Linux: administrowanie
wirtualne urządzenie blokowe **loopback**

1.możemy utworzyć system plików który będzie przechowywany w zwykłym pliku:

```
## tworzymy plik o dlugosci 300K
dd if=/dev/hda1 of=mojdysk.img bs=1K count=300
## w pliku tym tworzymy system plikow ext2
mkfs -t ext2 mojdysk.img
```

2.teraz montujemy system plików ...

```
mount -o loop,rw mojdysk.img mojdysk
```

3.używamy ...

4.odmontowujemy ...

```
umount mojdysk
```

Opcja **loop** decyduje że to urządzenie loopback

Punkt montowania (katalog)

Zastosowania loopback:

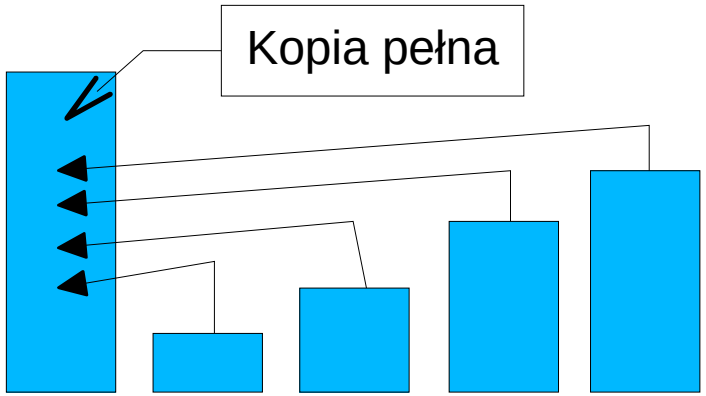
1. przechowywanie linuxowego systemu plików w pliku na partycji nielinuxowej
2. tworzenie "obrazu" dyskietki startowej, którą potem kopiujemy na dyskietki
3. przechowywanie systemu plików w zaszyfrowanym pliku zwykłym

Linux: administrowanie
kopia zapasowa (1)

1. kopie zapasowe tworzy się regularnie, aby w razie awarii dysków (uszkodzenia, włamania) można było odtworzyć możliwie najbardziej aktualną zawartość dysków ... (a także zawartość z danego dnia tygodnia)
2. sposób postępowania:
   a) raz na tydzień tworzy się pełną kopię
      (zawierającą wszystkie pliki/ katalogi)
   b) codzienie tworzy się kopie "częściowe"
      (zawierające zmodyfikowane pliki/ katalogi)
3. metody tworzenia kopii zapasowych:
   a) kopia pełna + kopie **przyrostowe**
      (kopia przyrostowa: zawiera pliki zmodyfikowane i nowe od czasu poprzedniej kopiii przyrostowej)
   b) kopia pełna + kopie **różnicowe**
      (kopia różnicowa: zawiera pliki zmodyfikowane i nowe od czasu ostatniej kopii pełnej)

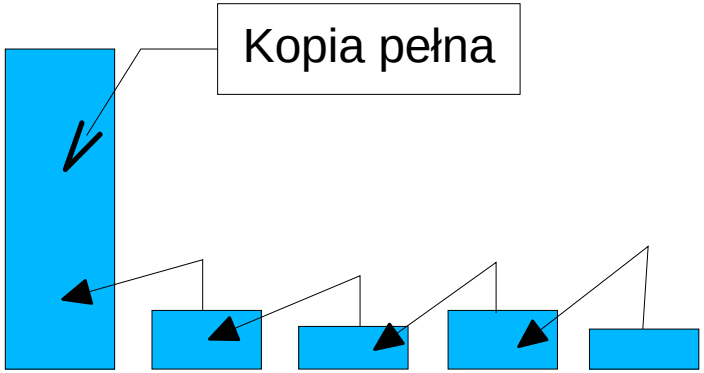
Linux: administrowanie
kopia zapasowa (2)

1. wady i zalety kopii "częściowych" przyrostowych i różnicowych:



Nd Pon Wt Środa ...

Pełna + różnicowe



Nd Pon Wt Środa ...

Pełna +przyrostowe

2. rozwiązanie mieszane: **poziomy** kopii zapasowych

poziom 0: kopia pełna

poziom k, 1≤k≤9: kopia zawierająca wszystkie pliki,
zmodyfikowane lub nowe,
od czasu ostatniej kopii o mniejszym poziomie

Linux: administrowanie
kopia zapasowa (3)

1. poziomy kopii pozwalają uzyskać efekt kopii różnicowych i przyrostowych:
   0 1 1 1 1 1 1 - kopie różnicowe
   0 1 2 3 4 5 6 - kopie przyrostowej
2. polecenia do tworzenia kopii zapasowych:
   a) tar -N

```
tar -N {data} cvf {plik.tar} {katalogi}
## archiwizuje pliki zmodyfikowane po podanej dacie
```

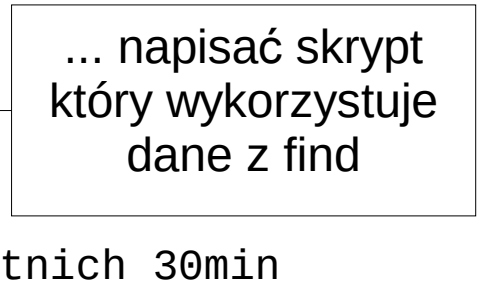
   b) find+tar

```
find . -mmin -30 -print ## wypisuje pliki zmodyf. w ostatnich 30min
find . -mtime -2 -print ## wypisuje pliki zmodyf. w ostatnich 2*24godz
tar rvf qqq.tar eee/a1.txt ## dopisuje plik do archiwum qqq.tar
```

   c) dump+restore (pakiet rpm "dump")

```
## dump -{poziom}u -f {plik.dump} {plik_spec}
dump -0u -f 07062006.dump /dev/hda1
```

   używa poziomów kopii !!!
   opcja -u powoduje że jest uaktualniany plik /etc/dumpdates
   działa tylko dla "prawdziwych" sys. plików ext2/3

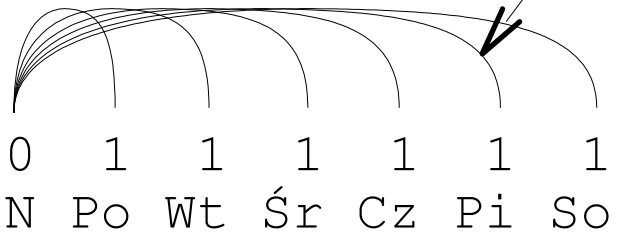


Linux: administrowanie
poziomy kopii (1)

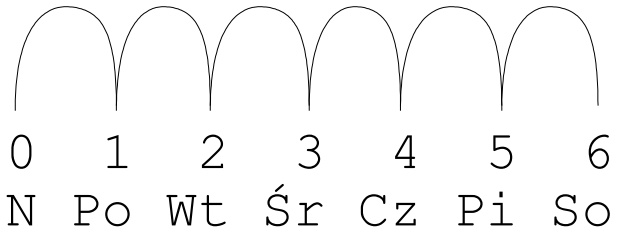
1. Co optymalizujemy stosując “poziomy kopii” ???
   a) Objętość kopii
   b) Liczbę kopii potrzebnych do odtworzenia plików z danego dnia

W **Pi** robimy kopię plików zmodyfikowanych od ostatniej kopii niższego poziomu (czyli od **N**)

Codzienne kopie robimy np o godz 1:05 (rano)

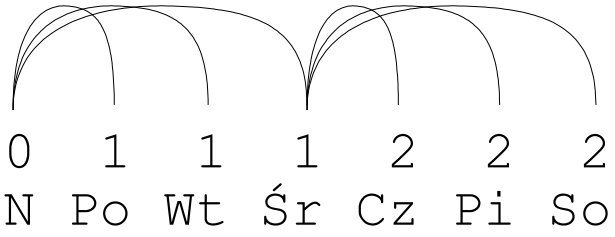


Kopie różnicowe; małe (b), duże (a);
- aby odtworzyć np pliki z Cz potrzebujemy dwóch kopii (z Cz i z N)
- plik zmodyfikowany jedynie w N o 15:00 występuje w 6 kopiach !!!



Kopie przyrostowe; małe (a), duże (b);
- aby odtworzyć np pliki z Cz potrzebujemy 5 kopii (z Cz, Śr, Wt, Po, N)

Linux: administrowanie
poziomy kopii (2)



Rozwiązanie mieszane;
- aby odtworzyć np pliki z Cz potrzebujemy kopii z Cz, Śr, N

W rzeczywistości używa się znacznie bardziej wyrafinowanego ciągu poziomów (TOH – Towers of Hanoi)