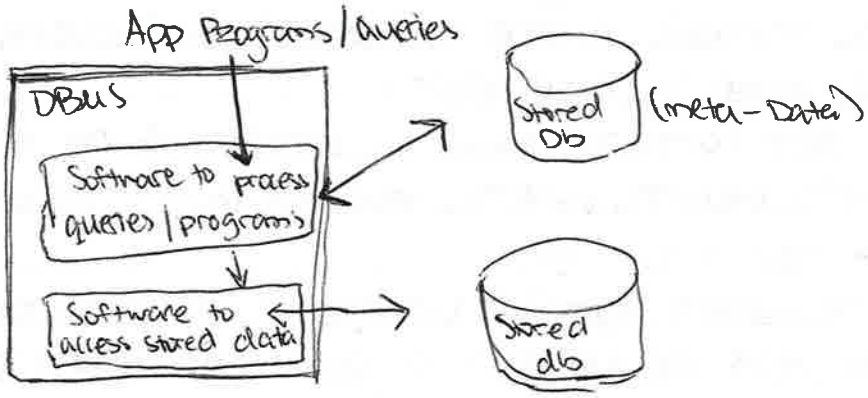
Introduction to Database Concepts

- Database → a collection of stored operational data used by the application systems of some particular enterprise. "a collection of related data."
- Enterprise → a generic term for any reasonably large-scale commercial, scientific, technical, or other application (manufacturing, financial, medical, university, government)
- Operational Data → data maintained about the operation of an enterprise (products, accounts, patients, students, plans) - does not include input/output data
- DBMS (Database Management System) → collection of programs that enables users to create and maintain a database. general-purpose software system that facilities definition of databases, construction of databases, manipulation of data within a database, sharing of data between users/applications
- Defining a database
  For the data being ~~defined~~ stored in the database, defining the database specific
  - the data types
  - the structures
  - the constraints
- Constructing a database
  - the process of storing the data itself on some storage device
  - the storage device is controlled by the DBMS
- Manipulating a database
  Includes functions that
  - retrieve specific info in query
  - updates the db to include changes
  - generates report from data
- Sharing a db
  - allows multiple users & programs access to the db simultaneously.
  - any conflicts between apps are handled by the DBMS.

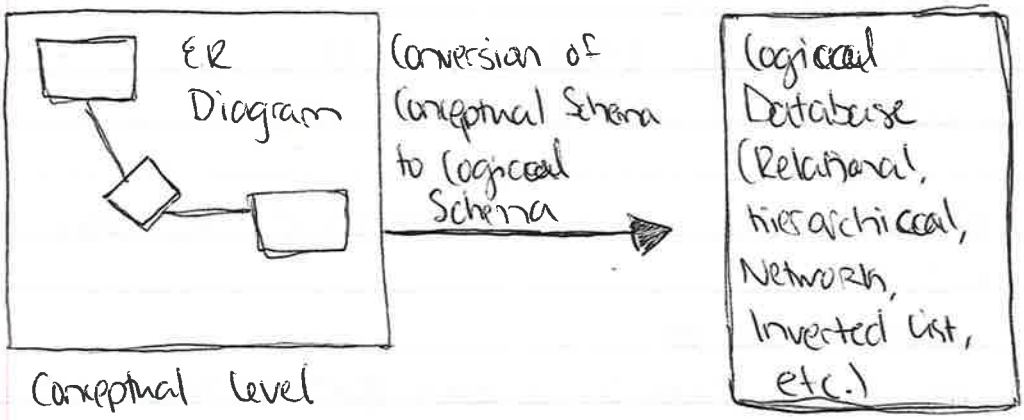
- Three Important Functions of a DB
  - Protection (system and security)
  - Maintenance (allows updates to be performed easily)
- Simplified Db System Environment

Users/Programmers

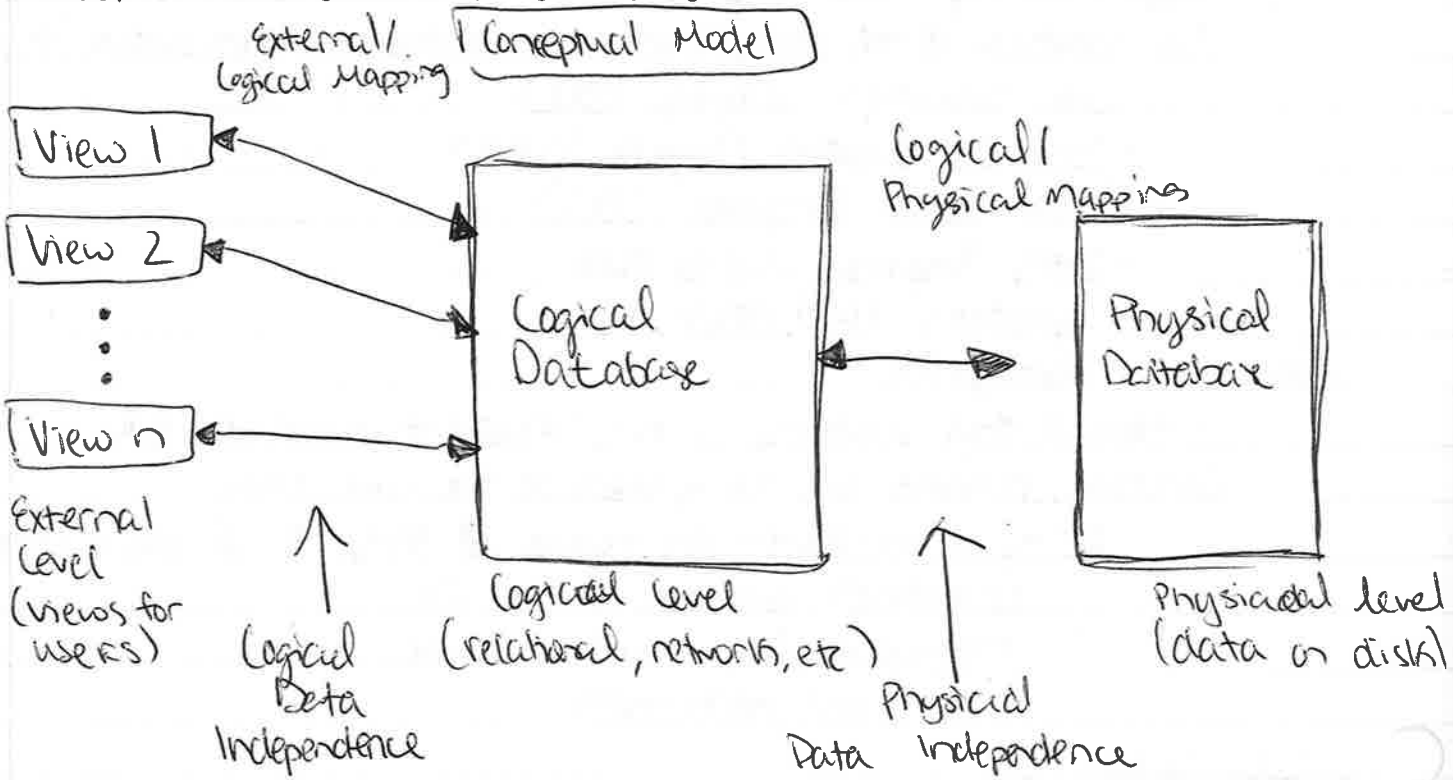


- Db Systems Characteristics
  - Self-describing nature of a dbs
  - Insulation between programs and data, and data abstraction
  - Support for multiple views of data
  - Sharing of data & multi-user transaction processing
- Other capabilities of DBMS systems
  - Support for one (at least) data model through which the user can view the data
    - there is at least one abstract model of data that allows the user to see the "info" in the db
    - Relational, hierarchical, network, inverted list, or object-oriented
    - efficient file access which allows us to "find the boss of Susie Jones"
    - allows us to "navigate within the data
    - allows us to combine values in 2 or more dbs to obtain "info"

- Support for high-level languages that allow the user to define the structure of the data, access that data, and manipulate it
  - Data Definition Language (DDL) - CREATE TABLE, ALTER, DROP
  - Data Manipulation Language (DML) - Adding, deleting, modifying
  - Data Control Language (DCL) - Authorization, GRANT
  - query language access data
  - operations (add, delete, replace).
- Transaction Management
  - a feature that provides, correct, concurrent access to the database, possibly by many users at the same time
    - ability to simultaneously manage # large # of transactions
      - procedures operating on the db
      - transaction from around the world
      - "lock out" mechanisms
- Access Control
  - the ability to limit access to data by unauthorized users along with the capability to check the validity of the data
    - protect against loss of if db crashes
    - prevent unauthorized acces to portions of data
- Resiliency
  - the ability to recover from system failures without losing data (sabotage, acts of God, hardware + software failure)
  - would also require offsite backups
- Use of Conceptual Modeling

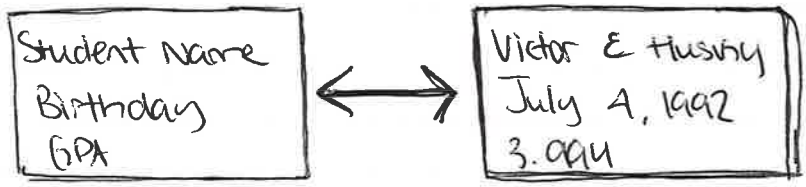


- Leveled Architecture of a DBMS



- External level
  - view or sub-schema
    portion of the logical db.
- Logical level
  - abstraction of the real world as it pertains to the users of the db
  - DBMS provides a DDL to describe the logical schema in terms of a specific data model such as relational, hierarchical, network, inverted list
- Physical level
  - collection of files and indices
  - stored on secondary device (HDD, SSD)
  - this is the actual data
- Instance
  - an instance of the db is the actual contents of the data
    - extension of the db
    - current state of the db
    - a snapshot of the data at a given point in time

- Schema
  - the data about what the data represents
    - plan of the db
    - logical plan
    - physical plan
    - the intention of the db
- Schema vs Instance

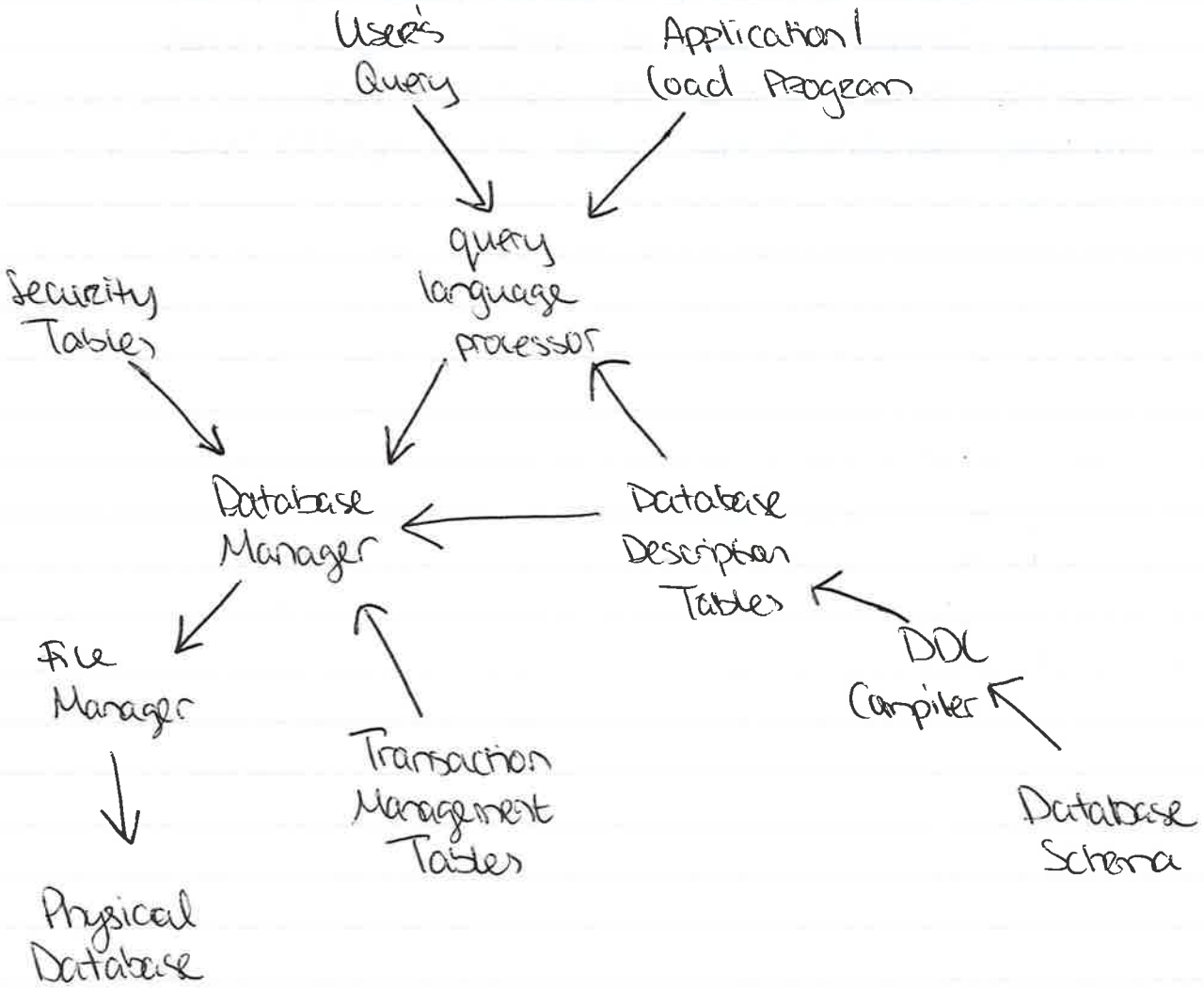


Schema
- description of what data can be stored

Instance
- the actual data that is stored

- Data Independence
  - a property of an appropriately designed db system
    - Mapping of logical level to physical level, and logical to external
    - physical data independence
      - physical schema can be changed without modifying logical schema
    - logical data independence
      - logical schema can be changed without having to modify any of the external views
- DCL, DDL, DML
  - may be completely separate
  - may be intermixed

- DBMS Components



- Overall DBMS Usage Scenario
  - DBA defines the conceptual, logical and physical levels using DDL
  - stores instances of these in schemas
  - users defines views (external schema) in DDL
  - users access db using DML
- Advantages of a DB
  - controlled redundancy
  - reduced inconsistency in the data
  - shared access to data
  - standards enforced
  - security restrictions maintained
  - integrity maintained more easily
  - provides capability for backup and recovery
  - permitting inferences and actions using rules

- Disadvantages of a DB
  - increased complexity needed to implement concurrency control
  - increased complexity needed for centralized access control
  - security needed to allow the sharing of data
  - necessary redundancies can cause complexity when updating