RAID-Systeme

Proseminar Speicher- und Dateisysteme
SoSem 2012

Kai Frederking

Gliederung

- RAID – Was ist das? (I)
- 1 Geschichte
  - Historische Situation
  - Das Problem / Der Lösungsansatz
- 2 Implementation(en)
  - Komponenten eines RAID Systems
  - RAID Level
  - Praktische Gesichtspunkte
- 3 Ausblick und Zusammenfassung

Geschichte – Die Ausgangssituation (I)

Festplatten in den 1980ern :

Quelle:
http://de.wikipedia.org/wiki/Datei:IBM3380DiskDriveModule.agr.jpg

- Ein (!) Modul einer IBM 3380 Platteneinheit
- In 1987: 1 GB, Zugriffszeit 20ms, Transferrate 3MB/s
- Ca. $60.000, $60/MB, (1987: 15$), Netzteil 6.600Watt

Geschichte – Die Ausgangssituation (II)

Auch in den 1980ern :

Quelle: http://en.wikipedia.org/wiki/File:5.25_inch_MFM_hard_disk_drive.JPG

- Die Neuerung PC führt zu Massenproduktion „billiger“ Festplatten
- Ca. 100MB, Zugriffszeit 35ms, Transferrate 1MB/s
- Ca. $1.500, $15/MB, Netzteil 10 Watt

Geschichte – Das Problem

- Plattenfehler, insbesondere Headcrashes (auch: „Spanabhebende Datenverarbeitung“), bedrohen die Daten
- Deren Vermeidung erfordert aufwändige, Produktionsverfahren – Platten sind **teuer**.
- Kapazitätssteigerung bei Platten erfolgt zwar analog zu Moore‘s law ...
- ... aber die Übertragungszeit wird mehr und mehr von der Zugriffslatenz dominiert.

Head Crash bei moderner Platte

Geschichte
RAID – Was ist das (II) und woher kommt es

- 1978: Norman Ken Ouchi, IBM, patentiert ein "System for recovering data stored in failed memory unit.“, ähnlich RAID 5, ist aber seiner Zeit voraus.
- 1980: Seagate entwickelt erste 5.25“ Festplatte
- 1981: IBM PC vorgestellt - 5.25“ Floppy Laufwerk(e))
- 1982: SASI (Shugart Associates Standard Interface) wird SCSI
- 1983: IBM PC XT mit 10MB HDD (5.25“ MFM, volle Bauhöhe) kommt raus
- 1986: ANSI standardisiert SCSI
- 1987: Patterson, Gibson und Katz, von der UC Berkeley veröffentlichen „A Case for Redundant Arrays of Inexpensive Disks“
  = „Ein Argument für Redundante Anordnungen Billiger Platten“
- Heute: „Redundant Array of Independent Disks“, eine Fehlbenennung aus Marketingüberlegungen heraus

Geschichte- Der Lösungsansatz (I)

- Vergleichen wir noch einmal Mainframe Disk und PC-Laufwerk 1987:
- Kapazität - 7,5 GB : 0,1 GB
- Preis, ca. - $15/MB : $15 /MB
- Zugriffszeit - 20 ms : 35 ms
- Übertragungsrate - 3MB/s : 1MB/s

- Also: PC kleiner, fast so schnell, Preis/Kapazität gleich, Preis/Leistung überlegen.

Geschichte – Der Lösungsansatz (II)

- Was passiert, wenn wir Daten statt auf einer auf zwei Platten speichern?
  - Wir können schneller auf sie zugreifen wenn sie verteilt sind.
  - Wenn wir sie doppelt halten schützen wir sie vor Verlust.
  - Oft erreichen wir sogar beides.
- Betrachten wir zuerst die Datensicherheit ...

Redundanz als Sicherheitsfaktor – Einschub: Ausfallrate (I)

- Ausfallraten (NICHT: Fehlerraten) von technischen Geräten und Anlagen werden oft als MTBF angegeben, „Mean Time Between Failures“
- Definition nach IEC 60050 (191): *Der Erwartungswert der Betriebsdauer zwischen zwei aufeinanderfolgenden Ausfällen.*
- Die Wahrscheinlichkeit eines Ausfalls im Zeitintervall T ergibt sich zu

  $$p(T) = 1 - \exp(-T / MTBF)$$

- Bei Festplatten ist der erste meist auch der letzte Ausfall (MTTF).
- ...

Redundanz als Sicherheitsfaktor – Einschub: Ausfallrate (II)

- ...
- MTBF für Festplatten typischerweise um 10^6 h, Größenordnung „1 Jahrhundert“. Ausfallwahrscheinlichkeit innerhalb eines Jahres also ca. 1 - exp(-1 / 100) = 1%
- MTBF wird allerdings vom Hersteller unter Laborbedingungen ermittelt und kann in der Praxis erheblich höher ausfallen (Temperatur, Last, mechanische Einflüsse, Vibrationen, ...)
- Maß für Bitfehlerraten: URE – Unrecoverable Read Error. Typisch: $10^{-14..-16}$ (zur Orientierung: 10^{12} = 1 TiBit). Kann bei der Rekonstruktion großer Platten problematisch werden. Mehr dazu später.

Geschichte - Der Lösungsansatz Redundanz als Sicherheitsfaktor (I)

- Einfachster Fall: Datenspiegelung auf gleicher Platte. Platte 2 = Platte 1
- Datenverlust bei Plattenschaden tritt nur auf, wenn die zweite Platte vor Restaurierung des ersten Opfers auch stirbt.
- Nennen wir das Intervall, in dem wir einen Defekt bemerken, die Platte austauschen und die Spiegelung mit der verbliebenen wieder herstellen können Δt.
- $p(\Delta t)$ sei die Wahrscheinlichkeit, dass in diesem Zeitintervall eine Platte ausfällt.
- Nehmen wir für folgendes Beispiel ein Δt von 10h und ein $p(\Delta t)$ von 1/50.000 an ...

Geschichte - Der Lösungsansatz Redundanz als Sicherheitsfaktor (II)

- Läuft das Speichersystem 10.000h (etwas mehr als ein Jahr), dann ...
- fällt eine ungespiegelte Platte mit der Wahrscheinlichkeit
  $1 - (1 - p) \hat{} (10000 / \Delta t) = 1 - (49999 / 50000) \hat{} 1000 = 0{,}02 =$ **2%**
  aus,
- Ein gespiegeltes Paar, bei dem beide innerhalb von 10h ausfallen müssen, mit
  $1 – (1 – p^2) \hat{} (10000 / \Delta t) = 0{,}0000004 =$ **0,00004 %**
- Für den gleichzeitigen Ausfall von zwei aus fünf Platten ist die Wahrscheinlichkeit $\binom{5}{2} = 10$ mal höher.

Geschichte - Der Lösungsansatz Redundanz als Performanzfaktor

- Faktoren der Leistungsfähigkeit:
  - Kapazität
  - Zugriffsgeschwindigkeit
  - Übertragungsgeschwindigkeit
- Mehr Platten = mehr Kapazität minus Redundanz
  - Größere logische Laufwerke möglich
- Redundante Daten auf unterschiedlichen Platten = schnellster Zugriff bestimmt Latenz
- Daten verteilt auf mehrere Platten = paralleler Zugriff bis zur Buskapazität möglich.

Implementationen – Komponenten eines RAID-Systems (I)

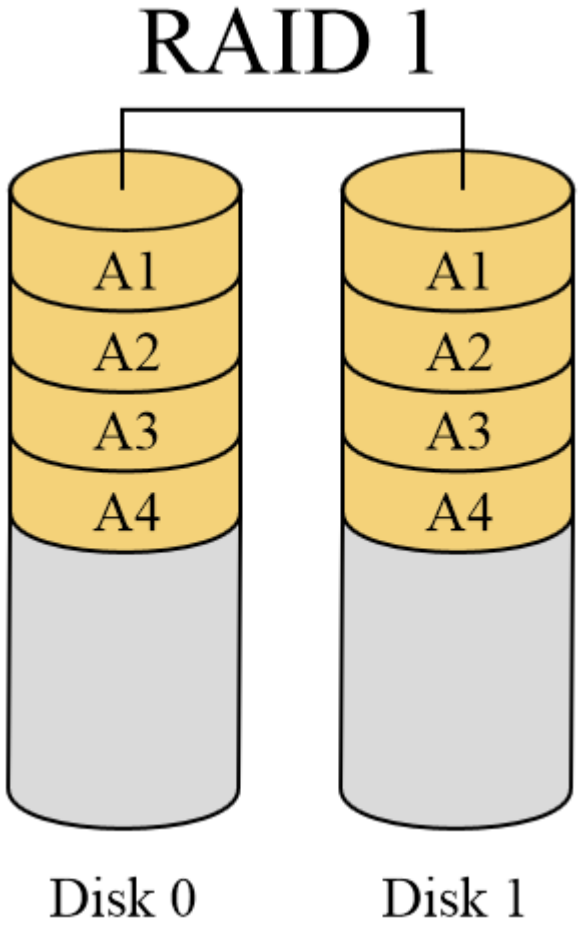
- Die Platten
  - Im Verbund meist gleich groß
  - Idealerweise Typgleich
  - ... Allerdings mit der Gefahr von Serienfehlern
  - Heute eigentlich immer mit (begrenzter) Fehlererkennung, z.B. S.M.A.R.T.
- Der oder die Controller (I)
  - Software RAID
    - Billig, keine Hardware die ausfallen kann
    - Belastet CPU, leistungsfähiger Bus ist separat einzurichten
  - ...

Implementationen – Komponenten eines RAID-Systems (II)

- Der oder die Controller (II)
  - …
  - Hardware RAID
    - Einer pro Platteneinheit, pro Array, oder sogar pro Platte
    - Oft mit eigener CPU und zusätzlichem Cache-Speicher
    - Entlastet die CPU, optimiert den Durchsatz
    - Selber ein Ausfallrisiko
  - Hybrid: Host RAID
    - Vereinigen die Nachteile der beiden anderen, sind aber billig und oft schon „on-board“.
    - Für professionelle Nutzung ungeeignet, für nicht-professionelle fragwürdig.

Implementationen – RAID Level (I)

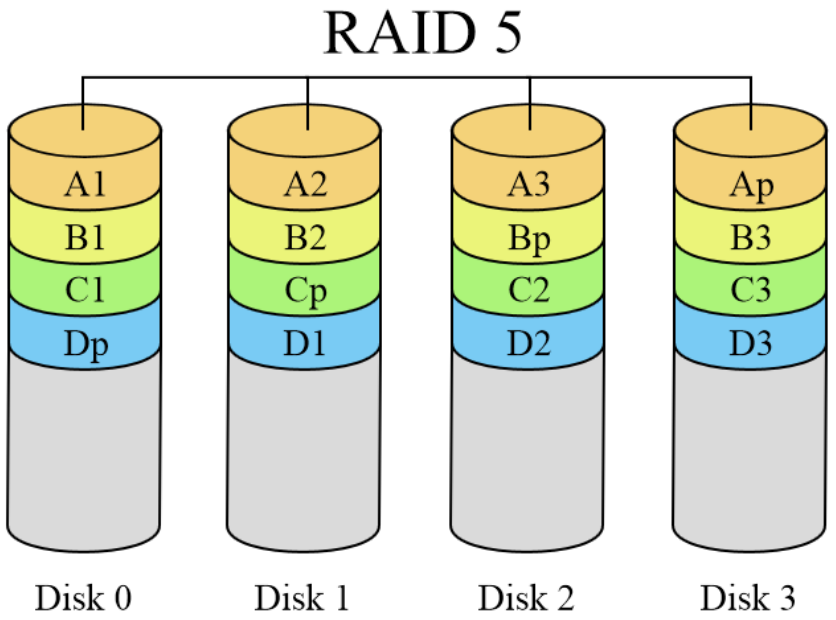
- RAID 1: Datenspiegelung
  - Volle Redundanz, hohe Sicherheit
  - Schneller Ersatz, trivialer „rebuild“
  - Leistungssteigerung möglich
  - Einfach
  - Verdoppelt Preis für Speicherplatz



Quelle: http://de.wikipedia.org/wiki/RAID

Implementationen – RAID Level (II)

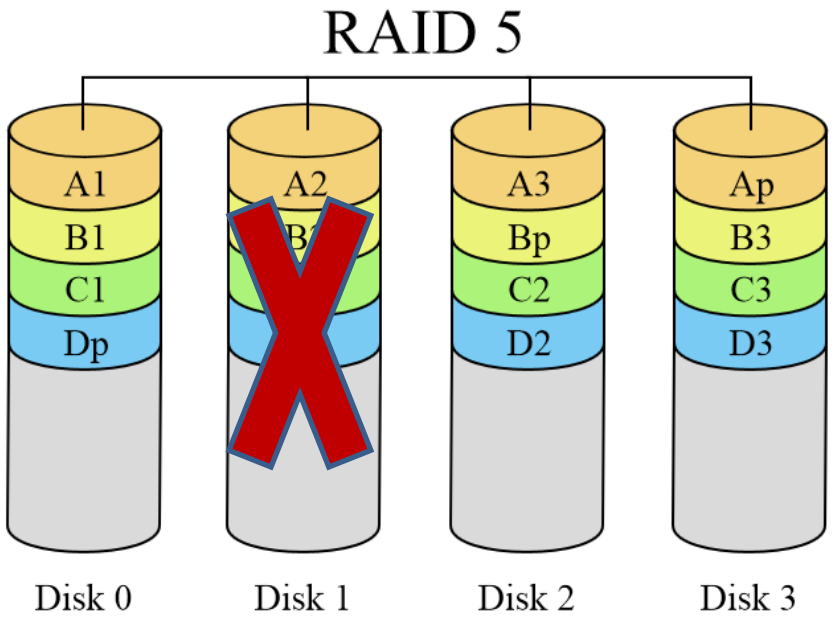
- RAID 5: Block-Level Striping mit verteilter Paritätsinformation
  - Meist drei oder fünf Platten
  - Parität per XOR (oder XOR inkrementell)
  - Billiger: z.B. 4:1 statt 1:1
  - Wiederaufbau nach Fehler erforderlich (im Betrieb)
  - Leistungssteigerung beim Lesen
  - Häufige „kleine“ Schreibvorgänge langsam



Quelle: http://de.wikipedia.org/wiki/RAID

Implementationen – RAID Level (II)

- Wenn‘s kracht:
- Verbund läuft weiter
- Disk 1 ersetzen
  - Hot swap
  - Hot spare
- Rebuild
  - A2 := A1 ⊕ A3 ⊕ Ap
  - B2 := B1 ⊕ B3 ⊕ Bp
  - ...
  - Cp := C1 ⊕ C2 ⊕ C3
  - Kann im Betrieb erfolgen
  - Während Rebuild verwundbar: Keine Redundanz

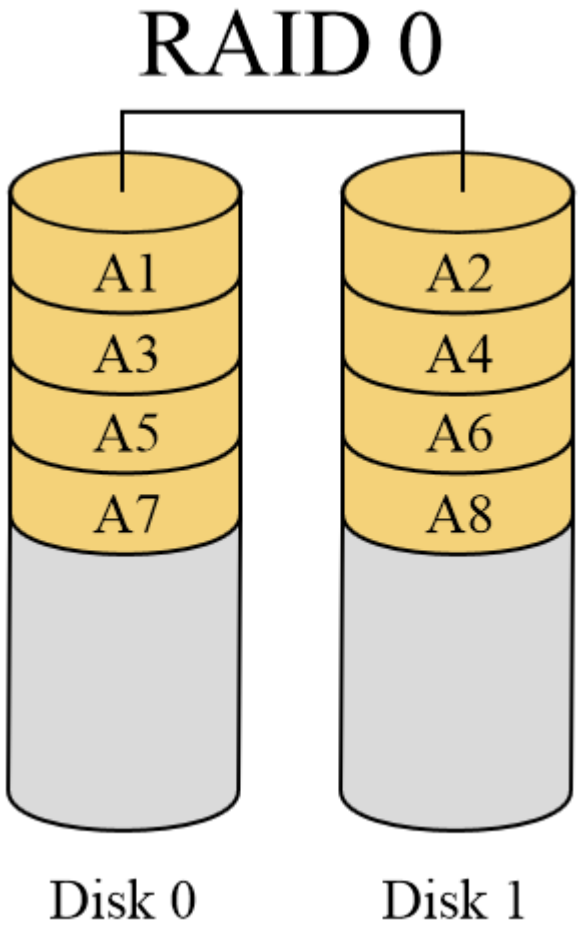


Quelle: http://de.wikipedia.org/wiki/RAID

0	0	0	0
0	1	0	1
1	0	0	1
1	0	1	0

Implementationen – RAID Level (III)

- RAID 0: Striping
  - Uneigentliches RAID:
    Risky Array .../Rapid Array ...
  - Erhöht Verlustwahrscheinlichkeit auf 1 - (1 - p)^n bei n Platten
  - Erhöht Lese- und Schreibgeschwindigkeit, nicht Zugriffsgeschwindigkeit
  - Wird attraktiv als Hybrid in Verbindung mit „echtem“ RAID

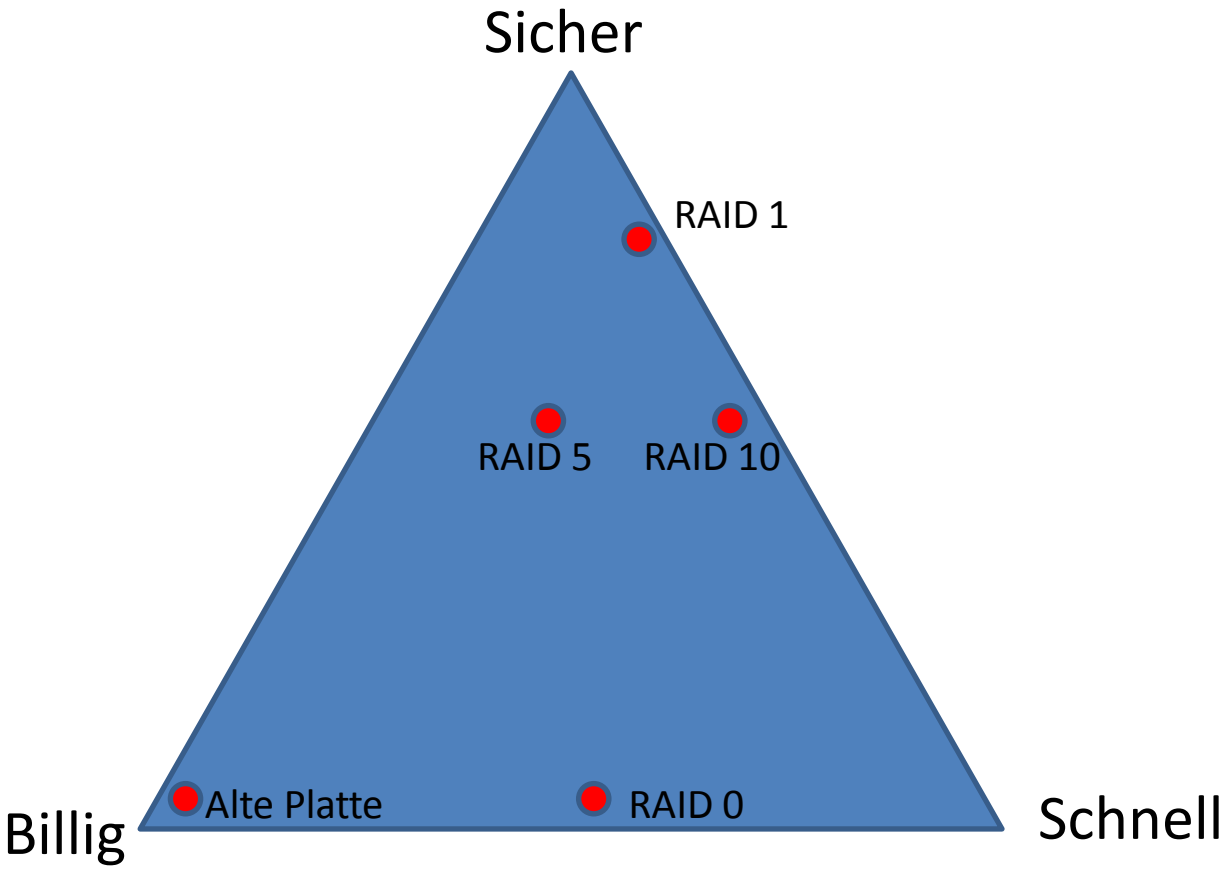


Quelle: http://de.wikipedia.org/wiki/RAID

Implementationen – RAID Level (IV) Der RAID-Level Zoo und seine Tiere

- Die exotischen
  - 2 (Bit-Level Striping), 3 (Byte-Level Striping)
  - 4 (ähnlich 5, aber eigene Platte für Paritätsinformation)
  - 6 (ähnlich 5, etwas langsamer, aber verkraftet Ausfall von zwei Platten wegen doppelter Parität)
- Die Chimären
  - 01 (0 über 1), 10, 05, 50, 15, 51, 100, ...
    Einige davon, wie z.B. 10, durchaus von hoher praktischer Bedeutung, allerdings nicht wichtig für das grundlegende Verständnis von RAID. Details siehe Literatur.
- Die Ableger und Weiterentwicklungen
- Die Uneigentlichen
  - NRAID, JBOD, ... (Festplattenverbunde)

Einschub: Bermuda-Dreieck der Systementscheidung



Implementationen - Praktische Gesichtspunkte

- Wovor RAID nicht schützt
  - Logische Fehler, z.B. Programmfehler
  - Malware, Sabotage, andere Außeneinflüsse
  - Brand, Wasserschaden
  - ... und daher: Fehlende Datensicherung
  - Ausfälle die nicht statistisch/unabhängig sind: Stromausfall, Überhitzung, Fehlerkaskaden ...
  - Controllerfehlern
- Nicht jeder Fehler wird erkannt
  - Unerkannte Fehler können gelegentlich erst beim „Rebuild“ zum Vorschein kommen
  - Fehlererkennung kostet Geschwindigkeit (z.B. „read after write“, Deaktivierung Disk-Caches)
- Unglückliches Timing
  - RAID 5 „Write Hole“: Ausfall nach Schreiben der Daten, aber vor Aktualisierung der Parity-Informationen
  - Inkonsistenz kann oft nicht einmal festgestellt werden

Ausblick

- RAID auf SSDs nur begrenzt anwendbar
- Hauptfehlerquelle nicht statistisch
- Redundanz „verbraucht“ SSDs
- Fehlerkorrektur (ECC) implementiert
- Bus beschränkt Übertragungsgeschwindigkeit

Aber:

- Einfache Zusammenfassung kleiner Platten zu großen logischen Einheiten
- Große SSD schneller als zwei kleine
- Mal schaun was die Zukunft bringt ...

Zusammenfassung

- Effektiver, effizienter, kostengünstiger Schutz vor Datenverlust durch Plattenausfälle und Plattenschäden.
- Maßnahme zur Erhöhung der Verfügbarkeit, nicht Datensicherung.
- Entschärft Flaschenhals zwischen Sekundärspeicher und CPU/Primärspeicher (im Austausch gegen Sicherheit).
- Technische und logische Implementationen abhängig vom Anwendungsumfeld und den spezifischen Anforderungen.
- In der Praxis wird heute wohl kein Serverpark mehr zu finden sein, in dem RAID nicht in der einen oder anderen Form zum Einsatz kommen.

Herzlichen Dank für Eure Aufmerksamkeit!

- **Quellen:**
- „A Case for Redundant Arrays of Independent Disks (RAID)“, David Patterson, Garth A.Gibson, Randy Katz, Computer Sciences Division UC Berkeley, 1987 http://www.eecs.berkeley.edu/Pubs/TechRpts/1987/CSD-87-391.pdf
- http://www.heinlein-support.de/upload/slac08/Heinlein-RAID_Mathematik_fuer_Admins.pdf
- http://www-03.ibm.com/ibm/history/exhibits/storage/storage_3380c.html
- http://referate.mezdata.de/sj2003/festplatte_tobias-allinger/ausarbeitung/geschichte.html
- http://de.kioskea.net/contents/histoire/disque.php3
- http://de.wikipedia.org/wiki/RAID
- http://en.wikipedia.org/wiki/RAID
- http://de.wikipedia.org/wiki/Festplattenlaufwerk
- http://en.wikipedia.org/wiki/Hard_disk_drive
- http://de.wikipedia.org/wiki/Mean_Time_Between_Failures