

## Database Quick Revision Notes

\*Standard database stuff

-Insert,Delete,Update,Query

-Sharing Data

-Centralised Control

--Reduces Redundancy

--Increases Consistency

--Integrity

--security

-valuable asset

-underpins ebuisness

-Relation, attribute, domain, tuple, degree, cardinality

-Keys

--superkey - A key that can identify a tuple

--candidate key - A superkey with no subsets

--primary key - Candidate key for that relation

--foreign key - An attribute thats used as a primary key elsewhere

-entity integrity

--Primary key cant be null

-refrential integrity

--Foreign key consistency

Conceptual Models

- what the text says it should look like

Logical Models

- add what is actually required to make a db to the conceptual model

- what is requiried in practice

SQL - std and joins

-INNER JOINS

--SELECT A,B FROM FLIGHTS INNERJOIN PLANE WHERE plane.ID=Flight.ID Where Date=today

Relation schemes

- Foreign keys NOT shown

Lists of schemes - {...}

Normalization

-0

--One flat database

-1st

--Only 1 item per a column

-2nd

--Each non-key must attribute must depend on primary key

--remove all obvious things

-3rd

--Attributes MUST FULLY depend on primary key

--remove picky little things

-Why

--stability

--endurance

-Why not

--Slower

--easier

## Higherarchical

- Tree structure hard programmed

## Distributed

- organisation
  - client server
  - client multiple servers
  - shared on several machines)
- Why
  - improve performance
  - increase reliability
    - data replicated
    - sharing data
  - Maintain consistency thru replicated data
  - load balancing over multiple copies of db
  - recover from crashes and hardware failures

## Network Databases

- Records can be of different types
- all records linked with 1-n relationships
- no constrain on number or direction of links
- no need for root record type
- data stored in records
- can have virtual data items derived from other data
- How
  - Sets using pointers
  - duplication reduced or eliminated
  - Manipulation is HARD requires programming
  - Navigation is programmed in
  - Modifications can require re programming

## Relational Databases

- good queries
- simple data

## Object Oriented Databases

- complex data

## Object Relational

- complex data
- queries

## ACID properties

- Atomicity - Happens or doesn't
- Consistency - Always does the same
- Isolation - Effect does show until finished
- Durability - Changes are permanent

## Locking

- Can cause deadlocks
- 2 Phase locking
  - Guarantees serialisable
  - All locking precede first unlock in trans
  - Locks only released after transaction ends

## 2 phase commit protocol

- Commit C
- Prepare S

--Ready Commit C  
--Global S

Serial - runs one thing at a time  
Non Serial - items are interweaved  
Serialisable - items are interweaved but can be serialised

lost update  
dirty read  
unrepeatable read  
incorrect summation

? Data Models ?  
-----DOESNT EXIST THIS YEAR HOPEFULLY-----Three Level  
architectures