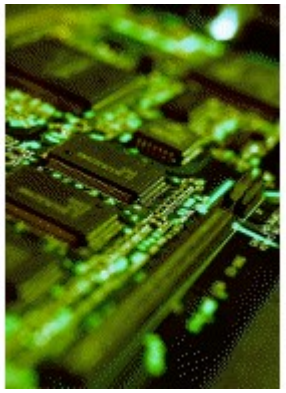
Systemy Operacyjne i Sieci Komputerowe



Sprzęt komputerowy

System Operacyjny
+
Programy

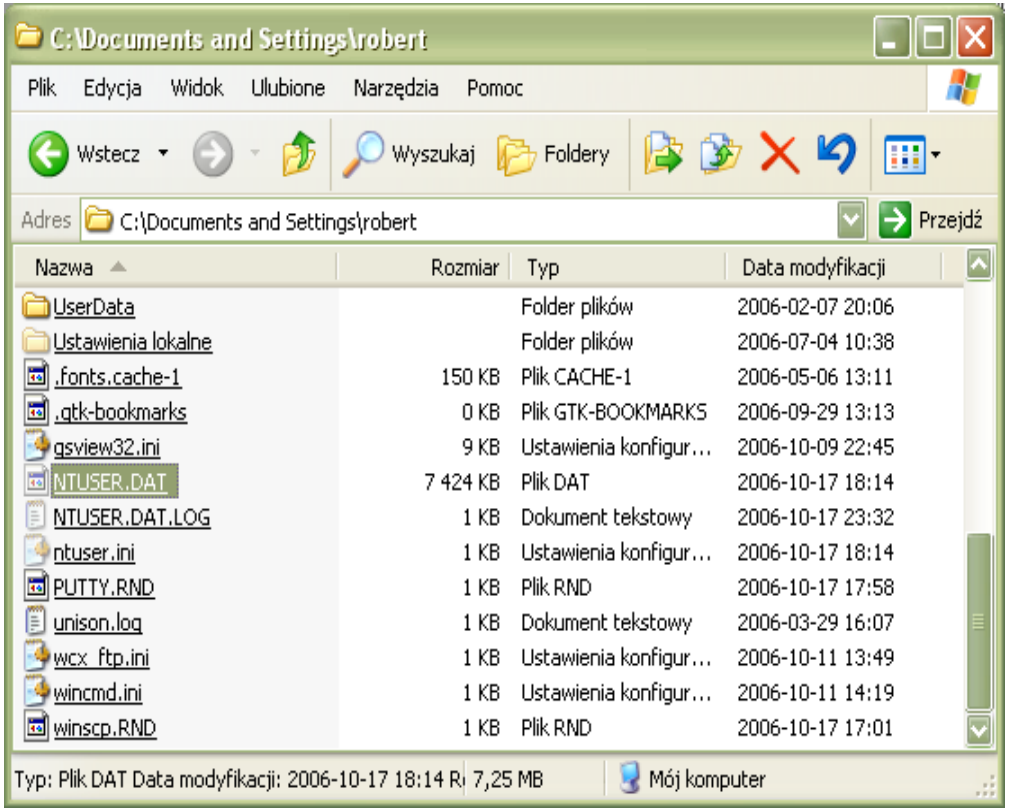
Łatwe użytkowanie

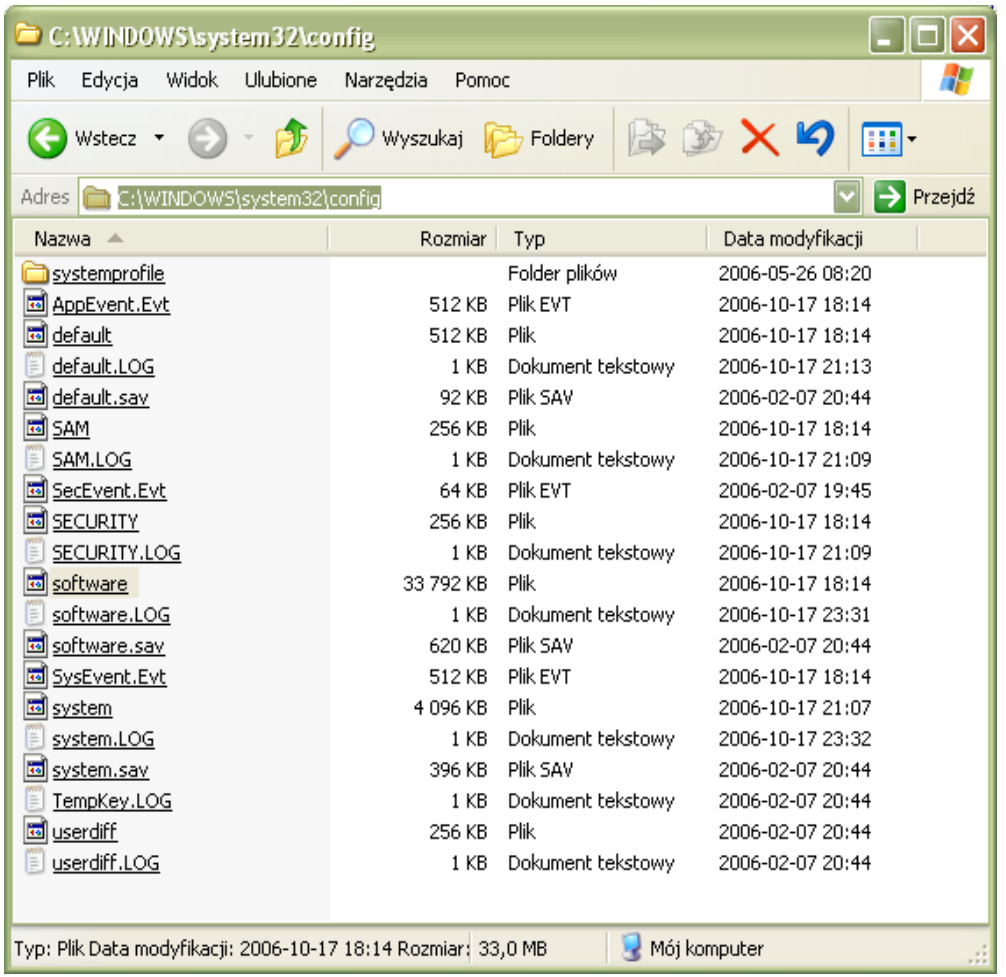
Prowadzący: *Robert Szmurło*
szmurlor@iem.pw.edu.pl
GE 229

Konfiguracja Windows - Rejestr

Binarny rejestr systemowy rozbity na dwie części:

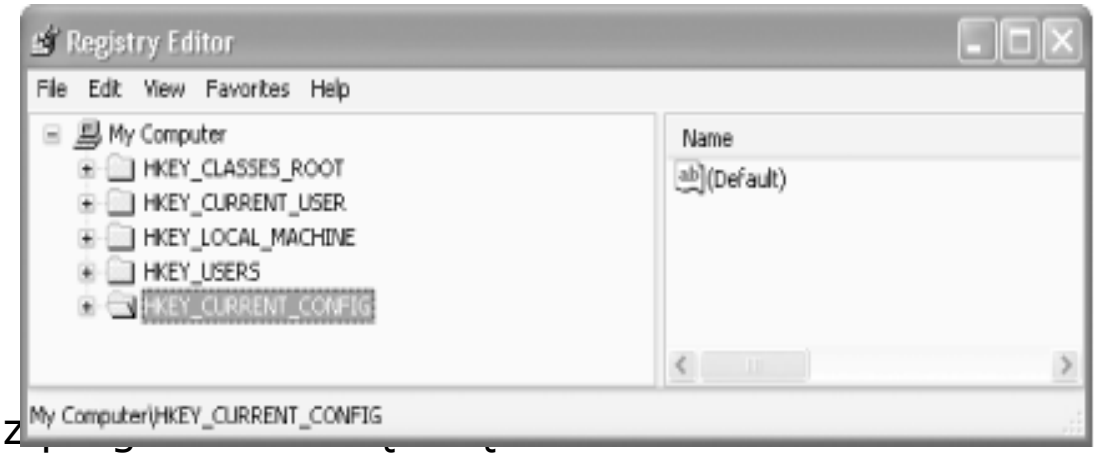
- Część systemowa (HKEY_LOCAL_MACHINE)
- Część użytkownika (HKEY_CURRENT_USER)





Rejestr Systemowy

Składa się z pięciu korzeni:



- HKEY_CLASSES_ROOT – informacje o typach plików oraz skojarzeniach z odpowiednimi rozszerzeniami,
- HKEY_CURRENT_USER – zawiera konfigurację systemu oraz [illegible] zalogowanym użytkownikiem,
- HKEY_LOCAL_MACHINE – konfiguracja komputera oraz zainstalowanego systemu operacyjnego,
- HKEY_USERS – informacje o profilach wszystkich użytkowników utworzonych na danym komputerze,
- HKEY_CURRENT_CONFIG – konfiguracja aktualnego profilu sprzętowego.

Rejestr Systemowy

Klucze, podklucze oraz ich wartości:

- REG_SZ – napis,
- REG_MULTI_SZ – tablica napisów,
- REG_EXPAND_SZ – napis wskazujący ścieżkę do pliku,
- REG_BINARY – wartości binarne,
- REG_DWORD – liczby całkowite.

- Źródło podpowiedzi i ciekawostek związanych z rejestrem: http://www.winguides.com/registry/

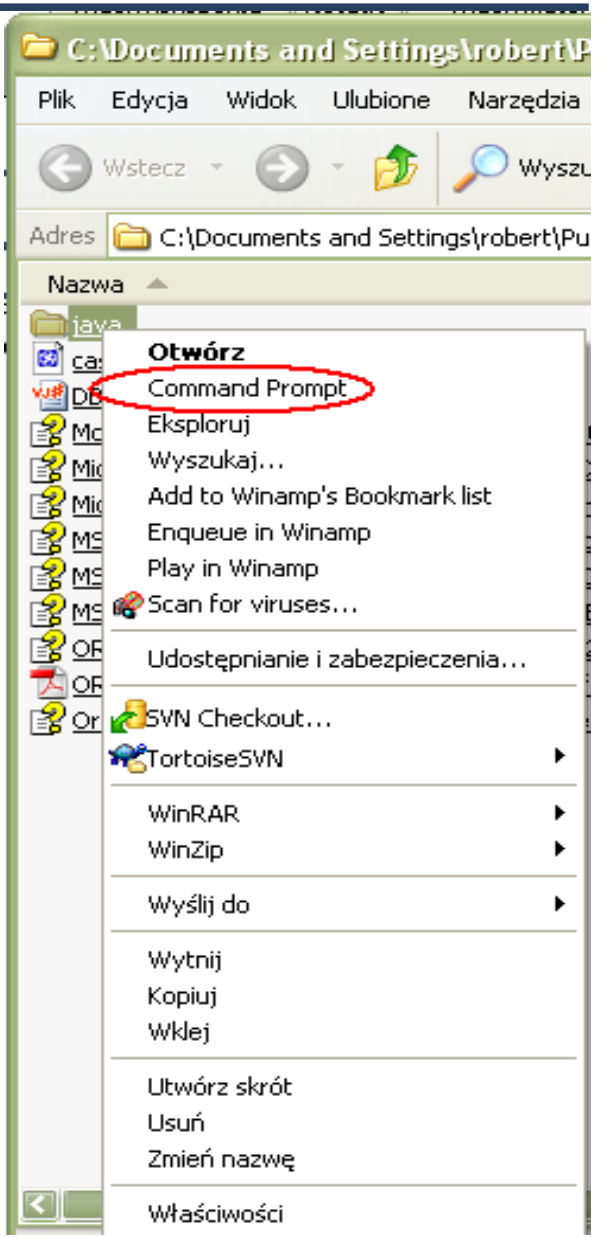
Konfiguracja Windows – Panel Sterowania

Wszyscy znamy panel sterowania.

- W narzędziach administracyjnych znajdziemy Zarządzanie komputerem.
- Panel sterowania zbudowany jest z apletów:
  - System Properties: sysdm.cpl
  - Display Properties: desk.cpl
  - Network Connections: ncpa.cpl
  - Add or remove programs: appwiz.cpl
  - Add Hardware Wizard: hdwwiz.cpl
  - Internet Properties: Inetcpl.cpl
  - Region and Language Options: intl.cpl
  - Sound and Audio Devices: mmsys.cpl
  - User Accounts: nusrmgr.cpl
  - ODBC Data Source Administrator: odbccp32.cpl
  - Power Options Properties: Powercfg.cpl
  - Phone and Modem Options: telephon.cpl

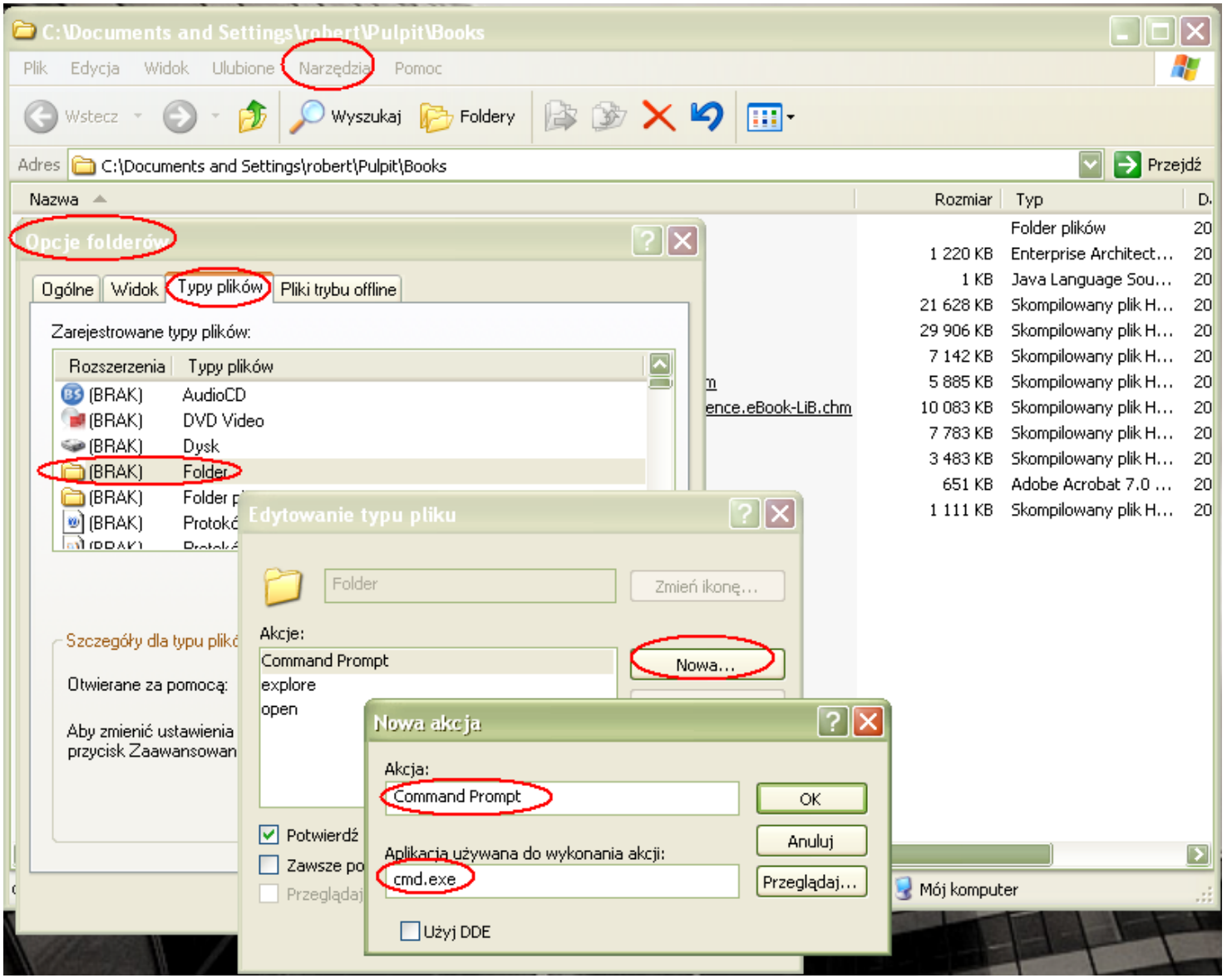
Przykład rozszerzenia możliwości

Zadanie: Pragnę, aby po rozwinięciu menu podręcznego na folderze, pojawiła się możliwość uruchomienia linii komend z bieżącym katalogiem ustawionym na aktualnie zaznaczony.



Pierwsze rozwiązanie

- Dla „klikaczy".



Drugie rozwiązanie

Dla „hakerów" rejestru windows.

1. Otwórz edytor rejestru i przejdź do klucza

   ```
   HKEY_LOCAL_MACHINE/Software/Classes/Folder/Shell
   ```

   utwórz klucz o nazwie „Command Prompt" (bez cudzysłowów).

2. W polu wartości domyślnej wprowadź napis, który pragniesz aby pokazywał się w menu (np. „Prompt here").

3. Utwórz nowy podklucz o nazwie „command" w przed chwilą stworzonym i ustaw wartość pola domyślnego na

   ```
   Cmd.exe /k pushd %L
   ```

   Czasami potrzeba użyć zmiennej systemowej %SystemRoot% w przypadku gdy cmd.exe nie może zostać odnaleziony.

4. Zmiany powinny być widoczne natychmiast. Kliknij prawym przyciskiem myszki na wybranym folderze.

Konfiguracja Unix

- Szereg rozproszonych plików tekstowych:
  - /etc
  - /usr/local/etc
- Zmienne środowiskowe
  - export EDITOR=vi
- Konfiguracja użytkownika w katalogu domowym w plikach ropoczynających się od kropki (czyli plikach ukrytych):
  - /home/user/.configrc
- Problem: bak standardu. Każdy program przechowuje konfigurację we własny sposób. Stosowane są jedynie pewne zalecenia. Najpopularniejszym przykładem formatu jest tzw. format 'ini':
  [Nazwa sekcji]
  Zmienna=Wartosc

Systemy Plików

- Pliki i system plików to integralna część systemu operacyjnego! Praktycznie wszystkie czynności administracyjne wymagają modyfikacji w plikach.
- Techniczne aspekty systemów plików, czyli fizyczna reprezentacja danych na dyskach jest bardzo różnorodna.
  - Unixie: drzewo i-node'ów,
  - Windows NT: podobnie jak w Unix, drzewo
  - DOS, Windows 3.x: struktura tablicowa (tablica FAT – file allocation table)
- Systemy transakcyjne i nie transakcyjne:
  - Nie transakcyjne: Linux: ext2, Windows: FAT32
  - Transakcyjne: Linux: ext3, ReiserFS, Windows: NTFS

Porównując Unix z Windows

- Pomimo różnych struktur plików i katalogów oba systemy posiadają takie same podstawowe narzędzia (chociaż inaczej się nazywają :-))

Zestawienie komend z Unixa i Windows

SO Unixowy	Windows
chmod	CACLS
chown	CACLS
chgrp	No direct equivalent
emacs	Wordpad or emacs in GNU tools
kill	kill command in Resource Kit
ifconfig	ipconfig
lpq	lpq
lpr	lpr
mkfs/newfs	format and label
mount	net use
netstat	netstat
nslookup	nslookup
ps	pstat in Resource Kit
route	route
setenv	set
su	su in resource kit
tar	tar command in Cygnus tools
traceroute	tracert

Porównanie struktury katalogów

- Znak rozdzielajacy katalogi: \ oraz /
- W unixie jest jeden korzeń całego systemu. W Windows każdy dysk ma własny korzeń.

OS Unixowy	Windows
/usr	%SystemRoot% zazwyczaj wskazuje na C:\WinNT lub C:\Windows
/bin or /usr/bin	%SystemRoot%\System32
/dev	%SystemRoot%\System32\Drivers
/etc	%SystemRoot%\System32\Config
/etc/fstab	Brak odpowiednika.
/etc/group	%SystemRoot%\System32\Config\SAM* (binarny)
/etc/passwd	%SystemRoot\%\System32\Config\SAM* (binarny)
/etc/resolv.conf	%SystemRoot%\System32\DNS*
/tmp	C:\Temp
/var/spool	%SystemRoot%\System32\Spool

Struktura Katalogów - Unix

- **/bin** - podstawowe programy narzędziowe, wykorzystywane zarówno podczas uruchamiania systemu w trybie single user jak i multi user. Wykorzystywane są one do tworzenia otoczenia systemowego.
- **/etc** - większość plików konfiguracyjnych, pliki z konfiguracją systemu używaną do procesu bootowania, tzn rc.conf, rc.local.
- **/usr** - większość aplikacji i programów użytkowych dla użytkowników. Programy tutaj są mniej krytyczne dla działania systemu.
    - /usr/bin
    - /usr/sbin
    - /usr/local
- **/sbin** – większość programów narzędziowych przeznaczonych dla administratorów systemu

Struktura Katalogów - Unix

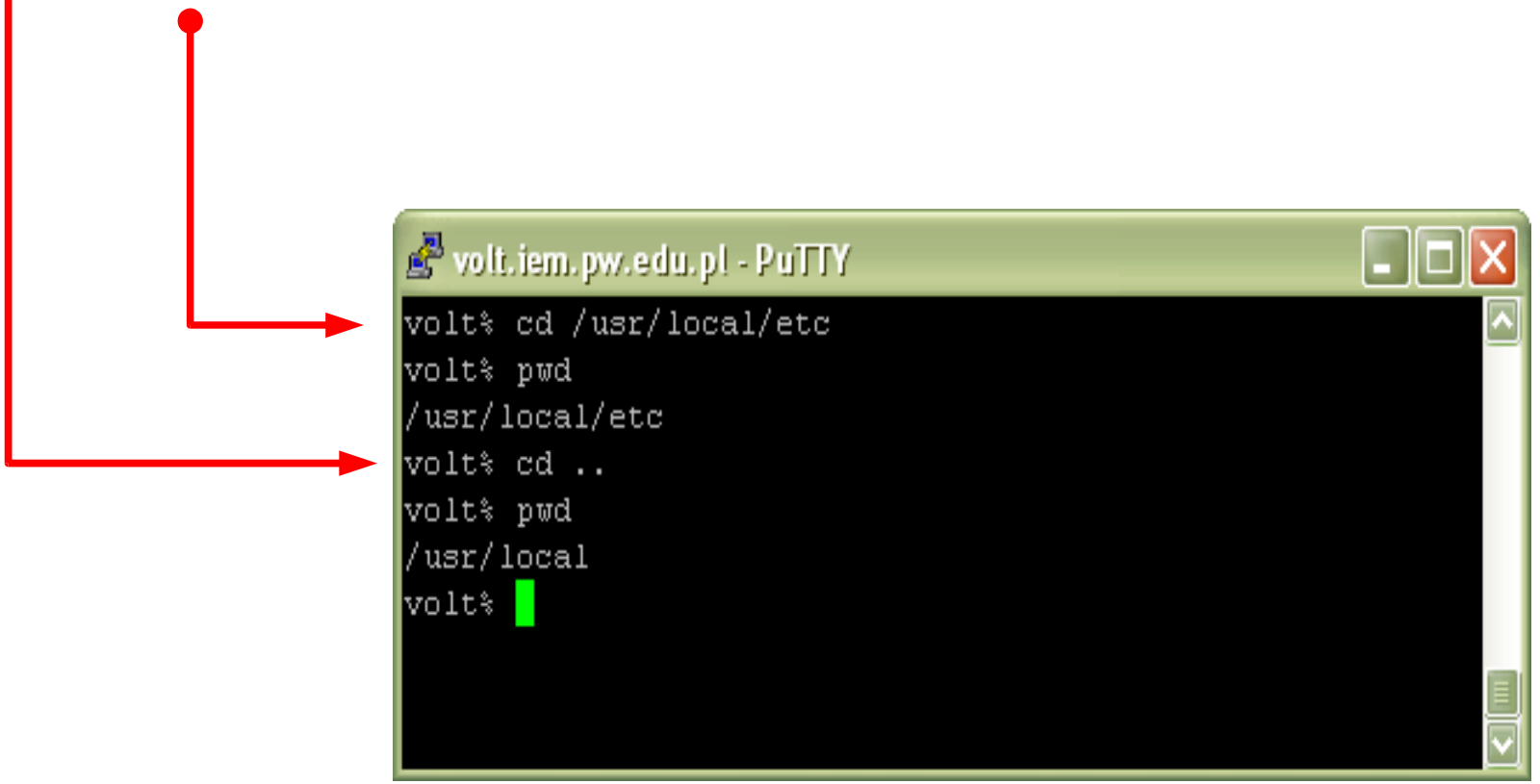
- **/sys** lub **/usr/src** – źródła jądra systemu
- **/dev** - bloki i pliki urządzeń wykorzystywane przez system.
    - W Unixie, każde urządzenie reprezentowane jest jako specjalny plik w tym katalogu. Komunikacja z urządzeniami realizowana jest za pomocą operacji zapisu i odczytu z pliku.
- **/home** – zwyczajowa lokalizacja katalogów użytkowników systemu
- **/root** – katalog domowy administratora systemu (pamiętajmy, że root w unixie może 'wszystko')
- **/var** – katalog gdzie przechowywane są logi o pracy serwera, bufory różnych kolejek itp.
    - /var/mail – skrzynki pocztowe użytkowników
    - /var/log – logi różnych serwisów uruchomionych na serwerze

Struktura Katalogów - Unix

- Każdy katalog w unixie posiada specjalne katalogi wirtualne oznaczone kropką '.' i podwójną kropką '..'. Jedna kropka oznacza katalog bieżący, dwie kropki katalog piętro wyżej w hierarchii.

Struktura Katalogów - Unix

- Nawigacja po katalogach w Unixie.
- Jeden korzeń / dla całego i wszystkich systemów plików.
- Ścieżka względna: nie rozpoczynająca się od znaku /
- Ścieżka bezwzględna: rozpoczynająca się od znaku /



Acykliczny Graf Katalogów - linki

- WINDOWS - skrót do programu lub katalogu, który jest interpretowany jedynie przez program Explorer.
- UNIX - linki są zaimplementowane na poziomie systemu operacyjnego. W unixie występują dwa typy linków:
  - linki symboliczne - przechowują jedynie ścieżkę oraz nazwę do pliku oryginalnego. Przykład użycia:

    ```
    ln –s ../jaki/plik.txt nowy_link.txt
    ```

  - linki twarde - kopia wpisu struktury informacyjnej pliku do nowego miejsca (nowa struktura nadal odowolueje się do tego samego miejsca na dysku fizycznym) Przykład użycia:

    ```
    ln ../jaki/plik nowy_link_twardy.txt
    ```

Co nam pokazuje ls?

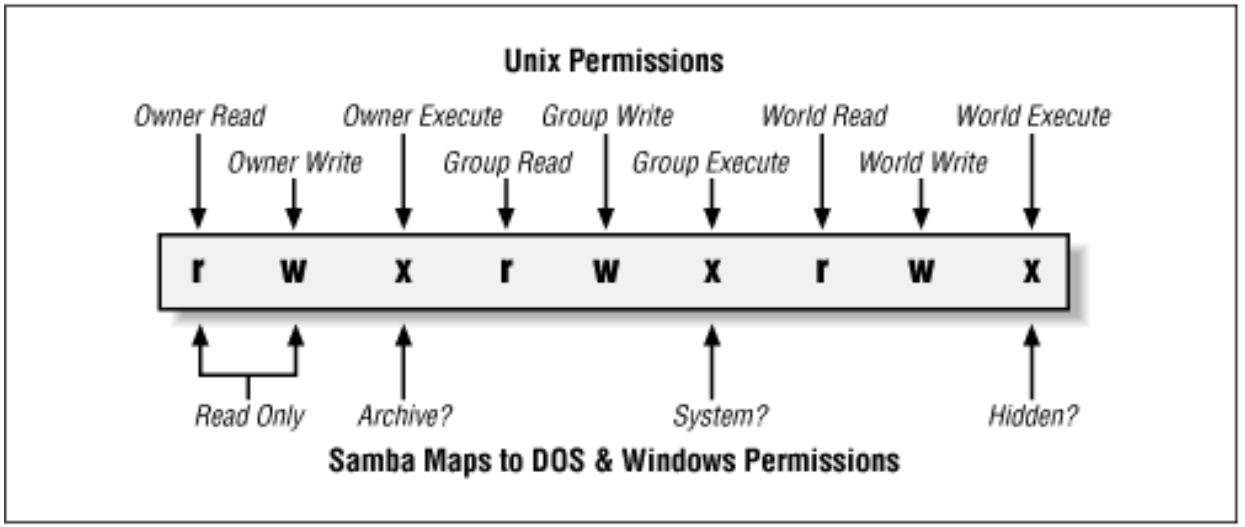
```
volt% ls -la
total 1
drwxr-x---   3 szmurlor  prac    512 11 paź 21:42 .
drwx------  12 szmurlor  prac   1536 11 paź 21:20 ..
drwxr-x---   2 szmurlor  prac    512 11 paź 21:40 katalog
-rw-r-----   1 www        prac      0 11 paź 21:41 plik_pusty
-rw-r-----   1 szmurlor  prac    296 11 paź 21:41 plik_z_zawartoscia
-rw-r-----   1 szmurlor  prac    353 11 paź 21:41 program
-r-xr-x---   1 szmurlor  prac  15324 11 paź 21:42 program1
-rwxr-x---   1 szmurlor  prac    411 11 paź 21:41 program2
volt%
```

Poziomy Uprawnień Użytkowników

- Konto uprzywilejowane:
  - Unix: root (ma automatycznie dostęp do absolutnie wszystkiego)
  - Windows: Administrator (musi mieć udostępnione prawo do danego obiektu, całe szczęście Administrator może udostępniać prawa do absolutnie wszystkich obiektów)
- Konta uprzywilejowane NIGDY nie powinny być używane podczas normalnej pracy użytkownika.
- Otrzymanie uprzywilejowanego konta kojarzy się często z otrzymanie władzy nad innymi użytkownikami, tymczasem zostało ono stworzone ponieważ użytkownicy nie chcieli posiadać zbyt szerokich uprawnień ze względu na odpowiedzialność.

Uprawnienia w Unix

- W Unix każdy proces (uruchomiony program) posiada trzy główne identyfikatory:
  - PID (Process ID) – liczba integer określająca identyfikator procesu,
  - UID (User ID) – liczba integer określająca właściciela procesu,
  - GID (Group ID) – liczba integer określająca grupę procesu.
- Dostęp do zasobów jest określany na podstawie UID i GID.
- Zasoby w Unix posiadają trzy poziomy praw dostępu:
  - dla właściela zasobu (user lub owner),
  - dla grupy będącej właścicielem (group),
  - dla reszty (others).



Zarządzanie Uprawnieniami w Unix

- Zmiana właściciela zasobu:

```
volt% ls -l plik_pusty
-rw-r-----  1 szmurlor  prac  0 11 paź 21:41 plik_pusty
volt% sudo chown www plik_pusty
volt% ls -l plik_pusty
-rw-r-----  1 www  prac  0 11 paź 21:41 plik_pusty
volt% sudo chown :www plik_pusty
volt% ls -l plik_pusty
-rw-r-----  1 www  www  0 11 paź 21:41 plik_pusty
volt%
```

- Zmiana atrybutów:

```
volt% chmod a+rx plik_pusty
volt% ls -l plik_pusty
-rwxr-xr-x  1 szmurlor  www  0 11 paź 21:41 plik_pusty
volt% chmod o-rx plik_pusty
volt% ls -l plik_pusty
-rwxr-x---  1 szmurlor  www  0 11 paź 21:41 plik_pusty
volt% chmod g-x plik_pusty
volt% ls -l plik_pusty
-rwxr-----  1 szmurlor  www  0 11 paź 21:41 plik_pusty
volt% chmod 754 plik_pusty
volt% ls -l plik_pusty
-rwxr-xr--  1 szmurlor  www  0 11 paź 21:41 plik_pusty
volt%
```

Rozpoznawanie typów plików Unix

- Rozszerzenie ma znaczenie tylko umowne. System nie rozpoznaje typu pliku po rozszerzeniu.
- atrybuty plików:
  - + r - prawo do czytania
  - + w - prawo do zapisywania
  - + x - plik wykonywalny
- pierwsza linia w plikach tekstowych:
  - #!/bin/zsh - oznacza że plik jest skryptem napisanym w języku zsh.
  - pliki wykonywalne zapisywane są w specjalnych formatach. Np w linuxie obowiązuje standard ELF. Wstandardzie tym piewsze cztery bajty są zawsze takie same: [0x7f],E,L,F

Uprawnienia w Unix szczegóły

- Komendy:
  - **chmod [-R] [uprawnienia] [nazwa_pliku]**
    gdzie [uprawnienia] mogą być postaci:
    - wyrażenia: a+, u+, g+, o+ oraz a-, u-, g-, o- połączonymi z uprawnieniami rwx,
    - trzema liczbami reprezentującymi binarną postać uprawnień:
      - 777, pierwsza liczba to właściciel, druga grupa, trzecia reszta.
      - Binarnie: 111 jest równe dziesiętnie 4 + 2 + 1 = 7
      - Przykłady:
        - r-x = 101 = 5,
        - rw- = 110 = 6,
        - --x = 001 = 1
        - rw-xr--r-x = 110 100 101 = 645

Rekursywnie przejdź po katalogach

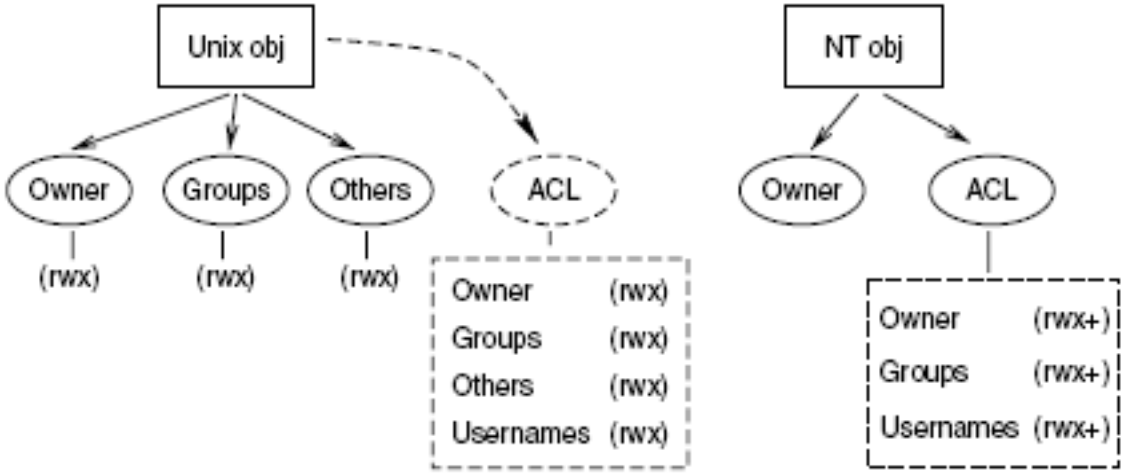
001	1	--x
010	2	-w-
100	4	r--
110	6	rw-
101	5	r-x
-	644	rw-r--r--

Uprawnienia w Unix szczegóły

- Przykładowe czynności:
  - pozwól zapisyć wszystkim: **chmod a+w mojplik**
  - Pozwól właścicielowi uruchamiać: **chmod u+x mojplik**
  - Pozwól wszystkim czytać i urucham.: **chmod 755 mojplik**
  - Ustaw s-bit dla grupy: **chmod g+s mojplik**
  - Przejdź rekursywnie po podkatalogach: **chmod -R a+r .**
- umask – odwrotność chown – zawsze usuwa zaznaczone bity. (Uwaga, nie zmienia bitów nie zaznaczonych).
  - **umask 077** -> powoduje, że plik może mieć rwx --- ---
  - **umask 022** -> odbiera grupie i reszcie prawo pisania czyli odbiera:
    --- -w- -w-

Access Control Lists

- Każdy użytkownik i grupa mogą mieć indywidualne uprawnienia do danych plików.



- W Unix ACL funkcjonują równolegle do standardowych uprawnień i ze względu na skomplikowaną obsługę i kontrolę nie zyskały popularności.

Rozpoznawanie typów plików Windows

- na podstawie rozszerzenia nazwy pliku:
  .exe, .bat, .btm, .sys, .com, .txt, etc.
- struktura danych wewnątrz pliku:
  - .com - plik zawiera tylko kod programu w postaci skompilowanej. Pierwszy bajt pliku to konkretna instrukcja maszynowa.
  - .exe - plik jest w specjalnym formacie. Znakiem rozpoznawczym pliku exe dodatkowo są pierwsze dwa bajty które są zawsze równe: MZ. Pliki exe zawierają oprócz kodu nagłówek, informacje o stosie, stercie oraz pamięci operacyjnej która jest wymagana przez proces.
  - .bat – plik zawiera skrypt, stanowiący zbiór komend

Zarządzanie plikami Unix

- listowanie zawartości katalogu: ls [-l -a -t -r]
- uworzenie katalogu: mkdir [dir]
- usunięcie katalogu: rmdir [dir] lub rm -r [dir]
- przejście do katalogu: cd [dir], cd /dir, cd ../dir, cd ./dir
- Sprawdzenie aktualnego katalogu (.): pwd
- skopiowanie pliku: cp [skadplik] [dokadplik],
- skopiowanie rekursywnie katalogu: cp -r [dir] [dokad]
- usunięcie pliku: rm [plik]
- zmiana nazwy pliku, lub przeniesienie do innego katalogu:
  mv [skad] [dokad]
- wyświetlenie zawartości: cat [plik], more [plik], less [plik]
- utworzenie pustego pliku, lub zmiana daty ostatniej modyfikacji na aktualną:
  touch [plik]

Pomoc w Unix

- Pomoc:
  - Dla posiadaczy internetu: www.google.com
  - Podręcznik złożony z rozdziałów: man [slowo] (np.: man man)
  - Wyszukiwanie haseł: apropos [hasło]
  - System info
  - Dokumentacje aplikacji: /usr/share/doc lub /usr/local/share/doc
- Wyszukiwanie plików i w plikach:
  - find . lub find . -name ''nazwa''
  - find . | grep ''nazwa''
  - locate (oparte na systemowej bazie danych updatedb, która jest uruchamiana zazwyczaj raz dziennie)
  - grep [wyrazenie] [gdzie]
  - whereis [komenda]

Monitorowanie Stanu Systemu

- Obciążenie systemu: top
- Liczba zajętego obszaru na dyskach: df
- Czas od uruchomienia systemu: uptime
- Wyświetlenie wersji systemu: uname -a
- Sprawdzenie ile mamy dozwolonego miejsca na dyskach: quota -v

Konfiguracja środowiska pracy użytkownika

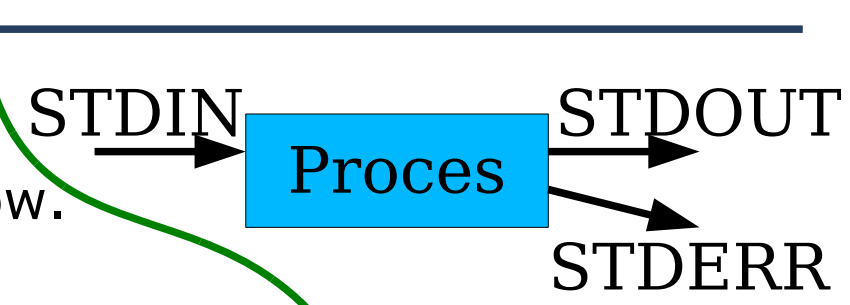
- Powłoka: sh, zsh, bash
- Zmienne środowiskowe:
  - env
  - export EDITOR=pico
  - echo $zmienna
- Pliki konfiguracyjne:
  - /etc/bashrc
  - ~/.bash_profile
  - ~/.bashrc
- Aliasy:
  - alias ll='ls -la'
- Wyświetlanie plików:
  - cat (cat /etc/fstab)
  - more (ls -l | more)
  - less (less /etc/dhcpd.conf)

Czyli na co ma wpływ zwykły użytkownik.

- Zmiana znaku zachęty:
  - export prompt="%n%\:%m% #"
    - Przykładowe znaki:
    - %n nazwa użytkownika,
    - %m krótka nazwa hosta ,
    - %M pełna nazwa hosta (np.: apple.cs.byu.edu)
    - %T aktualny czas w formacie 24h
- Katalog domowy:
  - /home/ziutek
  - Jak szybko przejść do katalogu domoge nie pamiętając dokładnej lokalizacji?
    - cd $HOME
    - cd ~
    - cd

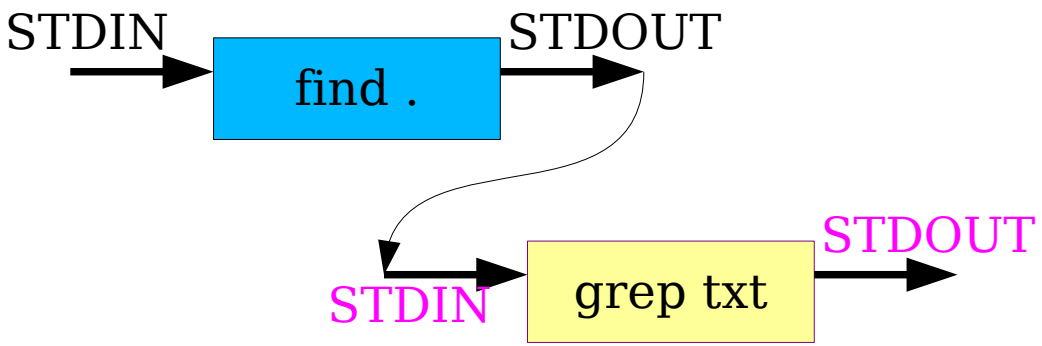
Przetwarzanie Potokowe

- STDIN - Standardowe wejście,
  STDOUT - standardowe wyjście,
  STDERR - standardowe wyjście błędów.
- Programowanie strumieni:
  - przekierowanie do i z plików:
    - ls -la > do_pliku.txt
    - ls -la >> do_pliku_dopisujac.txt
    - grep ala < z_pliku.txt
    - CTRL+C, CTRL+D
  - przkierowanie jednego wyjścia na wejście drugiego:



Przetwarzanie potokowe pomaga analizować większe zbiory danych oraz automatyzować niektóre zadania. Wynik jednego programu jest przekazywany jako dane wejściowe do następnego, itd...

```
find . | grep txt
```



Przykłady Przetwarzania Potokowego

- ps -aux | grep ziutek
- ls -la | more
- find . | grep txt
- make 2>&1 | tee PLIK_LOG
- Z ilu różnych hostów zostały zrejestrowane odpytania naszego serwera www:
  tail -10000 /var/log/www | cut -d ' ' -f 1 | sort | uniq | wc
- Wyświetlanie fragmentów plików:
  - od początku: head -20 /var/log/messages
  - od końca: tail -20 /var/log/messages
- Sortowanie wyników zwracanych przez program:
  - sort

Dalsze Podstawy Linii Komend

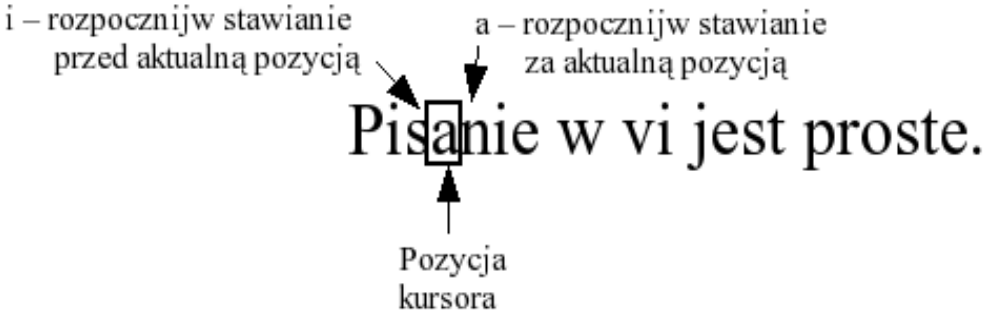
- Pliki ukryte rozpoczynają się od kropki:
  - .nazwa
- Zmienna systemowa PATH definiuje liste katalogów, które są przeszukiwane w celu znalezienia pliku wykonywalnego.
  - export PATH=$PATH:.
  - export PATH=/opt/fx/bin:$PATH
- Znaki specjalne:
  - . - bieżący katalog,
  - | - strumień
  - $ - zmienna
  - ', '' - zmienne tekstowe
  - ` - podstawienie komendy

- Kontrola zadań:
  - zawieszenie zadania: CTRL+Z
  - przerwanie zadania: CTRL+C
  - przywrócenie na wierzch: fg
  - przywrócenie w tło: bg
- Uruchamianie procesów w tle:
  - Na końcu komendy dodać znak &.

Archiwizacja Danych

- Kompresja pojedynczego pliku:
  - gzip [nazwa.txt] (w wyniku powstanie nazwa.txt.gz)
  - bzip2 [nazwa.txt] (w wyniku powstanie nazwa.txt.bz2)
- Dekompresja plików:
  - gzip -d * lub gzip -d [nazwa.txt.gz]
  - bzip2 -d nazwa.txt.bz2
- Scalanie drzew folderów i plików w jeden plik:
  - tar -cvf [n_archiwum.tar] [katalog]
- Przywracanie drzewa z pliku:
  - tar -xvf [n_archiwum.tar] [katalog docelowy]
- Scalanie i przywracanie z kompresją:
  - tar -cvzf [n_archiwum.tar] [katalog docelowy] lub tar -cvjf [n_archiwum.tar] [katalog docelowy]
  - tar -xvzf [n_archiwum.tar]

Edytory Unix



- Wszędzie jest edytor vi.
- Nie wszędzie są: pico, nano, mcedit, joe.
- Podstawowe informacje o vi:
  - vi umożliwia pracę w dwóch trybach: edycji i komend. Standardowo po uruchomieniu edytora pracujemy w trybie komend.
  - a - przejście do trybu edycji (rozpoczęcie dodawania nowego tekstu za znakiem aktualnie zasłoniętym przez kursor).
  - i - przejście do trybu edycji (rozpoczęcie dodawania nowego tekstu przed znakiem aktualnie zasłoniętym przez kursor).
  - r - przejście do trybu edycji tylko na jeden znak (Zastąpienie znaku aktualnie zasłoniętego przez kursor).
  - R - przejście do trybu edycji (rozpoczęcie zastępowania wszystkich znaków).
  - ESC - przejście do trybu komend.
  - : - przejście do trybu komend wpisywanych na ekranie.

Edytory Unix: vi

- Przykład
  - Utworzenie nowego dokumentu wpisanie jednego zdania, zapisanie zmian i wyjście:
    - max# vi nowy.txt
  - Rozpoczęcie dodawania tekstu:
    - [Klawisz i]Pisanie w vi jest proste.[ESC][:][w][q]
  - Objasnienie:
    - [Klawisz i] - ropoczęcie wstawiania
    - 'Pisanie w vi jest proste.' - wpisane zdanie
    - [ESC] - powrót do trybu komend
    - [:] - przejście do trybu komend wpisywanych na ekranie
    - [w] - zapisanie zmian
    - [q] - wyjście z vi
  -

Edytory Unix: vi

- Jak wyjść z vi?
  - [ESC]:q! - wyjście z vi bez zapisywania zmian
  - [ESC]:wq - wyjście z vi z zapisaniem zmian (dokument musi mieć przyporządkowaną nazwę)
- Jak zapisać dokument pod inną nazwą?
  - [ESC]:w nowa_nazwa.txt - zapisanie dokumentu pod nową nazwą
- Jak skopiować i wkleić fragment dokumentu?
  - [ESC]yy - skopiowanie linii w której aktualnie znjaduje się kursor
  - [ESC]dd - wycięcie linii w której aktualnie znjaduje się kursor
  - [ESC]p - wklejenie skopiowanego tesktu za kursorem
  - [ESC]P - wklejenie skopiowanego tekstu przed kursorem

Automatyzacja Zadań

- Pętle w powłoce:
  - for i in [zbior]; do [komenda]; done
- Warunki:
  - if [$a -eq ''iles'']; then [komenda]; fi
- Przykład:
  - Chcemy pobrać z konkretnego adresu internetowego 100 plików o nazwach różniących się liczbą pod konkretnym numerem:
  - for i in `seq 1 100`; do wget http://www.adres/com/plik$i.jpg; done

Automatyzacja Zadań: Skrypty

- Skrypt w unix:
  - musi mieć uprawnienia do wykonywania (chmod u+x skrypt.sh)
  - pierwsza linijka skryptu musi być postaci:
    #!sciezka_do_powloki
    (np: #!/bin/bash)
  - każda linijka skryptu to po prostu komenda
  -

```
#!/bin/sh
if [ "$LANG" = "pl_PL" ]
  then
    echo Czesc
  else
    echo Hello
fi
```

```
#!/bin/bash
for i in $( ls ); do
    echo item: $i
done
```

```
#!/bin/bash
COUNTER=0
while [  $COUNTER -lt 10 ]; do
    echo The counter is $COUNTER
    let COUNTER=COUNTER+1
done
```

```
#!/bin/bash
echo Please, enter your name
read NAME
echo "Hi $NAME!"
```

Automatyzacja Zadań: cron

- Zaplanowane zadania:
  - cron
  - /etc/crontab : ustawienia systemowe
  - crontab -e : zaplanowane zadania jednego użytkownika.
- minuta, godzina, dzień, miesiąc, dzień tygodnia, komenda
- Liczba oznacza wartość o której ma być uruchamiana komenda.
- Konstrukcja: */5 oznacza: uruchamiaj co 5...

```
# Rotate log files every hour, if necessary.
0       *       *       *       *       root    newsyslog
#
# Perform daily/weekly/monthly maintenance.
1       5       *       *       *       root    periodic daily
16      5       *       *       6       root    periodic weekly
31      5       1       *       *       root    periodic monthly
#
# Adjust the time zone if the CMOS clock keeps local time, as opposed to
# UTC time.  See adjkerntz(8) for details.
1,31    0-5     *       *       *       root    adjkerntz -a
# AT
*/5     *       *       *       *       root    LC_ALL= /usr/local/bin/mrtg /usr/local/etc/mrtg.cfg
```

Automatyzacja zadań: Perl

- Pierwsza linijka: #!/usr/bin/perl
- Zmienne: $ - skalarne, @ - tablice, % - mapy (hashe, slowniki)

```
#!/usr/bin/perl
$login = $ARGV[0];
open PASSWD, '/etc/passwd';
while(<PASSWD>){
  @t = split ':';
  $IN{$t[0]} = $t[4];
}
print "$login to ".$IN{$login}."\n";
```

```
#!/usr/bin/perl
while(<STDIN>){
  @t = split ':';
  $IN{$t[0]} = $t[4];
}
@logins = keys %IN;
foreach $login (@logins) {
  print "$login to $IN{$login}\n"
}
```

```
#!/usr/bin/perl
$imie = $ARGV[0];
$i = 0;
while(<STDIN>){
  if ( $_ =~ /$imie/ ) {
    $i = $i + 1;
  }
}
print "W standardowym wejsciu znalazlem $i linii, w ktorych wystapilo imie $imie.\n";
```

Interakcja

- Jeżeli coś cię zainteresowało i chciałbyś aby na następnym wykładzie zostało rozszerzone, powtórzone, omówione dokładniej, to nie krępuj się i napisz maila:

  szmurlor@iem.pw.edu.pl

- Jeżeli coś było nie jasne, napisz maila:

  szmurlor@iem.pw.edu.pl

- Jeżeli coś cię znudziło, napisz maila:

  szmurlor@iem.pw.edu.pl