Exam - Preview

Databases and Information Systems

Fabian Panse

panse@informatik.uni-hamburg.de

University of Hamburg

General Information

- Duration: 120 minutes
- 8 exam questions (each with some subquestions)
- 120 points (100 points to achieve 1.0)
- 50% first part (Prof. Dr. Ritter), 50% second part (Dr. Panse)
- Arithmetic problems (discussed here), knowledge questions (not discussed here)
- Questions are described in English
- Answers can be written in English or German
- No calculator!
- Dates:
  - Tuesday, 06.08.2019, 09:30 a.m., ESA A & B
  - Monday, 23.09.2019, 09:30 a.m., HS A, Chemie

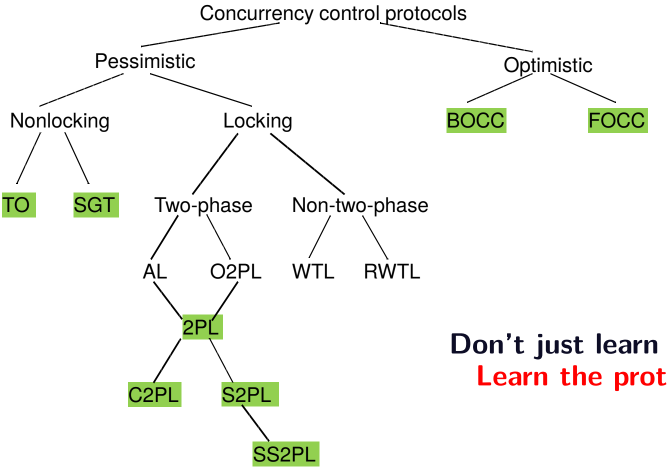
Section 1

Summary: Part I

Concurrency Control: Correctness and Synchronization (1)

- Synchronization Problems (Dirty Read, Lost Update, etc.)
- Isolation Levels (DB2 or ANSI)
- Histories and Schedules
- Serializability Classes (VSR, CSR, OCSR, COCSR)
- Definitions: Conflict and Conflict Equivalence
- Monotonicity, prefix- and commit-closeness
- Conflict- and Wait-For-Graph
- Different Synchronization Protocols (see next slide)
- Multi-granularity Locking (Intention Locks, Compatibility)
- Multi-Version Concurrency Control
- Predicate locks

Concurrency Control: Correctness and Synchronization (2)



Don't just learn the graph.

Learn the protocols!

Logging and Recovery (1)

- Goal of Recovery (most recent TA-consistent DB state)
- Basic Forms of Recovery (Forward, Backward)
- Failure and Recovery Classes
  (TA, Crash, Media and Disaster Recovery)
- Components of a Recovery System
  (Log Buffer, temporary Log File, Archive Log, etc.)
- Logging Techniques (Logical, Physical and Physiological)
  - Advantages and Disadvantages
- Structure of Log-file/record
  - e.g. Update: [LSN, TAID, PageID, Redo, Undo, PrevLSN]

Logging and Recovery (2)

- Insertion-, replacement and propagation-strategies
  - Atomic vs. Non-Atomic
  - Steal vs. No Steal
  - Force vs. No Force
- Commit Procedure
  1. Ensuring Repeatability of TAs
  2. Releasing Locks
- Checkpoints (TOC, TCC, ACC, fuzzy)
- Restart Procedure:
  - Analysis-Phase: Winner- and Loser-TA
  - Redo- and Undo-Phase
  - Compensation Log Record

Distributed Transactions

- Commit Structure (centralized, hierarchical, linear)
- Multi-phase-commit-protocol:
  - Centralized, linear and hierarchical 2PC
  - 1PC, 3PC
- Concurrency Control:
  - Homogeneous and Heterogeneous Federations
  - Deadlock Detection (centralized and decentralized)

NoSQL (1)

- Motivation (4Vs, Impedance Mismatch)
- Characteristics
  - Non-relational
  - Open-Source
  - Schema-less (schema-free)
  - Optimized for distribution (clusters)
  - Tunable consistency
- Scale-up vs Scale-out
- Sharding (Partitioning Strategies: Hash, Range)
- Replication
  - Consistency Model (synchronous vs. asynchronous)
  - Coordination (Master-Slave, Multi-Master)
- CAP-theorem
- ACID- vs. BASE-principle

NoSQL (2)

- Classes of NoSQL-databases
  - Key-Value Stores (e.g. Redis, Dynamo)
  - Wide Column Stores (e.g. Google BigTable)
  - Document Stores (e.g. Mongo)
  - Graph Databases (e.g. Neo4J)
  - Other including Object-oriented, XML, RDF
- NoSQL Consistency Models (Monotonic Reads etc.)
- Big-Data Processing
  - Batch (Hadoop, MapReduce)
  - Stream
- Dynamo
  - Consistent Hashing
  - Versioning (Vector Clocks) and Consistency (Quorums)
- Redis, Google BigTable, Mongo

Section 2

Summary: Part II

DW: Foundations & Architecture

- Purpose (OLAP on an integrated database)
- Differences between operational databases (OLTP) and Data Warehouses (OLAP)
- Reasons for separate implementation (performance, data structure, etc.)
- Virtual vs. physical data integration
- Six phases of data warehousing
- Components of the reference architecture
- Monitoring and ETL (four extraction strategies etc.)
- Data Marts (dependent vs. independent)
- Meta Data Management
    - What is meta data? Purpose of meta data management
    - Architectures (central, distributed, federated)
- Operational Data Store, Master Data Management

DW: Multi-dimensional Modeling & Querying

- Cube concept (facts, dimensions, cuboids/aggregation grid, dimension hierarchies)
- Cube operations (slice, dice, roll-up, drill-down, drill-across)
- Precalculation of aggregation results
  - Benefit: short response time
  - Drawback: requires much memory and many updates
- MOLAP (multi-dimensional matrices)
- ROLAP (star/snowflake/galaxy schema)
- Comparison MOLAP and ROLAP (fast access vs. compact storage)
- HOLAP (vertical partitioning vs. horizontal partitioning)
- Queries
  - Star Join
  - Grouping (Group By, Cube, RollUp, Grouping sets)

Data Mining

- Six mining phases (problem formulation, model selection, etc.)
- Goals and important aspects of data preparation (no algorithms)
- Four classes of data types (nominal, ordinal, etc.)
- Levensthein distance (including the dynamic programming approach)
- Class distinction capability (no formula)
- Classification
    - Input, goals, general approach (learning, application)
    - Concepts of different learning models (k-NN, decision trees, etc.)
    - Idea and usage of information gain (no formula)
    - Quality measures for binary classification

Data Mining

- Clustering
  - Input, goals, problems
  - K-Means, canopy and hierarchical clustering
  - Distance measures for clusters
- Association Rules
  - Input, goals, problems
  - Downward closure, frequent itemsets, apriori algorithm
  - Generation of rules, support, confidence, lift

UDBs: Foundations

- Incomplete database (set of possible worlds)
- Probabilistic database
  (probability distribution over possible worlds)
- World schema/key? (schema/key of the possible worlds)
- Certain/maybe tuples
- Marginal tuple probabilities
- Different types of tuple dependencies
- Possible worlds semantics
  (separate processing of each possible world)

UDBs: Representation Systems

- Need for compact representation
- Difference between world schema/keys and representation schema/keys
- Semantic correctness of representation systems
- Goals of representation systems
  (compactness, modeling power, representation/query complexity)
- Completeness of representation systems
- Closeness of representation systems
- Five representation systems:
  - Modeling concepts
  - Computation of the possible world representations
  - Computation of the most probable world
  - Differences in modeling power of these systems

UDBs: Query semantics

- Principle of the possible answer semantics (set of tuple-probability pairs)
- Benefit of the possible worlds semantics (compositional)
- Benefit of the possible answers semantics (simple representation form)
- Set of possible query answers (all answer tuples with a probability greater than zero)

UDBs: Intensional Query Evaluation

- Underlying idea (two steps)
  - Computation of possible answers along with lineage
  - Lineage based probability computation
- Data lineage (concept to reconstruct the origin of a tuple)
- Using lineage for probability computation (derivation of answer tuple probabilities based on the possible worlds of the queried database)
- Lineage construction rules (projection, cross product/join)
- Problem of probabilistic inference (general-purpose vs. problem specific)
- 1OF and Shannon Expansion
- Concept of Monte-Carlo simulation (processing a set of sample worlds)
- Using Monte-Carlo simulation to infer probabilities from lineage formulas

UDBs: Extensional QE & Aggregate Queries

- Underlying idea of extensional query evaluation
  - Probability computation integrated in relational operators
- Probability computation rules in the case of exclusion/independence (projection, cross product/join)
- Safe/unsafe query plan, safe/unsafe query
- Benefits and drawbacks of extensional query evaluation
- Evaluation of aggregate queries
  - Three different aggregation semantics (distribution, range, expected value)
  - Factors of computation complexity (aggregation semantics, aggregate function, representation system)

Similarity Search in Multimedia Data

- Motivation, content-based access
- Feature extraction and feature representation (signatures vs. histograms)
- Properties of a metric distance function
- Difference between Bin-by-bin and cross-bin distance functions
- Principles of Earth Mover's Distance (not exact definition)
- Distance-based similarity queries
  - Query types (range, k-NN, ranking)
  - Multi-step query architecture with filter distances
  - Optimal multi-step k-NN query
  - Lower bound distance functions (Minkowski, Ind. Minimization)
- Indexing
  - Inverse triangle inequality
  - Pivot Table, Pivot Space

Section 3

Concurrency Control: Correctness and Synchronization

Concurrency Control: Correctness and Synchronization (1)

- **Provided**: A schedule (and perhaps some context on the applied concurrency control scheme such as FOCC or BOCC)
- **Requested**:
  - Is the schedule in CSR/VSR? Why/why not?
  - Does it produce any anomalies? Which ones?
  - Draw the conflict graph!
  - Draw the wait-for graph!
  - Which transaction does commit, which does not? Briefly justify your answer!

Concurrency Control: Correctness and Synchronization (2)

- **Provided**: SQL statements on a timeline, isolation levels for the different transactions
- **Requested**:
  - Is there a deadlock? Which one?
  - Is there an anomaly? Which one?
  - Could the same have happened, if transaction 2 had isolation level XY?
  - What value does statement XY return?
  - What locks does transaction 1 hold at timestamp 5?

Section 4

Logging and Recovery

Logging and Recovery (1)

- **Provided**:
  - Some database configuration (e.g. no-force/steal/non-atomic)
  - An excerpt from the database log and/or a short story (e.g. about a system crash values that were recovered from reading out the disk)
- **Requested**:
  - What are the winner/loser transactions?
  - At what LSN does analyze/redo/undo phase begin?
  - What value does X have after recovery?

Logging and Recovery (2)

- **Provided**: A database configuration, some statements and a short context story
- **Requested**:
  - Fill in the empty log table!
  - Does the system have to do redo/undo recovery? Why/why not?
  - Could we change the system configuration in such a way that redo/undo recovery becomes unnecessary? How?

Section 5

Distributed Transactions

Distributed Transactions

- **Provided**: A protocol (e.g. 2PC) and a scenario description
- **Requested**:
  - What would happen if server B crashed? Will the coordinator abort or commit?
  - Given XYZ happens, what will server C do?

Section 6

NoSQL

NoSQL (1)

- **Provided**: Some set up, e.g. configuration and overview over a Dynamo instance
- **Requested**:
  - Which Dynamo nodes participate in a write on key X?
  - Given no node has ever been down, which nodes participate in a read of key Y?

NoSQL (2)

- **Provided**: Description of a replication protocol
- **Requested**:
  - Does this protocol ensures Read-Your-Writes, Monotonic Reads, Causal Consistency, etc.?
  - In which way do we need to change this protocol in order to ensure linearizability?

NoSQL (3)

- **Provided**: A number of *n* Replicas and *w* Write Acks
- **Requested**:
  - How many Read Acks do we need to guarantee that a read will include the newest version?

Section 7

Data Warehouse

Data Warehouse (1)

- **Provided**: DW Schema and two data sources (schema and instance)
- **Requested**:
  - What transformations are required before the source data can be loaded into the DW?
  - Name two additional preparation activities which are helpful to increase the data quality of the DW!

Data Warehouse (2)

- **Provided**: Star schema, query expressed in natural language
- **Requested**:
  - Name an SQL query which provides the wanted information!
  - What is the aggregation grid of the cube represented by this schema?

Data Warehouse (3)

- **Provided**: Fact table, query with GROUP BY CUBE/ROLLUP/GROUPING SETS
- **Requested**:
  - Compute the result of the given query!
  - Draw the aggregation grid of this query!
  - What is the difference between an additive and a non-additive fact? Name two examples!
  - When is a fact table called cumulative? and when a snapshot table?
  - Is the given fact table cumulative or a snapshot table?

Data Warehouse (4)

- **Provided**: Multiple Group By clauses
  (with CUBE/ROLLUP/GROUPING SETS)

- **Requested**:
  - Which of these clauses are semantically equivalent?

Data Warehouse (5)

- **Provided**: Specific Group By clause (with CUBE/ROLLUP/GROUPING SETS)
- **Requested**:
  - Write another Group By clause which is equivalent to the given one but uses the XYZ operator!

Data Warehouse (6)

- **Provided**: Aggregation grid of a star schema, SQL query with Group By
- **Requested**:
  - Mark all cuboids within the given grid that are computed by the given SQL query!

Data Warehouse (7)

- **Provided**: Text describing the requirements for a DW
- **Requested**:
  - Construct a relational schema (star, snowflake, galaxy) satisfying the given requirements!
  - What are the dimension & fact tables of this schema?
  - What are the primary keys of the fact tables?
  - Is there any hierarchy between the attributes of one dimension table?

Section 8

Data Mining

Data Mining (1)

- **Provided**: Two string values
- **Requested**:
    - The Levenshtein distance between both strings computed by using dynamic programming! (fill in a matrix)

Data Mining (2)

- **Provided**: Semantical description of a data type
- **Requested**:
  - Decide and justify if it is nominal, ordinal, interval or ratio!

Data Mining (3)

- **Provided**: Decision tree, test data objects
- **Requested**:
  - Classify the test data objects buy using the decision tree!
  - Compute the sets of true/false positives/negatives!
  - Compute the fall-out/miss rate/sensitivity/specificity based on the sets of true/false positives/negatives!

Data Mining (4)

- **Provided**: List of 1-(or 2-)dimensional data objects
- **Requested**:
  - Cluster these objects by using the k-means algorithm! (initial centroids are given)
  - Cluster these objects by using the canopy clustering algorithm! (distance function and thresholds are given)

Data Mining (5)

- **Provided**: List of data objects, distance matrix
- **Requested**:
  - Cluster these objects by using the hierarchical clustering algorithm where method XY should be used to compute distances between two clusters!
  - Draw a dendrogram of this clustering approach!

Data Mining (6)

- **Provided**: List of transactions
- **Requested**:
  - Use the apriori algorithm to compute all frequent itemsets!
  - How many association rules can be derived from itemset XY?
  - List two association rules that can be derived from itemset XY!
  - Compute Support, Confidence and Lift for association rule XY!

Section 9

Uncertain Databases

Uncertain Databases (1)

- **Provided**: Possible worlds representation or possible world space
- **Requested**:
  - What implications (positive or negative) and exclusions exist between these tuples?
  - Can you use a XYZ-database to represent this possible world space? If not, why?
  - Model this possible world space by using a XYZ-database!
  - Evaluate SQL query XYZ on this possible world space by using the possible worlds semantics (or possible answers semantics)!

Uncertain Databases (2)

- **Provided**: XYZ-database
- **Requested**:
  - How many possible worlds are modeled by this database?
  - What is the most probable world of this database?

Uncertain Databases (3)

- **Provided**: XYZ-database, SQL query, Results of several Monte-Carlo iterations
- **Requested**:
  - Compute the lineage formulas of the result tuples!
  - Compute the tuple probabilities which are approximated by the Monte-Carlo Simulation!

Uncertain Databases (4)

- **Provided**: XYZ-database, SQL query
- **Requested**:
  - Compute the lineage formulas of the result tuples!
  - Determine which of these formulas are in 1OF!
  - Expand one formula until each of its exclusive subformulas is in 1OF!

Uncertain Databases (5)

- **Provided**: XYZ-database, Extensional SQL query plan
- **Requested**:
  - Compute the probabilities of the result tuples based on the given query plan!
  - What is a safe plan?
  - Is the given plan safe?

Uncertain Databases (6)

- **Provided**: XYZ-database
- **Requested**:
  - Transform this database into a pc-database so that the world-table is minimal!

Section 10

Similarity Search in Multimedia Data

Similarity Search in Multimedia Data (1)

- **Provided**: Three feature representations, ground distance
- **Requested**:
  - What is the difference between feature histograms and feature signatures?
  - Are the given feature representations histograms or signatures?
  - What transportation flow from the first to the second feature representation has the minimal cost?
  - Which two feature representations have the smallest Earth Mover's Distance?
  - What is the Independent Minimization Lower Bound of two feature representations?

Similarity Search in Multimedia Data (2)

- **Provided**: Pivot Table, query object, range query
- **Requested**:
  - Compute the Euclidean/Manhattan Distance of the query object to each pivot object!
  - What database objects can be filtered out by using the pivot table?

Similarity Search in Multimedia Data (3)

- **Provided**: Metric space
- **Requested**:
  - Compute the pivot space for two given pivot objects!
  - Which objects are candidates for a given range query (query object, range ϵ)?
  - Draw the corresponding bounding box!