Anna Östlin Pagh and Rasmus Pagh
IT University of Copenhagen
Advanced Database Technology
April 1, 2004

MEDIA FAILURES

Lecture based on [GUW, 11.6-11.7]

Today

- Volatile storage
  - RAID.
- Concurrency control
  - How transactions cope with living together:
    - Serializability
    - Locking
    - Optimistic concurrency control

The need for volatile storage

- DB systems often store data of large value.
- Losing data is unacceptable (consider e.g. the account information in a bank).
- We need storage that is very robust.
- Problem: Single disks may fail because of head crashes, fire, explosions, etc.

Can't get 100% guarantee

- No matter how data is stored, we can't make it 100% safe.
- However, the probability of losing it can be made negligible, e.g., 10^{-10}/year.
- The basic idea is to introduce **redundancy** such that all data can be recovered even if part of what is stored is lost.

Duplication

- A simple solution is to make one or more copies (**mirrors**) of every disk.
- Data is lost only if all disks fail in a timespan too short to copy data to new disks.
- Extra advantage: Allows several I/Os to be carried out in parallel.
- Disadvantage: Need many disks.

Erasure correcting codes

- If a given piece of data is lost, we want to be able to recover it.
- Data encoded as an **erasure-correcting code** have this property.
- Simple example: To encode bits $b_1,b_2,..,b_i$ store them together with the extra bit $(b_1+b_2+..+b_i)$ mod 2.
- Question: How do we recover b_1 if lost?

RAID level 4

- Data resides on **n** identical disks.
- The last disk contains the sum (mod 2) of bits on the previous **n-1** disks. (This is also known as a **checksum**.)
- Any disk can be recovered from the other disks.
- Advantage: Only one redundant disk.
- Drawback: Last disk is bottleneck.

RAID level 5 and 6

- **Ideas**:
  - Distribute checksums among disks to avoid write bottleneck.
  - Make resistant to multiple simultaneous crashes.
  - Allow detection and correction of erroneous data from one disk (which is rare, but may happen).
- **Example**: 5 data and 2 redundant disks.

Conclusion

- Highly stable disk systems are possible, though either a rather high number of disks or high redundancy is needed.
- For protection against fire, meteorites, etc., disks need to be geographically separated.