OS introduction

**What is an operating system?**

* A software to manage a computers resources for its users and applications.

OS roles

* Referee
  + Resource allocation among user, applications.
  + Isolation of different user, application from each other, if two users at the same time.
  + Communication between user, applications.
* Illusionist
  + Each application appears to have the entire machine to itself.
  + Infinite number of processors, (near) infinite amount of memory, reliable storage, reliable network transport.
  + The computer just needs to e.g. store 10gb even if it not there.
  + You want it to behave in a similar way in different scenarios.
* Glue
  + Libraries, user interface, widgets…

**File system as an example with OS rules**

* Referee
  + Prevent users from accessing each other’s files without permission
  + Even after a file is deleted and its space re-used.
* Illusionist
  + Files can grow (nearly) arbitrarily large.
  + Files persist even when the machine crashes in the middle of a save.
* Glue
  + Named directions, printf…

**OS Challenges**

* Reliability
  + Does the system do what is was designed to do?
* Availability
* Security
  + Is important for reliability, availability and privacy. Without security you have no privacy.
* Privacy
* Portability
  + For programs
    - Application programming interface (API)
    - Abstract virtual machine (AVM)
  + For the operation system
    - Hardware abstraction layer
    - The OS does not need to know it is a SSD, just a storing device
* Performance
  + Latency/response time
  + Throughput
    - How many operations can be done per unit of time?
  + Overhead
  + How much extra work is done by the OS
    - When you are in Word you not just running word, but other programs.
  + Fairness
  + Predictability
  + How much consistent is the performance over time?
    - For example, with a memory leak, where a machine is slow after a week, but a restart fix it.

**OS history**