Reducing compute time

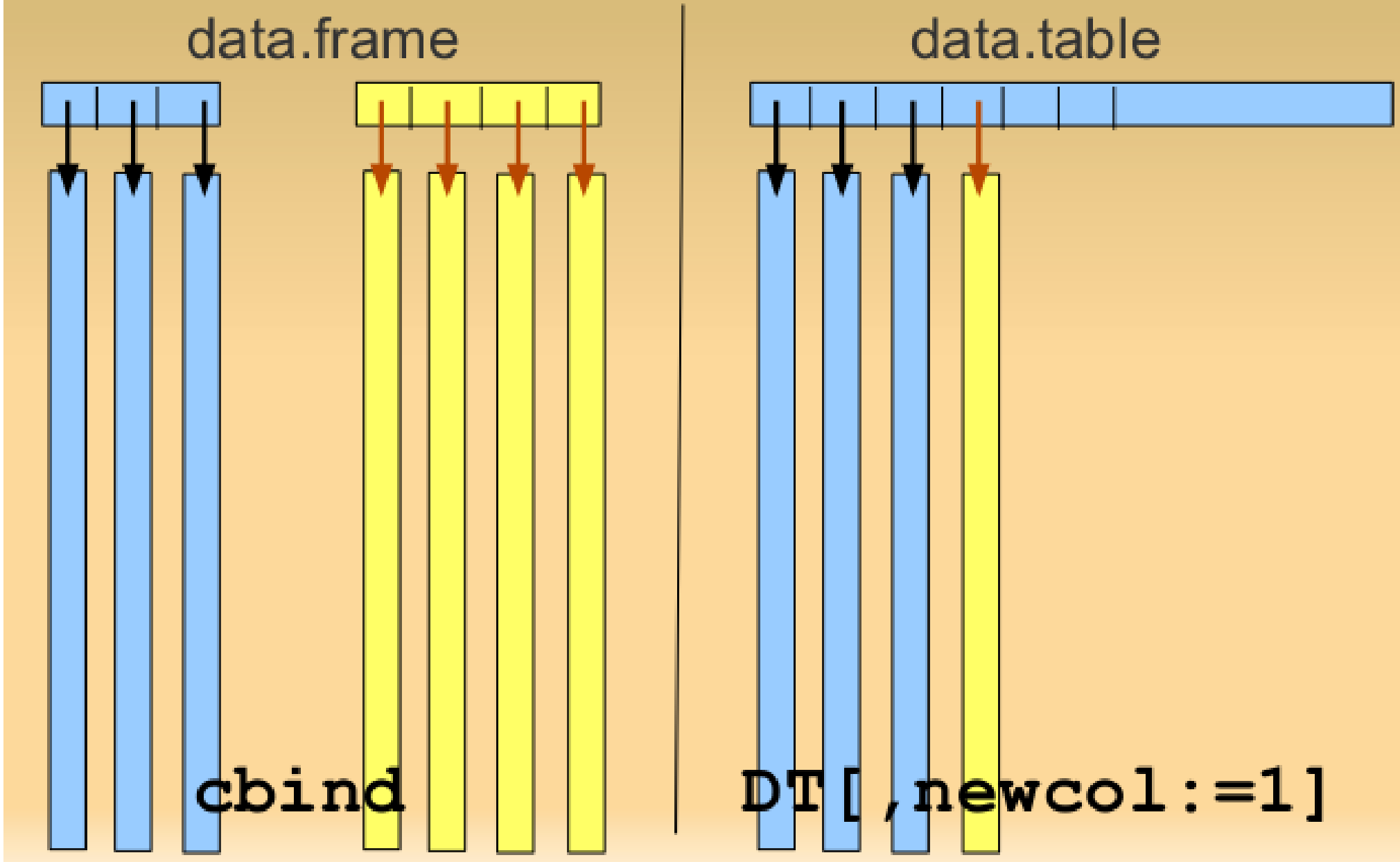
e.g. 10 million rows x 3 columns x,y,v 230MB

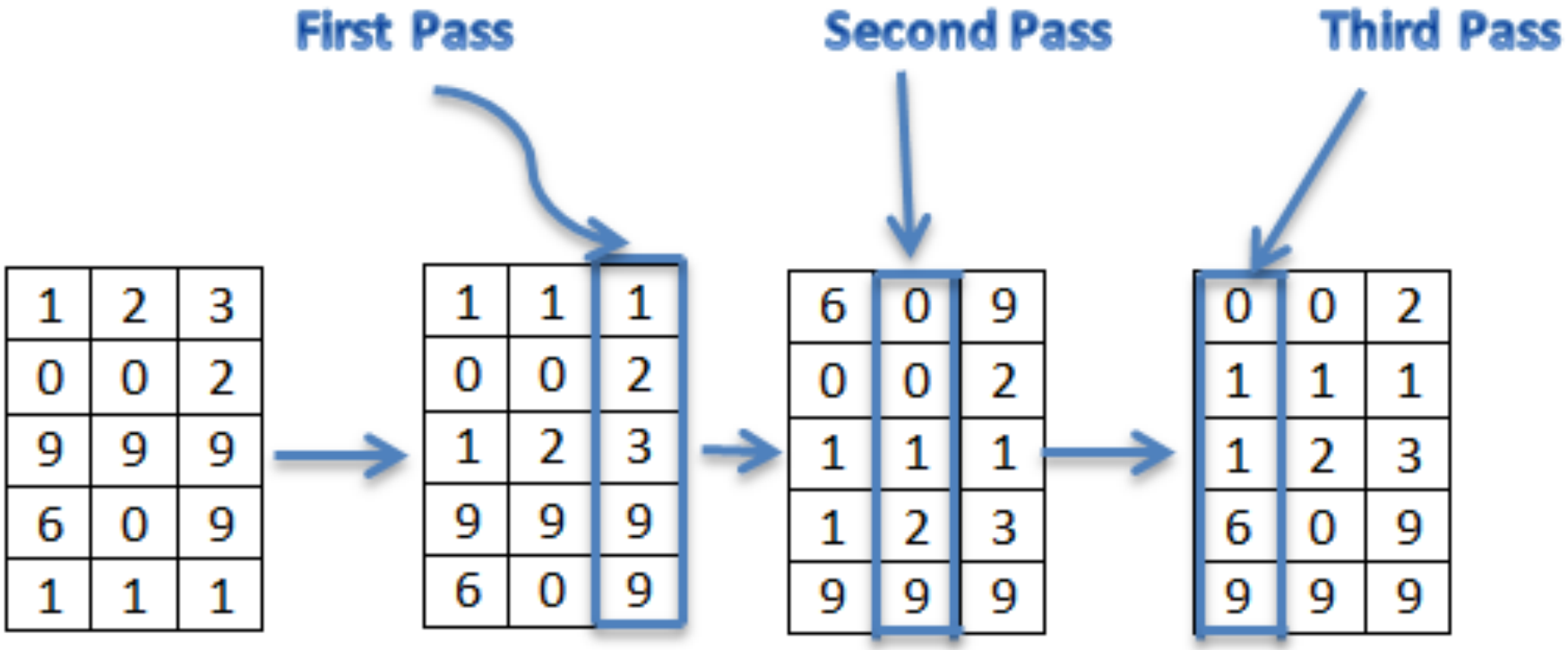
```
DF[DF$x=="R" & DF$y==123,]   # 8     s
DT[.("R",123)]               # 0.008s

tapply(DF$v,DF$x,sum)        # 22    s
DT[,sum(v),by=x]             #  0.83s
```

See above in timings vignette (copy and paste)

Over allocation





source: http://scanftree.com/Data_Structure/radix-sort

`set*` functions

- `set()`
- `setattr()`
- `setnames()`
- `setcolorder()`
- `setkey()`
- `setkeyv()`

Analogous to SQL

```
DT[ where,
    select | update,
    group by]
  [ having ]
  [ order by ]
  [ ]...[ ]
```

Reducing compute time

e.g. 10 million rows x 3 columns x,y,v 230MB

```
DF[DF$x=="R" & DF$y==123,]   # 8     s
DT[.("R",123)]               # 0.008s

tapply(DF$v,DF$x,sum)        # 22    s
DT[,sum(v),by=x]             #  0.83s
```

See above in timings vignette (copy and paste)