

# GDB [HA] zum 1. 11. 2013

Tim Dobert, Kai Sonnenwald, Arne Struck

31. Oktober 2013

1. a) Informationssysteme stellen einen so genannten MAT-System (Mensch/Aufgabe/Technik) dar. Sie sind zentrale Bestandteile in rechnergestützten Prozessen. Informationssysteme müssen Daten schreiben, manipulieren/modifizieren/nutzbar machen und löschen können.
- b)
  - Logische Datenunabhängigkeit bezieht sich auf die strukturellen Änderungen einer Datenbank. Eine solche Änderung würde also nichts an der grundlegenden Funktionsweise der Datenbank ändern.
  - Physische Datenunabhängigkeit bezieht sich auf eine physische Umstrukturierung, bei einer solchen hat der Eingriff keine Auswirkung auf die Organisation und Wirkungsweise der Datenbank.
- c)
  - Online Spiele müssen Accounts und zugehörige Attribute wie beispielsweise Spielerstatistiken, Ausrüstung, Kämpfe etc. verwalten.
  - Parteien/Vereine brauchen Informationssysteme für die Mitgliederverwaltung. Es müssen beispielsweise Mitgliederlisten mit den dazugehörigen Informationen, Beitragszahlungen etc. verwaltet werden.
  - Die Ergebnisse der NSA-Spionage, die ihre Ergebnisse einzelnen Personen/Profilen zuordnen muss. Beispielsweise: Bewegungs-/Aufenthaltsdaten, Kommunikationsabfragen.

2. a)

Objekttyp	Eigenschaft
User	UserID, UserName
Gemeinschaft	GemeinschaftsID, List: UserId
Mannschaft	Mannschaftsname
Begegnung	BegegnungsID, Ergebnis, List(2): Mannschaft
Tipp	TippID, UserID, BegegnungsID, StandTipp
Wettbewerb	WettbewerbsID, GemeinschaftsID, List: BegegnungsID, List: TippID
PunkteBoard	WettbewerbsID, UserID, Punkte

- b) Bei der Anmeldung eines neuen Nutzers, wird der Tabelle **User** ein neuer Eintrag hinzugefügt. Das Erstellen einer Gemeinschaft führt zu einem neuen Eintrag in der **Gemeinschaft**-Tabelle. Die dazu hinzugefügten Benutzer werden in der UserID-Liste gespeichert. Auf die Tabellen **Mannschaft** und **Begegnung** haben die Nutzer keinen Zugriff, sie werden von der Verwaltung aktualisiert. Das Hinzufügen eines Wettbewerbs zu einer Gemeinschaft, führt zu einem neuen Eintrag in der **Wettbewerb**-Tabelle mit ihrer ID. Dabei können vom Gemeinschaftsgründer neue Elemente in die Begegnungsliste und von Nutzern neue in die Tipp-Liste eingefügt werden.

Vergleichend mit den Anforderungen aus dem Skript halten wir folgende Eigenschaften für besonders wichtig in unserer Datenbank:

i. **Robustheit(Persistenz, Atomarität):**

- Durchgeführte Aktionen sollen nicht verloren gehen.
- Änderungen werden entweder ganz oder gar nicht ausgeführt, um zu gewährleisten, dass keine beschädigten Datensätze entstehen.

ii. **Nutzerfreundlichkeit:**

- Nutzbarkeit ohne Kenntnisse über Datenbanken muss als Spieler nutzbar sein.
- Die Programmschnittstellen (für Entwickler) sollten leicht und verständlich sein.

iii. **Portierbarkeit:**

- Da heutzutage von vielen Geräten und Systemen auf Funktionen zugegriffen wird, sollte sie portierbar sein.
- Unterschiedliche Anwendungsspezifikationen führen zu unterschiedlichen Zugriffsanwendungen (bspw.: Desktopanwendung für Verwaltende Ebene, Webanwendung für die User).

iv. **Isolationseigenschaften:**

- Mehrbenutzerbetrieb muss ohne Konflikte machbar sein.
- Leistung muss auch bei Belastung stabil bleiben (viele Spieler könnten zu ähnlichen Zeiten zugreifen).

v. **Kontrolle der operationalen Daten durch das System:**

- Verhinderung der Abgabe mehrerer Tipps einer Person auf das selbe Spiel.
- Verteilung unterschiedlicher Rechte an unterschiedliche Nutzergruppen (mehr Rechte für die Verwalter).

### 3.

#### **Dateisystem:**

A i. Informationen noch nicht geschrieben:

Die Änderung im Arbeitsspeicher hatte noch keinen Einfluss auf die Daten auf der Platte.

ii. Informationen bereits geschrieben (teilweise):

Wenn jedes Konto als eigene Datei gespeichert ist, wird Konto 5 der Betrag abge-

zogen, aber nicht auf Konto 7 addiert. das Geld ist also verloren.

- B
  - i. Informationen noch nicht geschrieben:  
Es wird ein Auszug für Konto 7 gedruckt, auf dem steht, dass eine Überweisung stattgefunden hat. Die Beträge haben sich aber nicht geändert.
  - ii. Informationen bereits geschrieben (teilweise):  
Die Überweisung hat richtig stattgefunden, es wird aber nur ein Auszug für Konto 7 gedruckt.

#### **Datenbanksystem:**

- A
  - i. Informationen noch nicht geschrieben:  
Die Änderung im Arbeitsspeicher hatte noch keinen Einfluss auf die Daten auf der Platte.
  - ii. Informationen bereits geschrieben (teilweise):  
Das System würde die Subtraktion rückgängig machen um zu verhindern, das Geld verloren geht.
- B
  - i. Informationen noch nicht geschrieben:  
Die Überweisung findet nicht statt, da nichts auf die Platte geschrieben wurde.
  - ii. Informationen bereits geschrieben (teilweise):  
Je nachdem ob das Drucken des Auszuges mit zu der atomaren Transaktion einer Überweisung gehört, wird sie entweder rückgängig gemacht oder es gibt keinen Ausdruck für Konto 5.

4. a)  
Die erste Anfrage erstellt eine Tabelle namens `gdb_gruppe041.user` mit den Spalten `id`, `name` und `passwort`. Die zweite Anfrage fügt dieser Tabelle eine Zeile hinzu. Dabei ist die `id=1`, der Name `gdbNutzer` und das `passwort` „geheim“.
- b)  
Die erste Zeile wählt aus allen Datenbankeinträgen in der Tabelle `gdb_gruppe041.user` diejenigen aus, bei denen `name` `gdbNutzer` ist. Es wird dann der Eintrag geliefert, der in a hinzugefügt wurde.
- c)  
Die Architektur von MySQL folgt der Drei-Schema Architektur. Sie besitzt ebenfalls eine externe, konzeptuelle und interne Ebene. Die Dokumentation ist noch um einiges detaillierter und zeigt an, welche Komponenten zu diesen drei Ebenen gehören.