным названием из BAT файла**

Чтобы можно было создать документ в ходе осуществления пакетного файла, нужно написать специальный символ, отвечающий за перенаправление: (>). Предполагается перенаправление потока с экрана на документ. Рекомендуется использование указанного способа:

**@echo Start file>C:\1.txt**

Исполнение команды создает текстовый документ, обладающий строкой **Start file**. В ходе создания документа станет возможным применение в качестве части или всего его названия значений некоторых переменных. Схожие моменты уже рассматривались ранее. В качестве примера требуется отметить создание файла с названием, соответствующим дате его запуска.

**set datetemp=%date:~-10%**

**@echo .>%SYSTEMDRIVE%\%DATETEMP%.txt**

Работа начинается с того, что создается некоторая переменная, которая получает название datetemp. Для неё производится задание десятка символов с правой части системной переменной DATE. Это позволяет обеспечить применение исключительно действующих дат. Во второй строке указывается перенаправление (.) в документ. Его название получается за счет использования значения переменной datetemp. Должно быть указано TXT расширение, поскольку требуется получить текстовый документ.

Когда происходит работа в сети и пользователь с правами администратора реализует сбор сведений, рекомендуется добавлять название главного устройства. Это позволит упростить задачу дальнейших проверок. Требуется использование команды:

**@echo .>C:\FolderName\%COMPUTERNAME%.txt**

Происходит создание на диске С документа TXT, обладающего названием, состоящим из имени устройства выполнения пакетного файла. Допускается модернизировать предложенные строчки путем внесения различных переменных.

**Процесс автоматизации запуска файлов BAT**

Ряд процессов требует осуществления задачи запуска документов BAT без привлечения к подобному процессу внимания со стороны пользователя. Автоматическая деятельность системы вполне может быть прописана. Такой подход используется довольно часто. Классический планировщик из инструментов операционной системы вполне способен помочь в выполнении начала работы пакетных файлов по расписанию. Таким образом, происходит запуск в точно установленное юзером время или через некоторые интервалы.

Помимо стандартного планировщика можно найти огромное количество специализированных программ. Значительная часть из них являются свободно доступными. В платных версиях предусматривается использование расширенных возможностей.

Ещё одним вариантом можно назвать автоматический запуск некоторых действий каждый раз при входе пользователем в учетную запись или включении устройства. Ранее уже рассматривались основные направления деятельности и способы задания команд для подобных ситуаций.

**Применяемые в документах BAT команды**

Как уже упоминалось, допускается использовать все команды, некоторые из них предназначаются специально для работы в пакетных файлах. Можно ознакомиться с полным перечнем. Для этого предполагается получить доступ к командной строке. Это можно сделать через Пуск > Выполнить > CMD.

Предусматривается несколько вариантов получения сведений:

**help**

Если осуществить ввод указанной команды, можно получить полный список существующих команд и небольшие описания к ним. Разрешается обеспечение более детальных сведений. Чтобы можно было их получить, требуется ввести **help** и название интересующей команды. В качестве примера рекомендуется указать возможность получения данных по АТ:

**help** **AT**

Результатом необходимо назвать выдачу перечня ключей для осуществления запуска указанной команды из пакетного документа. В данном плане Windows предоставляет гораздо более широкие возможности, чем DOS. Одним из примеров расширенного функционала требуется назвать открытие файла логов деятельности пакетного доку

**start название\_документа.txt**

Если осуществить запуск подобной команды, то произойдет открытие **название\_документа.txt**. Важным моментом является тот факт, что пакетный файл окончит свою деятельность. Использование подобного метода является оправданным в тех случаях, когда объемы лог-файла невелики.

**Создание BATCH файла**

Следует открыть программу «Блокнот» и в ней провести набор указанных далее строк:

**@echo This batch file**

**@pause**

После этого требуется выполнить сохранение в качестве файла 1.bat. Следующим этапом таких мероприятий, следует провести открытие пакетного документа. Это повлечет за собой выдачу строк:

**This batch file**

**Нажмите любую клавишу для продолжения...**

Когда предложение будет исполнено, произойдет закрытие пакетного документа, поскольку его работа завершается. Отдельное внимание должно уделяться этому символу - **@**. Он задает выдачу исключительно результата работы, а не непосредственно команды. Вы можете осуществить несколько экспериментов, связанных с изменением положения или удаления данного символа. Это позволит лучше понять особенности его использования.