• Reliability via Replication
  • General approach to building fault-tolerance systems
  • Single-disk failures: RAID
How to Design Fault-tolerant Systems in Three Easy Steps

1. **identify** all possible faults
Windows

A fatal exception 0E has occurred at 0028:C0011E36 in VXDI VM0(01) + 00010E36. The current application will be terminated.

* Press any key to terminate the current application.
* Press CTRL+ALT+DEL again to restart your computer. You will lose any unsaved information in all applications.

Press any key to continue _
Your PC ran into a problem and needs to restart. We're just collecting some error info, and then we'll restart for you.

25% complete

For more information about this issue and possible fixes, visit http://windows.com/stopcode

If you call a support person, give them this info:
Stop code: CRITICAL_PROCESS_DIED
You need to restart your computer. Hold down the Power button for several seconds or press the Restart button.

Veuillez redémarrer votre ordinateur. Maintenez la touche de démarrage enfoncée pendant plusieurs secondes ou bien appuyez sur le bouton de réinitialisation.

Sie müssen Ihren Computer neu starten. Halten Sie dazu die Einschalttaste einige Sekunden gedrückt oder drücken Sie die Neustart-Taste.

コンピュータを再起動する必要があります。パワーボタンを数秒間押し続けるか、リセットボタンを押してください。
How to Design Fault-tolerant Systems in Three Easy Steps

1. **identify** all possible faults
2. **detect** and **contain** the faults
3. **handle** the fault
quantifying reliability
dealing with disk failures
700,000 hours ≈ 80 years
(which seems.. suspicious)
dealing with disk failures
RAID 1 (mirroring)

😊 can recover from single-disk failure
😭 requires 2N disks
RAID 4 (dedicated parity disk)

- Can recover from single-disk failure
- Requires N+1 disks (not 2N)
- Performance benefits if you stripe a single file across multiple data disks
- All writes hit the parity disk
RAID 5 (spread out the parity)

😊 can recover from single-disk failure
😊 requires N+1 disks (not 2N)
😊 performance benefits if you stripe a single file across multiple data disks
😊 writes are spread across disks
• Systems have faults. We have to take them into account and build reliable, **fault-tolerant systems**. Reliability always comes at a cost — there are tradeoffs between reliability and monetary cost, reliability and simplicity, etc.

• Our main tool for improving reliability is **redundancy**. One form of redundancy is **replication**, which can be used to combat many things including disk failures (important, because disk failures mean lost data).

• **RAID** replicates data across disks in a smart way: RAID 5 protects against single-disk failures while maintaining good performance.