The Relational Model Part 2

Integrity Constraints: Conditions specified on a database schema

Legal instances: instances that satisfy ICs

Types of ICSs: keys, foreign keys, domain constraints

Keys

A set of fields is a **superkey** if: no two distinct tuples can have the same values in all key fields

A set of fields is a **key** for a relation if:

- It is a superkey
- No subset of the fields is a superkey
- I.e. a minimal superkey

Properties of Keys

Given a set of attributes A (can contain 1 or multiple attributes)

- If A is a key, any superset of A cannot be a key because the key must be minimal
- If A is not a key, any subset of A cannot be a key.

How to find all keys efficiently?

Naive way: enumerate all possible attribute sets?

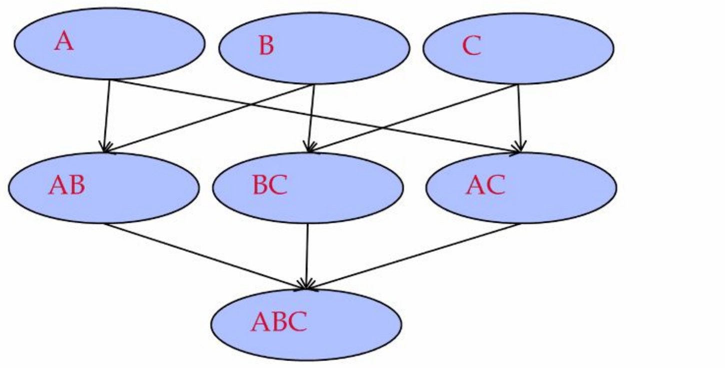
Efficient way: construct an attribute lattice and make use of the properties of keys when traversing the attribute lattice to find keys.

Attribute Lattice

Top of lattice: each node contains a single attribute

Bottom of lattice: one node of all attributes

Edge N1->N2: N1 is a strict subset of N2



An example of attribute lattice of 3 attributes {A, B, C}

Define candidate keys in SQL

When > 1 key for a relation, one is chosen as the primary key, while the others are called candidate keys.