Prüfungsfragen Datenbanken Jablonski 2000

Mehrrechner Datenbanksysteme
Prof. S. Jablonski
Januar 2000

Bemerkungen zu Prüfung und Prüfer

- Benoteter Schein

Fragen

- Klassifikation von MR-DBS
- Fragmentierung (speziell Abgeleitete Fragmentierung)
- Replikation in parallelen DBS
- Synchronisierung in Shared-Disk-DBS
- Fragmentierung in parallelen DBS

Konz. Entwurf v. Datenbanken, Kaufmännische DB, Datenbankanwendungspraktikum
Professor Jablonski
Mai 2000

Bemerkungen zu Prüfung und Prüfer

- Ergebnis: 1,0
- Sehr freundliche Atmosphäre.
- Jablonski sehr freundlich und nett.
- Ich hätte bei der einen oder anderen Frage durchaus noch mehr sagen können, aber wenn er gemerkt hat, dass ich Bescheid weiß, war er auch mit der Hälfte zufrieden.
- Die Fragen, die er stellte, waren eigentlich sehr direkt und klar. Ich wußte immer gleich, worauf er hinaus wollte.
- Jablonski hat schon ein paar klare Sachverhalte gefordert, aber er hat keinen großen Wert auf 100% gelegt, sondern war meines Erachtens schon zufrieden, wenn man das Ganze schön verständlich und einfach erklärt.
- Wie ist der Schwierigkeitsgrad? Einfach bis mittel, aber ich war auch relativ gut vorbereitet (auf Konz. Entwurf)
- Gehen Fragen über den Vorlesungsstoff hinaus? Nein
- Tipps für künftige Prüflinge: Konz. Entwurf gut beherrschen
- Fazit: Ich kann Jablonski nur empfehlen, da er sehr freundlich ist und keine fiesen Fragen stellt.

Fragen

- Konzeptioneller Entwurf:
  - ER-Diagramm beschreiben und hinmalen (ich habe das klassische Beispiel Auftrag, Lieferung und Lieferant genommen, das hat ganz gut gepasst, weil er später auch noch viel darauf eingegangen ist), Beziehungstypen grob erklären.
  - Aus dem ER-Diagramm Relationen erstellen (Primär- und Fremdschlüssel erklären und hinmalen). Dabei auch erklären,welche Möglichkeiten es gibt, 1:n-Beziehungen unterschiedlich darzustellen (mandatory oder nicht mandatory: entweder mit Realtionship-Relation oder ohne)
  - Normalisierung 1-3 und Boyce-Codd erklären anhand der gewählten Relationen
  - Müssen Relationen in einem Konzeptionellen Schema immer normalisiert sein? Antwort: Nein, z.B. bei Star Schema in einem Data Warehouse wegen Performance oder bei Anwendungen, die nicht ändern, löschen oder einfügen (vgl. Anomalien!) müssen.
  - Ich musste ein SQL-Statement auf der gewählten Beispielrelation erzeugen: Die Namen der Artikel aus Lieferungen mit mindestens 10 Artikeln
  - Dann zeigte er mir ein SQL-Statement: ich musste es umgangssprachlich formulieren.

 - Beschreiben der objektrelationalen Datenbanken: Antwort: Versucht, Nachteile (Armes Typenkonzept, Schwach bei komplexen Sachverhalten etc.) der relationalen und objektorientierten (Nebenläufigkeit, Mehrbenutzerbetrieb, Performance, Transaktionsverarbeitung etc.) Modelle zu vermeiden und die Vorteile zu vereinen etc.
 - Beschreibung von Triggern: Antwort: ECA (Event-Condition-Action) Prinzip erklärt. Before/After Update etc. Aufrufen von Stored Procedures etc.

- Kaufmännische Datenbanken
  - Unterschied zwischen Grund- und Sonderrechnung: Antwort: Grundrechnung für operativen Betrieb (mit Ändern, Löschen, Einfügen, eher auf Sachbearbeiterebene), Sonderrechnung für Auswertungen (kein Ändern etc.), typischerweise z.B. Jahresabschluss, GuV etc., eher auf Managementebene)
  - Was bedeuten die Flexfelder bei Oracle: Antwort: Ich sagte, dass ich mir nicht sicher bin, ob ich das so richtig verstanden habe, erklärte ihm dann, dass die Flexfelder im Prinzip nur Metarelationen darstellen (dabei lachte er und sagte, dass er das auch nicht anders verstanden hat). Dann ging ich noch auf die unterschiedlichen Ansätze von ORACLE und SAP ein. SAP: Ausblenden von Feldern. ORACLE: Verwenden der Metarelationen zur Definition des eigenen Schemas
  - In Bezug auf das ANSI-SPARC-Modell: Wo stehen Grund- (beim konzeptionellen Schema, Ändern etc. möglich) und Sonderrechnung (bei den externen Schemata, durch Views realisiert, i.d.R. nur Leseoperationen möglich)

Architektur von DB, Workflow-Management, Konzeptioneller Entwurf (DB1, die Grundvorlesung halt), und Data Warehouse
Prüfer: Prof. Stefan Jablonski, Beisitzer: Wolfgang Hümmer
Mai 2000

Bemerkungen zu Prüfung und Prüfer

- Es ist besonders gut, wenn man sich im Vorraus ein paar Beispiele ausdenkt, ein paar Relationen mit den verschiedenen Kardinalitäten, ein paar Workflows...
- Alles in allem ist die Prüfung gut gelaufen. Der Prüfer gibt einem ein Thema vor. Dann am besten schön runterbeten, was einem dazu einfällt. Am Besten immer so lange reden, bis er einen unterbricht und auf ein anderes Thema zusteuert.
- Auch Jablonski und sein Beisitzer haben ein ausgezeichnetes Pokerface, also man kann an den Gesichtsausdrüken nicht ablesen, ob die Antwort gerade falsch oder richtig war.
- Wenn man also schön aus dem Nähkästchen plaudert, ein paar selbstgestrickte Beispiele dabei hat und ein paar Bilderchen malen kann, hat man eigentlich nichts zu befürchten.

Fragen

- DB1:
  - E/R-Diagramme erklären mit Beispielrelation
  - E/R-Diagramme verschiedener Kardinalitäten auf Relationen abbilden (1:1, 1:n, m:n)
  - Normalisierung: Vorteile / Nachteile
  - Wann sollte man normalisieren, und wann nicht (Kommt drauf an, ob mehr geschrieben oder mehr gelesen wird)
  - Normalformen für Relationen (Erste, Zweite, Dritte, Boyce-Cod)
- Architektur:

- Schichtenmodell gesamt im Schnelldurchlauf
- Genauere Betrachtung von Schicht 2, 3 und 4 (Nummerierung von unten, also z.B. 2 ist Systempuffer)
- Satzzuordnungskonzepte (TID und DBTT/PPP)

• Data Warehouse:

- Warum Warehouse (Weil normales Abspeichern zu langsam wäre für statistische Auswertungen)
- Vergleich MOLAP/OLAP
- Nenne Multidimensionale Datenmodelle (z.B. Gridfile)

• Workflow:

- Aspekte bei der Modellierung von Workflows
- Genauere Betrachtung des funktionalen Aspekts:
- Möglichkeiten zur Strukturierung eines Workflows (Aggregation, Generalisierung, Faltung...)

Konzeptioneller Entwurf, Architektur, DataWarehouse, DW-Übung Prof. Jablonski Oktober 2000

Bemerkungen zu Prüfung und Prüfer

- Läßt einen lange erzählen, ohne zu unterbrechen, solange man etwas Wichtiges zu sagen hat (→ immer Reden, solange man richtige Sachen weiß, die auch nur entfernt mit der letzten Frage zu tun hatten)
- Eher Verständniswissen gefragt, Details sehr unwichtig, Überblick über gesamten Stoff sollte vorhanden sein, Zusammenhänge erklären
- Einzelne Themen waren in etwa gleichberechtigt, aber bei KontEntw sollte man schon sehr sicher sein!
- Prüfer macht zu Beginn Witze, ist eher passiv, gibt sich während der Prüfung immer ein wenig desinteressiert

Fragen

• Konzeptioneller Entwurf

- Relationales Modell erklären (Relation / Beziehungen / ref. Integrität / math. Modell das zugrundeliegt / usw., ist ja wohl echt keine Überraschung!)
- CODASYL-Modell (Erklären / Realisation n:m Beziehung, Personal-Hierarchie)
- ERD (Erklären Entity, Relationship / Abb. von Diagramm auf rel. Modell)
- OODBMS (Erklären / Warum sinnvoll / Klasse, Vererbung, Implementierung von Schnittstellen / Probleme ausführlich darlegen, Überleitung zu nächstem Punkt!)
- ORDBMS (Erklären / Vorteil gegenüber OODBMS (Wichtigster: Man kann SQL verwenden!) / Bsp. geben, was die Hersteller gemeinhin anbieten (Arrays, ...))

• DataWarehouse + Übung

- Multidimensionales Modell (Definition / Unterschiede zu rel. Modell ausführlich)
- Realisierung MDM (2 grundsätzliche Wege / mult.-dim. Implementierung, Vorteile, Nachteile, wann gut nutzbar? / rel. Implementierung, Vorteile, Nachteile, wann gut nutzbar?)

• Architektur

- Mehrdimensionale Indexstrukturen (Erklären was das ist, wofür gebraucht (siehe auch vorhergehender Punkt) / die in der Vorlesung vorgestellten Möglichkeiten (mehrere eindim. Indices, lineare Einbettung, Dimensionsverfeinerung, divide&conquer, räumliche Objekte) jeweils erklären und Beispiel bringen)

Conceptual Design of Database Systems [CDDB] Data Management Systems [DMS] Commercial Database Systems [ComDB] Tutorial on Database Applications [DBA]
Prof. Jablonski
April 1999

Comments on the examination and the examiner

- DETAIL is important for him.

Questions

- About Supplier and Parts. First draw a E-R Diagram, then map it to tables(normalized), then write SQL statement to display the supplier's name who supply the Part 'XX', at last rewrite the SQL using join.
- ACID
- different kind of failure of DB system and the solution.(archive, checkpoint,log). Different recovery method.(R1 R5). Logging, BOT,AOT, how and when to use.
- Tuple ID.
- B-Tree and B*Tree. (Main idea and difference, and advantage)
- Basic Accounting and evaluation, Difference,which level in ANSI/SPARC
- Different customizing process of SAP and Oracle.

Mehrrechner Datenbanksysteme
Prof. S. Jablonski
Januar 2000

Bemerkungen zu Prüfung und Prüfer

- Benoteter Schein

Fragen

- Klassifikation von MR-DBS
- Fragmentierung (speziell Abgeleitete Fragmentierung)
- Replikation in parallelen DBS
- Synchronisierung in Shared-Disk-DBS
- Fragmentierung in parallelen DBS