

Regulaaravaldised ja skriptimine Windows ja UNIX operatsioonisüsteemides

WINDOWS

Windows käsurida

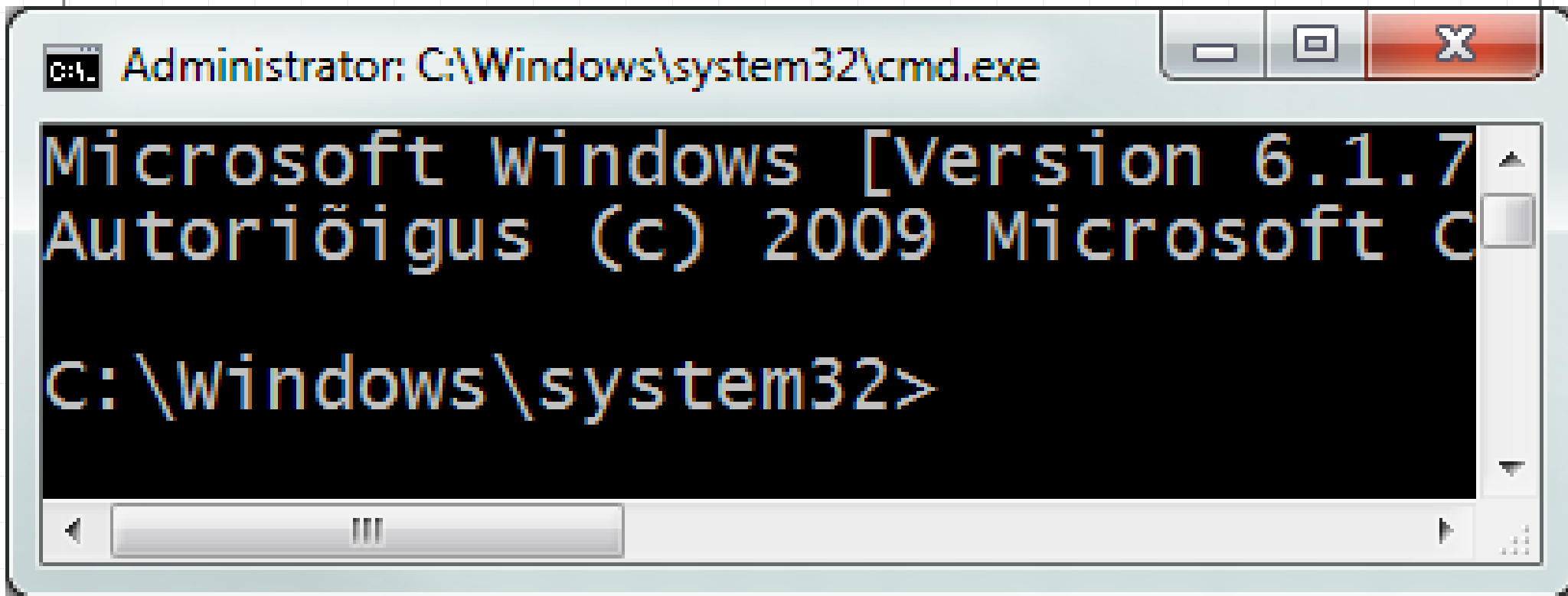
- Käsureaks nimetan programme:
 - `cmd.exe`
 - `powershell.exe`
 - `command.com` (tänapäevastes OS ei kasutata)
- Nendesse saab kirjutada käske, millega näiteks operatsioonisüsteemi tegevusi juhtida.
- Kiireks käivitamiseks – Windows + R ja tekkinud lahtrisse kirjuta `cmd` (`powershell`).



Windows OS administraator

- Osa asjadest saab teha ainult administraatori (Administrator) ehk süsteemiülema õigustes.
- Käsurea avamiseks süsteemiülema õigustes tuleb:
 - Otsida programm cmd.exe ja parema hiirenupuga klõpsates avanenud kontekstimenüüst valida **Käivita administraatorina**
 - Vajutad klaviatuuril lipumärki (Win); kirjutad otsinguaknasse cmd (powershell); kui programm leitud vajutad **Ctrl+Shift+Enter**
- Loomulikult peab teadma selleks õiget parooli.

Kelle õigustes käsurida on avatud?



The image shows a screenshot of a Windows command prompt window. The title bar reads "Administrator: C:\Windows\system32\cmd.exe". The window content displays the following text: "Microsoft Windows [Version 6.1.7...]", "Autoriõigus (c) 2009 Microsoft C...", and the current directory "C:\Windows\system32>". The window has standard Windows window controls (minimize, maximize, close) and a scrollbar on the right side.

- Vaata tiitliriba!
- Kirjuta „echo %username%“ (kehtib kui käivitatad cmd nt teise kasutaja õigustes).

Windowsi käsureda käske:

- DIR
- CD /D E:\KAUST
- CD ..
- DEL *.*
- COPY D:\xosl B:
- ROBOCOPY
- MOVE
- HELP
- CLS
- ECHO
- START
- EXIT
- MKDIR
- RMDIR
- REN

Käsk ja tema parameetrid

- Käsurealt sisestatav käsk lubab üldjuhul kasutada rohkem või vähem lisaparameetreid.
 - Need on käsust ja üksteisest eraldatud **tühiku**(te)ga;
 - Nende kasutamine mõjutab käsu toimimist.
- Näiteks:
`dir`
`dir /A`
`dir /A /X`

ABI

- Windows XP abiinfost sai otsida (lingid veebis olevale infole siinkohal):
 - Command line reference A-Z
 - To configure the command prompt
 - Using batch files
- Käsureal käsk HELP
- Käsk /?

Pakkfailid ja kehtaskriptid

Kuidas realiseerida?

- UNIX
 - Kestafail (*shell script file*)
- Windows
 - Pakkfail (*batch file*)

Millest koosnevad pakk- ja kestafailid?

- Vastava keskkonna käsurea vahenditest.
- Loogilistest operaatoritest ja muutujatest.
- Meenutab lihtsat programmeerimiskeelt.
- On tavalised tekstifailid, millele on antud käivitusõigus või teatud laiend.
- Loomiseks saab kasutada meelepärast tekstiredaktorit.

Pakkfailid

- Omavad vähem võimalusi kui UNIXi kehtaskriptid
 - UNIXis on käsura vahendeid rohkem
 - Vanemad Windowsid (95/98) omavad eriti kasinalt käsura vahendeid
 - Windows 2003 on kaasas hulk uusi käske.
 - Lisaks võib installeerida ka programme käsuraalt täitmiseks
 - Cygwin – lubab kasutada UNIXi keskkonda Windowsis
 - **Windows Powershell** – võimsamad skriptid

Pakkfaili ehitus

- Faili võib kirjutada samu käske, mida kasutatakse käsurealki.
- Näiteks:

```
programminimi.exe
```

```
"c:\kogu tee\programmi nimi.exe"
```

```
cd /d e:\kataloog
```

```
for /d %%a in (*) do if /i not  
  "%%a"=="admin" rmdir /s /q "%%a"
```

```
notepad kood.txt | programm.exe
```

```
echo tere kala
```

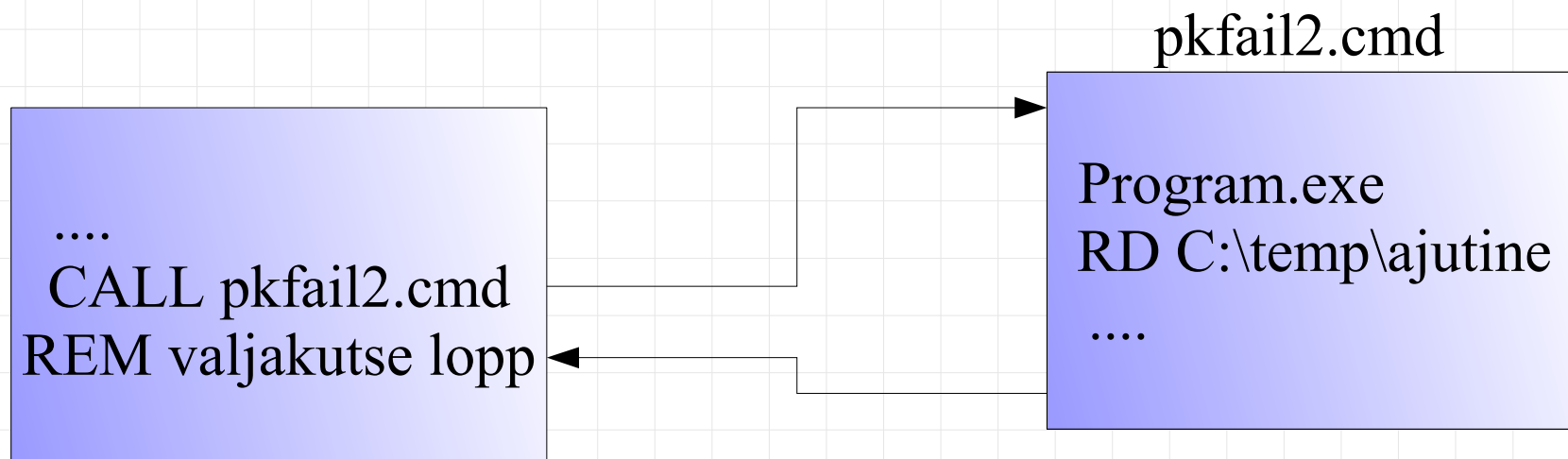
Pakkfailid

- Võib olla kaht tüüpi laiendit
 - .BAT – käivitatakse command.com abil (W7 - cmd)
 - **.CMD** – käivitatakse cmd.exe abil
- Pakkfaili loomiseks salvesta oma tekstiredaktoriga loodud käskude jada laiendiga .CMD
 - .BAT laiendi kasutamisel ei saa kasutada kõiki võimalusi.

Vahendid pakkfailides tegutsemiseks

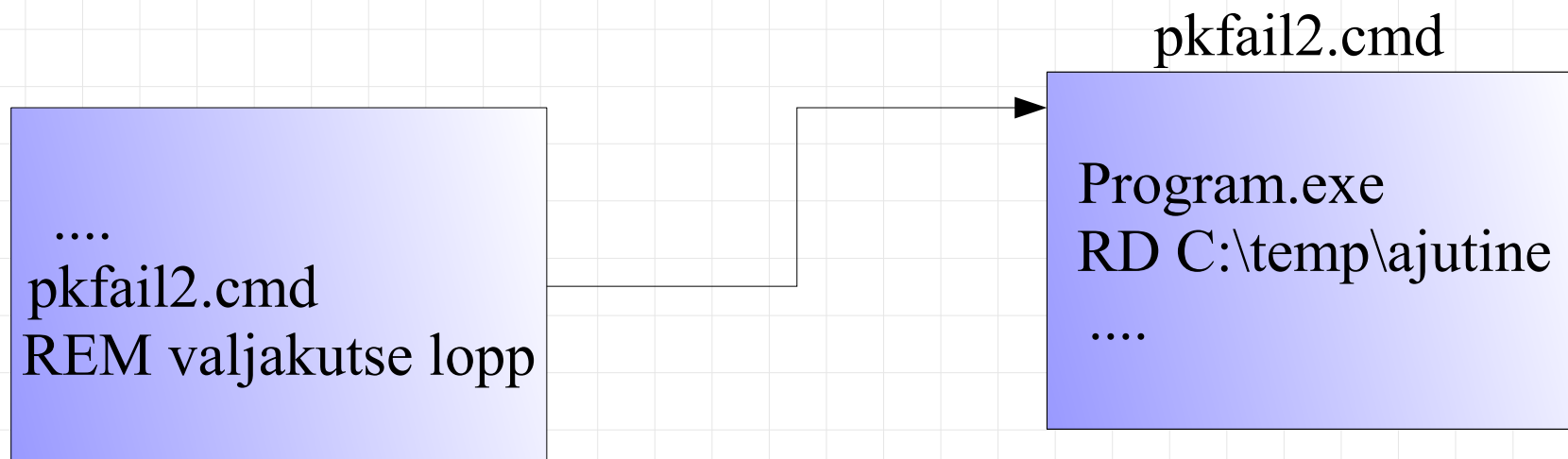
- `CALL` – kutsutakse välja teine pakkfail, pärast jätkatakse esimese tööd pooleliolevast kohast
- `@ECHO OFF` – lülitatakse kaja välja
- `CLS` – puhastatakse ekraan
- `PAUSE` – peatatakse töö kuni kasutaja sekkumiseni
- `REM` – kommenteerimiseks

CALL kasutamine pakfailides



- CALL kasutamine võimaldab välja kutsuda teist pakfaili. Selle töö lõpetamisel jätkatakse esimese faili tööd

CALL kasutamine pakfailides



- Ilma CALL väljakutseta lõpetatakse esimese pakfaili täitmine teise väljakutsumisel.
- Töö lõppeb teise pakfaili töö lõpuga.

Vahendid

- CHOICE – kasutaja saab sisendist valikuid teha
- ENDLOCAL
- FOR - tsükkel
- GOTO – pakfailis mingile käsule liikumiseks
- IF - tingimus
- SETLOCAL
- SHIFT – lubab opereerida teadmata arvu parameetritega

Pakkfaili parameetrid

- Nagu käsule, võin ka pakkfailile anda järgi parameetreid, mida pakkfailis olevad käsud kasutavad.
- Kui parameetriks lisati nt kaust, siis selle kirjutamiseks on mitmeid erinevaid võimalusi:
 - C:\program files <== Tühiku tõttu arvatakse, et on 2 parameetrit!
 - „Program files“
 - ../“Program files“
 - „../../program files“

Pakkfaili parameetrid

- Parameetrina antud faili/kausta erinevate osade kasutamiseks on järgmised muutujad:

%~1 <== 1. param. nii nagu esitati, va „“

%~f1 <== 1. param. täisteedega

%~d1 <== 1. param. kettadraivi täht

%~p1 <== tee 1. param. (va fail ise)

%~n1 <== 1. param. failinimi

%~x1 <== 1. param. faililaiend.

%~s1 <== 1. param. lühikeste nimedega

%~a1 <== 1. param. atribuudid

%~t1 <== 1. param. loomis/muutmisaeg

%~z1 <== 1. param. failisuurus

%~\$PATH:1 <== otsib määratud teelt 1. parameetrit ja laiendab selleni

%~dp1 <== 1. param. kettatäht ja tee

%~nx1 <== 1. param. nimi ja laiend

...

Sisend-väljundi suunamine (Torud)

- **>** - väljund saadetakse faili või seadmele
- **<** - sisend saadakse failist
- **>>** - väljund kirjutatakse olemasoleva faili lõppu
- **>&** - väljundite erinevatesse kohtadesse saatmiseks: `dir>C:\fail.txt 2>&1`
- **<&**
- **|** - eelmise käsu väljund läheb teisele sisendiks
- **%1** – esimene parameeter

Näited

Näide

- Kirjutada skript, mis saab ühe parameetri ja otsustab, kas tegemist on kausta või failiga. Kui on tegemist kaustaga – antakse vastuseks KAUST, kui fail, siis FAIL.

Ülesande analüüs

- Kontrollida, kas failile antakse sisend;
- kontrollida, kas sisendina esitatud asi on ka tegelikult olemas;
- kontrollida, kas on tegemist faili või kaustaga.
- Valida välja sobivad käsud (parameetrid).

Windows

- Uurida käske IF ja DIR ja nende laiendusparameetreid

- **@ECHO OFF**
- **IF /I EXIST %~f1 goto OLEMAS**
- **:VIGA**
- **ECHO : Sisestatud faili/kausta ei eksisteeri!**
- **GOTO END**
- **:OLEMAS**
- **IF /I NOT EXIST %~f1\nul ECHO %~nx1 on FAIL**
- **IF /I EXIST %~f1\nul ECHO %~n1 on KAUST**
- **:END**

Regulaaravaldised

- Regulaaravaldis – mustrikirjeldus
- Kasutatakse kindlate tekstiridade otsimiseks:
 - Unix süsteemides
 - Kontoritarkvaras
 - ...
- Mustreid otsitakse tekstiridade kaupa

Regulaaravaldised

- http://en.wikipedia.org/wiki/Regular_expression - Wikipedia artikkel regulaaravaldiste kohta
- <http://kuutorvaja.eenet.ee/programmeerimine/regulaaravaldised.html> - eestikeelne artikkel regulaaravaldiste kohta

Milleks kasutada regulaaravaldist?

- dokumentatsioonis sobiva koha leidmiseks;
- süsteemi logidest teatud mustriaga kirjete leidmiseks;
- mingi programmi töö tulemusena saadud väljundist vajalike kirjete leidmiseks;
- ...

KÕIK