System Programming

Basic Unix system administration

System administration

- Thus far, we've only discussed:
  - the use of UNIX from an end user point of view
  - System programming - accesses the core OS but doesn't change the way it operates
- System administration is another level: changing the way the system is set up and operates for end users
- Strongly related to security issues

The superuser

- Most sys. admin. tasks can only be done by the *superuser* (also called the *root* user)
- Superuser
  - has access to all files/ directories on the system
  - can override permissions
  - owner of most system files
- Shell command: `su <username>`
  - Set current user to superuser or another user with proper password access

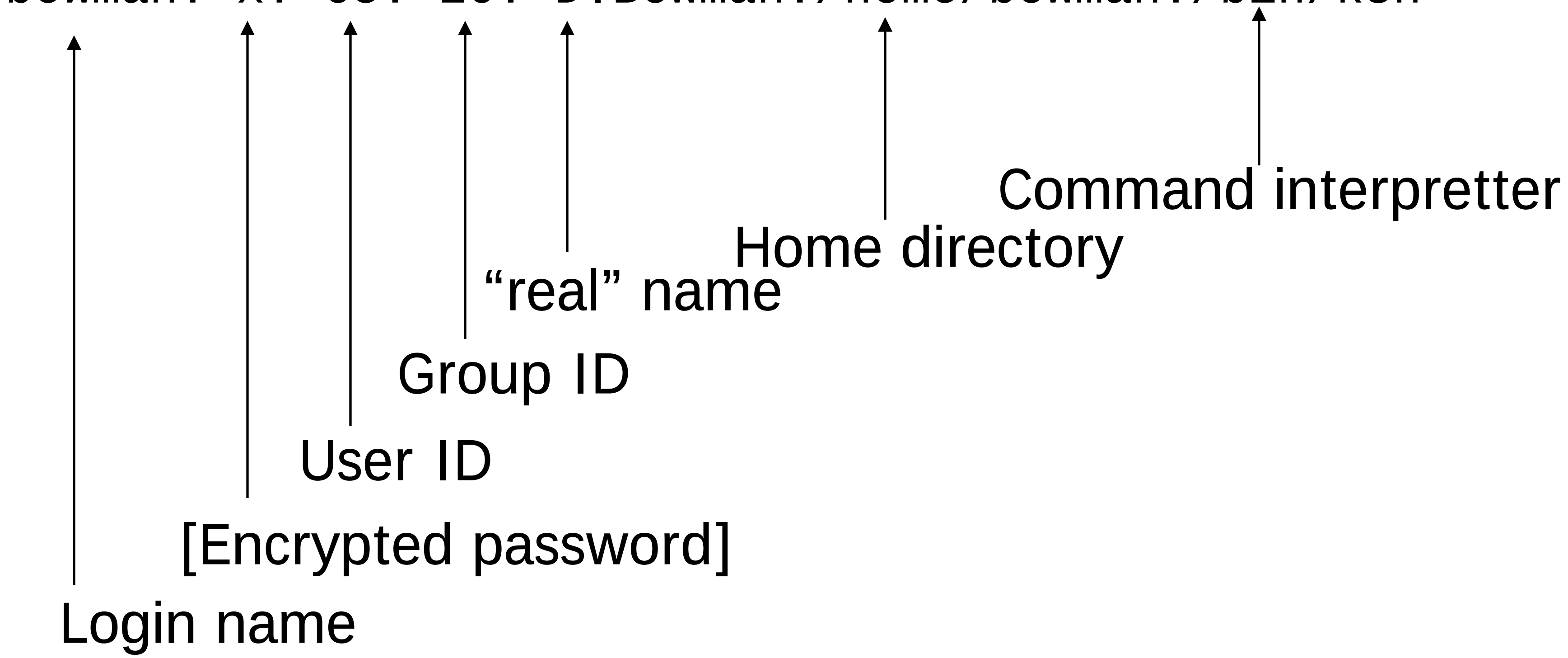
Administration through files

- As you would expect, system settings are stored in files
- Most of these are stored in `/etc`
- We'll look at files related to:
  - Users and groups
  - File systems
  - System initialization
  - System upkeep
- In section 5 of the man pages

/etc/passwd

- Information about system users

```
bowman: x: 65: 20: D.Bowman:/home/bowman:/bin/ksh
```



Command interpretter

Home directory

“real” name

Group ID

User ID

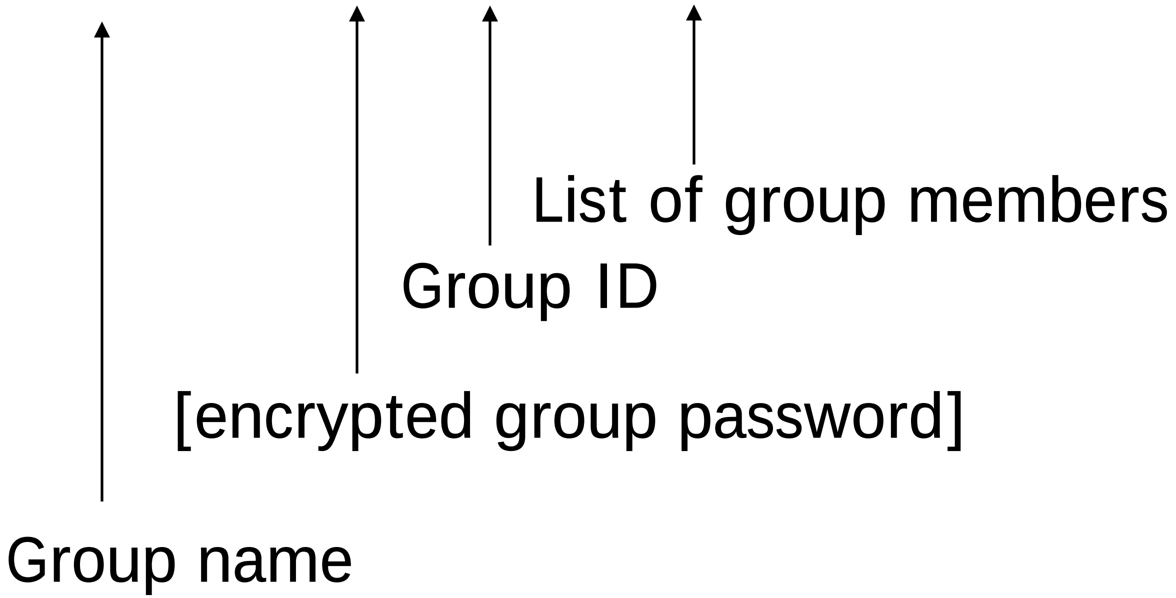
[Encrypted password]

Login name

/etc/group

- Information about system groups

```
faculty: x: 23: bowman, ribbens, mcquain
```



List of group members

Group ID

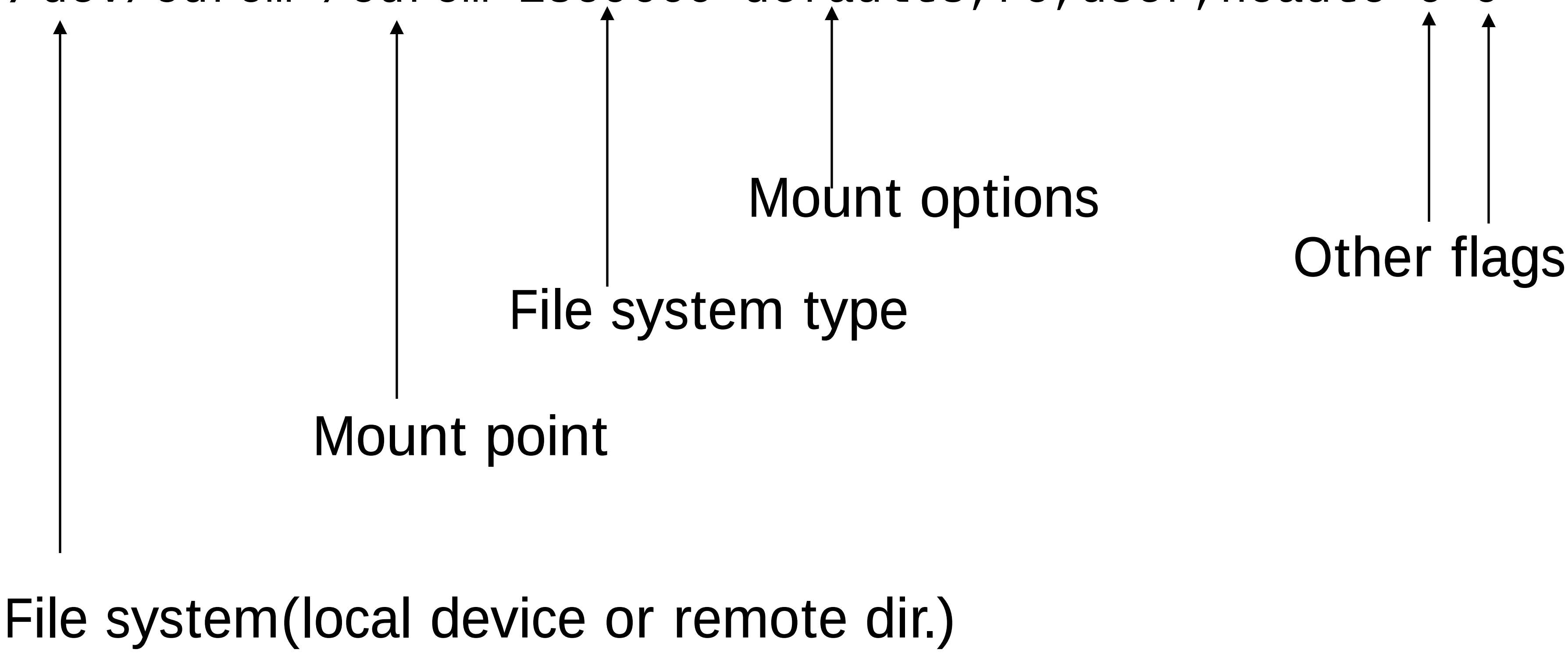
[encrypted group password]

Group name

/etc/fstab

- Information about file systems

```
/dev/cdrom /cdrom iso9660 defaults,ro,user,noauto 0 0
```



Mount options

Other flags

File system type

Mount point

File system(local device or remote dir.)

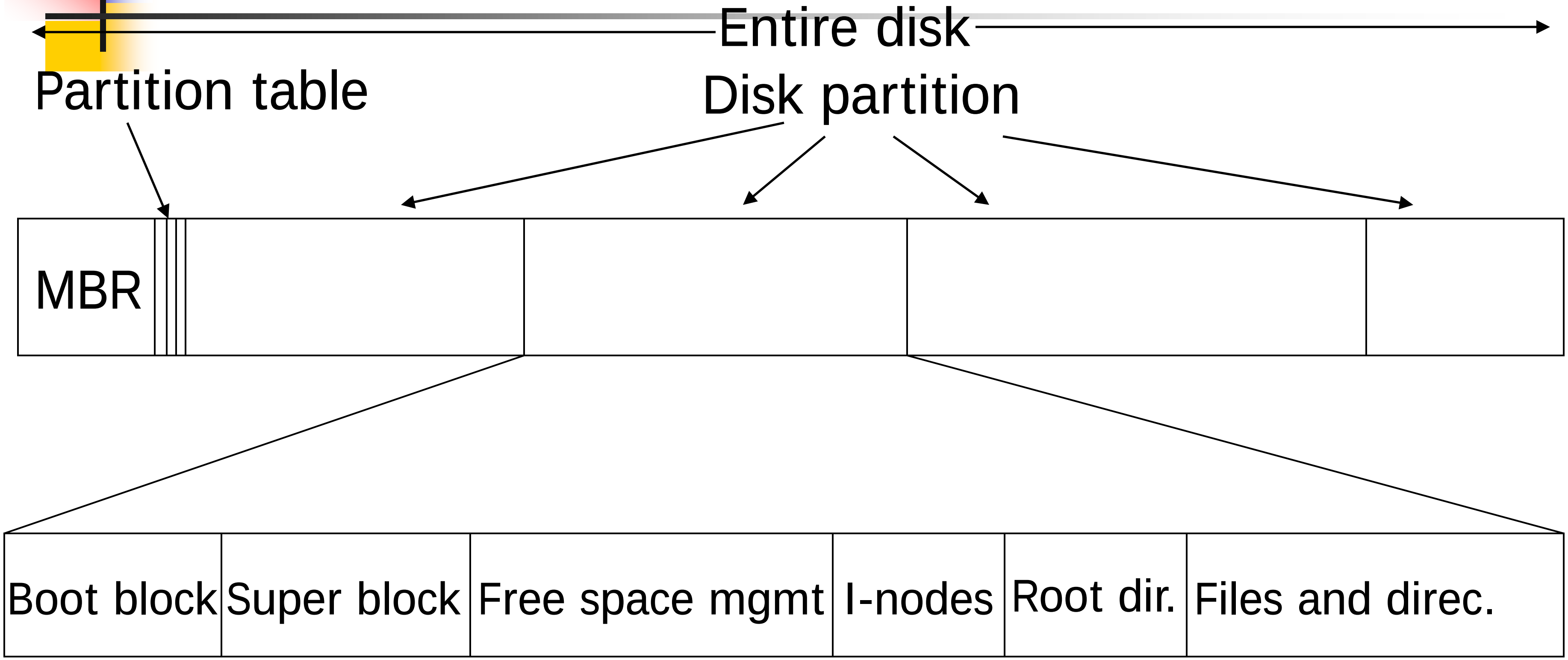
/etc/inittab and /etc/ init.d/

- `inittab:` configuration file for the init process
  - Defines “run levels”
- `init.d:` Directory containing system initialization scripts
  - Script `rc <n>` is run at run level n
  - Starts and stops various services

Configuring the system

- Often by directly editing files with a text editor
- In some cases there are programs that modify the files for you
- Many systems also have nice graphical user interfaces that let you manipulate these files indirectly

File System Implementation



Entire disk

Partition table

Disk partition

MBR

Boot block	Super block	Free space mgmt	I-nodes	Root dir.	Files and direc.

A possible file system layout

Internal View of a File System

- Boot Block:
  - The first block in a UNIX file system, contains the boot program and other initialization information or unused.
- Super Block
  - Always the second block, contains the complete “catalog” of specific information about the file system, including lists of free memory

Internal View of a File System

- Inode list blocks
  - List of inodes for the file system, contiguous and always follows the super block. The number of inodes is specified by the system administrator
    - File access and type information, collectively known as the mode.
    - File ownership information.
    - Time stamps for last modification, last access and last mode modification.
    - Link count.
    - File size in bytes.
    - Addresses of physical blocks.

Internal View of a File System

- Data blocks
  - OS files, user data and program files etc.

File system Commands

- mount
  - mount a file system
- umount
  - unmount a file system
- fsck
  - check and repair a Linux file system
- sync
  - flush filesystem buffers

crontab

- Useful to have a script or command executed without human intervention
  - a script to verify that the networks are working correctly
- cron daemon reads cron configuration files called “crontab” short for “cron table”
  - parse crontabs
  - find the soonest command to be run
  - go to sleep until the command’s execution time has arrived

What’s cron and crontabs?

- Under UNIX, periodic execution is handled by the **<u>cron</u>** daemon
  - read one or more configuration files containing as following
    - command lines
    - times at which they are to be invoked
    - (on some systems)login names under which they are to run
- **<u>crontabs</u>**
  - /etc/crontab

Format of Crontab files

- Seven or six fields
  - minute hour day month weekday [username] command
- an asterisk matches all possible values,
- a single integer matches that exact value,
- a list of integers separated by commas (no spaces) used to match any one of the values
- two integers separated by a dash (a range) used to match any value within the range

Example crontab

```
SHELL=/ bin/ bash
PATH=/ sbin:/ bin:/ usr/ sbin:/ usr/ bin
MAILTO= root
HOME=/

01 * * * * root nice -n 19 run- parts /etc/ cron. hourly
02 4 * * * root nice -n 19 run- parts /etc/ cron. daily
22 4 * * 0 root nice -n 19 run- parts /etc/ cron. weekly
42 4 1 * * root nice -n 19 run- parts /etc/ cron. monthly
```