Matthew J. Graham (Caltech)

Relational databases

Jet Propulsion Laboratory
INSTITUTE FOR SPACE STUDIES
Sponsored by the Jet Propulsion Laboratory and by the Keck Institute for Space Studies, California Institute of Technology
Caltech

relational model

- Data is organized as **relations**, **attributes** and **domains**
- A **relation** is a table with columns (attributes) and rows (tuples)
- The **domain** is the set of values that the attributes are allowed to take
- Within the relation, each row is unique, the column order is immaterial and each row contains a single value for each of its attributes
- Proposed by E. F. Codd in 1969/70

transactions

- An atomic sequence of actions (read/write) in the database
- Each transaction has to be executed **completely** and must leave the database in a consistent state
- If the transaction fails or aborts midway, the database is "rolled back" to its initial consistent state

Example:

Authorise Paypal to pay $100 for my eBay purchase:
- Debit my account $100
- Credit the seller's account $100

acid

By definition, a database transaction must be:

- **A**tomic: all or nothing
- **C**onsistent: no integrity constraints violated
- **I**solated: does not interfere with any other transaction
- **D**urable: committed transaction effects persist

concurrency

- DBMS ensures that interleaved transactions coming from different clients do not cause inconsistencies in the data
- It converts the concurrent transaction set into a new set that can be executed sequentially
- Before reading/writing an object, each transaction waits for a **lock** on the object
- Each transaction releases all its locks when finished

locks

- DMBS can set and hold multiple locks simultaneously on different levels of the physical data structure
- Granularity: at a row level, page (a basic data block), extent (multiple array of pages) or even an entire table
  - Exclusive vs. shared
  - Optimistic vs. pessimistic

logs

- Ensures atomicity of transactions
- Recovering after a crash, effects of partially executed transactions are undone using the log
- Log record:
  - Header (transaction ID, timestamp, ...)
  - Item ID
  - Type
  - Old and new value

partitions

- Horizontal: different rows in different tables
- Vertical: different columns in different tables (normalisation)
- Range: rows where values in a particular column are inside a certain range
- List: rows where values in a particular column match a list of values
- Hash: rows where a hash function returns a particular value

normalisation

- First normal form: no repeating elements or groups of elements table has a unique key (and no nullable columns)
- Second normal form: no columns dependent on only part of the key

Star Name | Constellation | Area

- Third normal form: no columns dependent on other non-key columns

Star Name | Magnitude | Flux