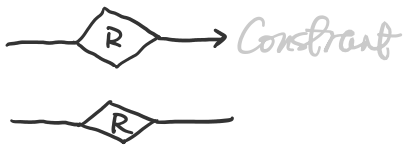
Integrity Constraints

Wednesday, May 22, 2019 1:30 PM

- **Integrity Constraint:** a condition specified on a database schema that restricts the data that can be stored in an instance of the database.
  - To make the data consistent.

- Constraints in E/R
  - keys
  - single value constraints
  - Referential integrity constraints
  - general constraints
- keys in E/R diagrams No formal way to specify multiple key.
- Single Value Constraint



- Referential Integrity Constraints

Suppose a→b.

b can be null.

b can't be null.

- Other Constraints

- Constraints in SQL : keys, attribute level, tuple-level constraints, general (complex) constraints

  - Key

```
CREATE TABLE T(
    a1 type PRIMARY KEY,
    a2 type
);
```

```
CREATE TABLE T(
    a1 type,
    a2 type,
    a3 type,
    a4 type,
PRIMARY KEY( a1 ),   <- at most one
UNIQUE (a2,a3));
```

It is not suggested to put primary key into unique.

```
CREATE TABLE T2(
    a1 type references T(a1),
    b2 type
);
```

references T(a1): only primary key can be referenced.

Referential Integrity Constraints

  - Deal with data change:

    - Search the key:

      BTree, BST] O(log(n))

      BTree has to be wide because it can only scan one block.

B+ tree is never deeper than 6.

- SQL policies for maintaining referential integrity when want to modify data.
  - NO ACTION
  - CASCADE delete the key, then delete its corresponding foreign key
  - SET NULL
  - SET DEFAULT create table (pid int default 42 references...)

```
CREATE TABLE Purchase (
        prodName CHAR(30),
        category VARCHAR(20),
        date DATETIME,
        FOREIGN KEY (prodName, category)
          REFERENCES  Product(name, category)
          ON UPDATE CASCADE
          ON DELETE SET NULL   )
```

- Constraints on tuples

  att type NOT NULL

  att type CHECK condition -- reject the tuple if doesn't satisfy.

  - CREATE TABLE Purchase
    prodName CHAR(30)
    CHECK (prodName IN
    (select P.name
    from Product P)),
    date DATETIME NOT NULL);

    → Work the same as foreign key.
    · p.name doesn't have to be key.
    · take more time than foreign key
    · when delete a key in P will also check Purchase.

- General Assertions

- CREATE ASSERTION myAssert CHECK
  (NOT EXISTS(
    ...
  ));

- But most DBMSs do not implement assertions
  ∵ hard to support them efficiently.