Datenbanksysteme WS 2016/17
Prof. Dr.-Ing. Sebastian Michel
BSc. Peter Brucker
TU Kaiserslautern, FB Informatik – Lehrgebiet Informationssysteme
http://dbis.informatik.uni-kl.de
Extrablatt Klausurvorbereitung: Ausgabe 31.01.2017, Abgabe 06.02.2017

Dies und Das

Ergänzend zu den bereits gestellten "Dies und Das" Fragen (vgl. Blatt 2 und 5), finden Sie hier zur Klausurvorbereitung weitere Fragen, die Sie vom Stil her auch in der Klausur erwarten können. Natürlich ist diese Liste nicht vollständig. Solche Fragen machen auch nur einen Teil der Klausur aus (vgl. Altklausur). Daneben sei noch angemerkt, dass selbstverständlich der komplette Stoff aus Vorlesung und Übung klaussurrelevant ist, sofern nicht explizit herausgenommen.

(1) Finden Sie ein Beispiel, das zeigt, dass VSR nicht monoton ist.

(2) Ist der Threshold Algorithmus nie schlechter als Fagins Algorithmus? Welches Maß für "besser" bzw. "schlechter" benutzen Sie?

(3) Beschreiben Sie was Aktionskonsistenz bzw. Transaktionskonsistenz bedeutet und wo dies für welche Checkpoint-Verfahren betrachtet wird.

(4) Beschreiben Sie ein Deadlock-Vermeidungsverfahren, bei dem jeder Transaktion eine Priorität $\in \mathbb{N}$ zugeordnet ist und indem bei der Vermeidung eines Deadlocks die Summe der Prioritäten der neugestarteten TA minimiert werden soll.

(5) Wo liegt der Unterschied zwischen SS2PL, S2PL und 2PL? Welcher Scheduler ist restriktiver?

(6) Wieso muss man bei unscharfen Sicherungspunkten beim Redo zu einem Zeitpunkt vor dem eigentlichen Sicherungspunkt beginnen?

(7) Kann man bei einem Quadtree die Himmelsrichtungen N, S, E und W hinzufügen, um einen höheren Fanout zu bekommen?

(8) Warum wird in der Regel aus Performancegründen die Konfiguration "steal und ¬force" betrachtet?

(9) Ist ACA unbedingt erforderlich für ACID?

(10) Wieso ist es wichtig, dass vor einem Commit alle Informationen, die notwendig sind, um die TA nachzuvollziehen, auf persistenten Speicher geschrieben werden? Welchen "Buchstaben" aus ACID betrifft dies?

(11) Ist RC unbedingt erforderlich für ACID? Und falls ja, um welchen Teil von ACID geht es dabei?

(12) Beschreiben Sie die beiden Phasen von 2PL und geben Sie einen Schedule an, der nicht in Gen(2PL) liegt.

(13) Wie verhalten Sich statische Hashverfahren, wenn die Seitengröße beliebig groß oder klein wird?

(14) Beschreiben Sie, wie innere Knoten eines R-Baums aufgebaut sind.

(15) Wieso ist der 1-Bucket-Random Algorithmus korrekt und was bedeutet dies im Sinne von Theta-Joins?

(16) Beschreiben Sie die Idee hinter Min-Hashing und begründen Sie warum die Kollissionswahrscheinlichkeit dem Jaccard-Koeffizienten der beteiligten Mengen entspricht.

(17) Wieso ist beim TPUT-Algorithmus min_k/m eine korrekte Grenze für das sequenzielle lesen?

(18) Beschreiben Sie eine Kaskade von Map-Reduce-Jobs, die als Eingabe einen ungerichteten Graphen als Adjazenzliste nimmt, und als Ausgabe die Anzahle der Dreiecke in diesem Graphen (das sind drei Knoten, die alle miteinander verbunden sind) ausgibt.

(19) Welches Layout ist für OLTP (in der Regel) besser geeignet, Row- oder Column-Store?

(20) Wie unterscheiden sich die folgenden Berechnungen für den Belegungsfaktor? 1. |Gespeicherte Datensätze in Primärdatei|/(|Bucketkapazität| · |Buckets in Primärdatei|) und 2. |Gespeicherte Datensätze gesamt|/(|Bucketkapazität| · |Buckets gesamt|)

(21) Warum sind Row-Stores nicht (gut) geeignet, um Kompression anzuwenden?

(22) Wieso ist es wünschenswert, eine monotone Serialisierbarkeitsklasse zu benutzen?

(23) Was sind CLRs und warum braucht man diese?

(24) Geben Sie Map- und Reduce-Funktionen an, um einen invertierten Index aufzubauen.

(25) Was bedeuten OLTP und OLAP und wo liegen die Unterschiede?

(26) Ist es beim Reduce-Side-Join im Reduce-Task notwendig die Join Bedingung zu überprüfen?

(27) Welche Art von Index können Sie benutzen, um k-Skyband-Anfragen effizient zu beantworten?

(28) Wieso ist der Replicated-Join ein Map-Only-Join?

(29) Was ist der "Best-Case" für einen Performance-Vorteil von Column-Store vs. Row-Store und was ist der "Worst-Case"?

(30) Geben Sie eine Möglichkeit an, wie Daten aus einem höherdimensionalen Raum auf einen 1-dimensionalen Raum abgebildet werden können.

(31) Wie würden Sie die Pivot-Objekte für den GH-Tree auswählen?

(32) Beschreiben Sie eine Möglichkeit, den kd-Baum so zu erweitern, dass mehrere Datenpunkte pro Knoten abgespeichert werden können. Erklären Sie, wie dann eingefügt, gesucht und gelöscht wird.

(33) Was ist das "Problem" mit der intensionalen Auswertung?

(34) Wie können Sie durch den Reduce-Side-Join einen Full-Outer-Join berechnen?

(35) Was bedeuten DSM und NSM?

(36) Wieso sind Redo und Undo-Phase idempotent und was bedeutet dies überhaupt?

(37) Was ist der Unterschied zwischen Possible-Tuple und Possible-Answer-Set Semantik?

(38) Wie beurteilen Sie ein Verfahren, dass ähnlich dem R-Baum ist, aber nicht mit MB-Rechtecken sondern mit MB-Kreisen arbeitet?

(39) Beschreiben Sie, wie sich das Vorhandensein von sekundären Indexen auf die Transaktionsverwaltung und die möglichen Anfragen auswirkt.

(40) Widerlegen oder begründen Sie folgende Aussage. Im 1-Bucket-Random Algorithmus wird Tupel $s \in S$ und Tupel $t \in T$ auf keinen Fall zwei mal gejoined.

(41) Was bedeutet Recoverable? Geben Sie eine Historie an, die in RC liegt und eine, die nicht in RC liegt (aber in CSR).

(42) Wieso gibt es beim R-Baum den Parameter m, mit dem man fordert, dass jeder Knoten mindestens $m \leq \lceil M/2 \rceil$ Einträge hat?

(43) Was ist das #SAT Problem? Wo ist der Unterschied zu SAT?

(44) Beschreiben Sie ob Undo bzw. Redo nötig sind bei der "¬steal und ¬force".

(45) Was sind sichere Anfragen im Sinne probabilistischen Datenbanken?

(46) Können durch den Reduce-Side-Join Joins mit Prädikaten der Form $\theta(s,t) = |s.A - t.A| < 5$ berechnet werden?

(47) Kann man die gleichen Objekte, die man in einen R-Baum einfügt, auch in einen kd-Baum einfügen?

(48) Wie entsteht aus einem Konfliktschrittegraph ein Serialisierbarkeitsgraph?

(49) Beschreiben Sie Unterschiede zwischen transaktionskonsistentem und unscharfem Checkpointverfahren bzw. Performance zum Anlegezeitpunkt bzw. des Wiederanlaufs.

(50) Was ist Late-Materialization?

(51) Geben Sie für die beiden Priorisierungen MINDIST und MINMAXDIST je einen Fall an, in der die eine Strategie der anderen deutlich überlegen ist. Können Sie eine Faustregel angeben, wann man welche Strategie benutzen soll?

(52) Wann ist eine Aggregationsfunktion (im Kontext von Top-K Algorithmen) monoton?

(53) Beschreiben Sie, wie innere Knoten eines M-Baums aufgebaut sind.

(54) Können Sie auf LSNs verzichten, solange Sie die Transaktionsnummern in den Logeinträgen speichern?

(55) Beschreiben Sie wie im Timestamp-Ordering ein Konflikt erkannt wird.

(56) Beschreiben Sie, wie Sie Einfügen und Löschen für den GH-Tree implementieren würden.

(57) Beschreiben Sie Vor- und Nachteile von Kompression in Column-Stores?

(58) Sie haben zweidimensionale Datenpunkte in einer Datenbank gespeichert, in der Sie keine Nächste-Nachbar-Suche durchführen können. Dafür können Sie Anfragen vom Typ “Gib mir alle Punkte in folgendem Rechteck” ausführen. Wie können Sie auf dieser Datenbank trotzdem effizient die Skyline berechnen?

(59) Was passiert beim zufälligen Sondieren, wenn keine freie Seite gefunden wird? Wie wird bei einer Abfrage der korrekte Datensatz gefunden?

(60) Wie können Sie Skyline-Suchen mit gemischter Präferenz nach sowohl min als auch max nur unter Verwendung von min berechnen?

(61) Wie können Sie durch den Replicated-Join einen Semi-Join berechnen?

(62) Können im Replicated-Join Theta-Joins (d.h. Joins mit beliebigen Prädikanten) berechnet werden?

(63) Vergleichen Sie Quadtree, kd-Tree und Z-Kurve in Hinblick auf zweidimensionale Bereichssuchen (also “Welche Objekte liegen in diesem Rechteck?”).

(64) Welchen Konflikt umgeht Thomas' Write Rule?

(65) Warum ist die extensionale Auswertung überhaupt betrachtet worden und wo liegen deren Probleme?

(66) Geben Sie ein Beispiel an, was passieren kann, wenn das DBMS kein WAL benutzt bzw. die Commit-Regel nicht befolgt.

(67) Beschreiben Sie den Unterschied zwischen wound-wait und wait-die für eine Transaktion t_i die mit Transaktion t_j in Konflikt gerät und t_i vor t_j gestartet wurde.

(68) Im R-Baum müssen die MBRs aller Kindknoten komplett im MBR des Elternknotens liegen. Könnte man diese Forderung auch abschwächen und sagen, die Kind-MBRs müssen das Eltern-MBR nur schneiden?

(69) Geben Sie ein Beispiel einer BID-Datenbank an. Wofür steht BID?

(70) Erweitern Sie das in der Vorlesung vorgestellte Zeitstempelverfahren um mehrere Versionen eines Datensatzes. Dabei werden Leseoperationen erweitert, sodass mit $r(x,t)$ angegeben wird, dass das Objekt x im Zustand vom Zeitpunkt t gelesen wird. Wie sieht ein Schreibzugriff aus? Wie viele Versionen eines Objektes müssen gespeichert sein? Wird so noch ACID garantiert?

(71) Wieso gibt es verschiedene Isolationsstufen, wenn doch durch I in ACID eigentlich volle Serialisierbarkeit verlangt wird?

(72) Zeigen Sie die Echtheit der Inklusion der vorgestellten Serialisierbarkeitsklassen, CSR, VSR, FSR.

(73) Können TI-Datenbanken in PC-Datenbanken ausgedrückt werden, falls ja wieso?

(74) Wie ist die worst-case Komplexität beim Suchen in Quadtree und PR-Quadtree?

(75) Geben Sie einen Schedule an, der in RC aber nicht in ACA liegt und beschreiben Sie was ACA bedeutet.

(76) Was ist für leseintensive Workloads bei Linarem Hashing besser: kontrollierte Splitting mit niedrigem β_s, mit hohem β_s oder unkontrolliertes Splitting?

(77) Geben Sie einen Schedule an, in dem die Lost-Update Anomalie vorliegt.

(78) Was ist die Idee hinter PAX?

(79) Geben Sie ein Beispiel mit zwei Relationen an mit einem korrekten und einem inkorrekten Plan bzgl. extensionaler Auswertung. Wo liegt das Problem?

(80) Was bedeutet OCSR und wo liegt der Unterschied zur Klasse CSR?

(81) Geben Sie für die drei Pruning-Strategien für die Nächste-Nachbar-Suche im R-Baum jeweils einen Fall an, in dem die Strategie sehr nützlich, und einen, in dem die Strategie nicht nützlich ist.

(82) Um Ergebnisse zu berechnen müssen manchmal mehrere Map-Reduce-Jobs hintereinander ausgeführt werden. Die Ausgabe des Reducers ist dabei wieder eine Eingabe für einen Mapper.

(83) Beschreiben Sie anhand der Illustrationen im Script genau, wie die Suche nach einem Schlüssel bei Erweiterbarem Hashing funktioniert.

(84) Wieso vermeidet wound-wait bzw. wait-die, dass der Wartegraph Zyklen enthält?

(85) Welches Logging-Verfahren würden Sie benutzen, wenn a) Sie unendlich Speicher zur verfügung haben, aber das System so instabil ist, dass häufig Recovery gemacht werden muss, oder b) wenn Recovery sehr sehr selten nötig ist, aber Sie nur langsamen und wenig Permanentspeicher verwenden können.

(86) Wann eignet sich der Replicated-Join, und wann nicht?

(87) Vergleichen Sie die Hashverfahren mit Cachingverfahren, die Sie aus Rechnersysteme o.ä. Vorlesungen kennen. Was sind Gemeinsamkeiten und Unterschiede der Funktionsweisen und Anforderungen?

(88) Was geht beim FA Algorithmus schief, wenn die Aggregationsfunktion nicht monoton ist?

(89) Warum muss man, bei der in der Vorlesung hauptsächlich betrachteten Konfiguration, auch Verlierer-Transaktionen in der REDO-Phase nachvollziehen? Und wann muss man dies nicht?

(90) Was bedeutet MinDirtyPageLSN und wo kommt dies zum Einsatz?

(91) Während der Ausführung einer Transaktion stürzt das Datenbanksystem ab. Welche der "Buchstaben" aus ACID kommt nun zum Tragen?

(92) Der Log-Ringpuffer fasst 2^{15} Einträge. Wie viele Transaktionen können gleichzeitig im Datenbanksystem ablaufen?