Database Integrity

Martin Read

Aim: Introduction to data integrity

- Write SQL statements that manipulate the existing structure to ensure data integrity enforced.

Words of warning!

"Integrity" means different things to different people…

- In some books, "integrity" is linked to such things as transaction completion and concurrency
- We are considering only integrity as applied to Relational models and databases
- even then… "foreign key" is applied by many authors to mean the master of any relationship pair

Data Integrity

Domain	A pool of allowable values from which the real values are drawn
Entity	Ensures no attribute of a primary key can be null. The primary key is unique. EVERY table must have a primary key
Referential	Ensures where a foreign key exists in a table, the foreign key value must match a candidate (usually primary) key of a row in the home table or be completely null
Semantic	Ensures application constraints applied or that business rules are obeyed

Integrity in SQL

- Use SQL to implement integrity by using constraints
- **Column** constraints
  - Must apply to only one column
- **Table** constraints
  - May apply to one or more columns

- ALSO use the following
- UNIQUE
- NOT NULL
- CHECK

Entity integrity

This is concerned with the "realism" of a database

Implications

- every real world 'thing' about which data needs to be maintained in an application must be represented by a relation in the model
- every set of properties about a real world occurrence of the 'thing' must be uniquely identifiable
- every row (tuple) in a relation must have a unique-valued Primary key
- Primary key attributes must not be null

Entity integrity

- Primarily an Analysis and Design issue
  - You cannot allow nulls (blanks) in indexed fields
    - usually = key fields
    - Primary Key - Is unique and has to have a value in it

```
CREATE TABLE Orders
..
CONSTRAINT pk_Orders PRIMARY KEY
```

Referential integrity

In our terminology, this applies to any master-detail key attribute pairing

Analysis/data entry/processing requirements **must** drive the use of Referential Integrity

The Referential Integrity rule says:

every non-null value that exists in a detail (foreign) key attribute must also exist in the relation for which it is the master key.

In most cases, this means

"...for which it is the Primary key."

The problems show up in field use when a value is created, updated, or deleted

Referential integrity (2)

The problems show up in field use when a value is created, updated, or deleted

There are three strategies

- Restrict = ban alteratic
  references still exist
  ON DELETE/ON UPDAT
- Cascade = effect the a
  referenced by the original value (e.g. Triggers)
  ON DELETE/ON UPDATE CASCADE
- Set to null = allow alteration in the master value but set all values in detail rows to null
  ON DELETE/ON UPDATE SET NULL

```
CREATE TABLE OrderDetails
..
CONSTRAINT fk_Orders REFERENCES Orders(OrderID) ON
DELETE NOACTION
```

https://www.sqlshack.com/delete-cascade-and-update-cascade-in-sql-server-foreign-key/

On delete cascade

- Can be very dangerous
- The data is gone forever!
- Better to flag for deletion and write to an archive table with a trigger (you learn more about those later)
- Provide a clear audit trail with no loss of data

Relationships

- Unless the analysis indicates otherwise, Referential Integrity should always be enforced
- One-to-many relationships can be created which allow for optional detail relationships
  - done by not enforcing Referential Integrity

- Define relationships between tables to ensure referential integrity
  - Foreign key constraints

Domain integrity

- Every value in a database must be a member of some (real world) domain
- The set of domains on which the database is defined is the database's vocabulary and defines all the facts about the real world which can be represented
- Domain values must be single-valued (atomic)
- Domain integrity is concerned with ensuring that fields in the database contain only real world values

Domain integrity

- Generally managed
  - through the various Data Types (which constrain data values e.g. valid dates)
  - use of an Input Mask and/or Validation Rules to constrain patterns or ranges of values

- Constraints enforce data integrity, by defining rules for columns
  - CHECK, NOT NULL, UNIQUE

Domain Integrity

- Can be achieved using the following
  - Built in data types eg Date, Number
  - User defined data types
  - Restriction on a range of allowable values
  - Required value
  - Unique value

- Example: Number datatype
  - Number(p,s)
  - P = precision – total number of digits
  - S = scale – number of digits after decimal point
  - Eg: Number(5) = Number(5,0) which is an integer
  - A float would have Number(5,2) for example

Date and Time

- In most SQL implementations the Date data type represents both date and time.
- Represented internally as a number
- Need to use functions to convert
  - A date to a string for output
  - A string to a date for input
  - According to a format mask

```
SELECT getDate()
//get current date


SELECT
CONVERT(VARCHAR(10),
getdate(), 103);
```

Semantic integrity

Ensures that application constraints are applied or that business or enterprise rules are obeyed

- Refers to the meaning (actual or implied) which is given data by its context
- In a Relational database, data is passive, and is given its context by the application processing

Semantic integrity

- Semantic integrity is not an issue that is specifically addressed in relational databases
  - it is possible to implement some processing rules in programming

Semantic integrity

Semantic integrity relies on an understanding of what values should be present, & how they are created, read or referenced, updated, or deleted according to the specific business application

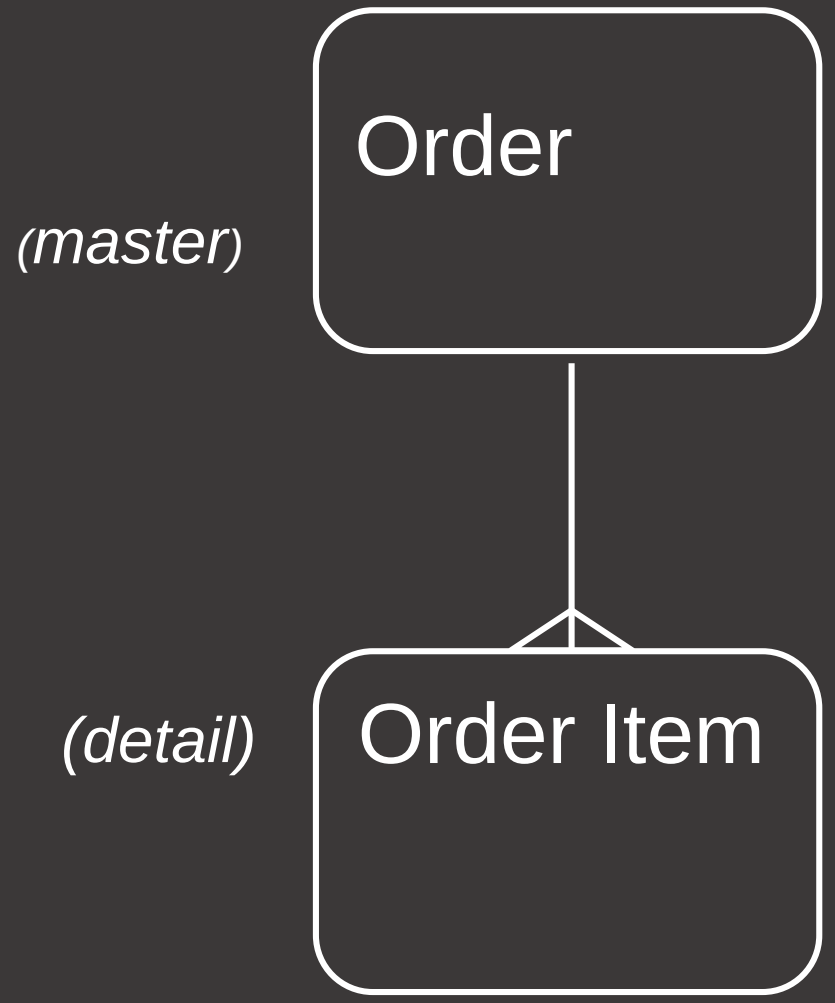
Example

The Date Paid attribute on the Penalty entity is drawn on the domain of all valid dates…

but must never be earlier than the Date Penalty Applied attribute, otherwise the data would be out of a correct context

e.g. CHECK (penalty_paid >= penalty_date)

other integrity issues



each order is a list of one or more items

each order item is listed on one and only one order

other integrity issues

- Rules about data (validation & integrity) are **set** at the TABLE level but **enforced** on FORMs

Crafting tables

- Incorporate as much integrity as possible into the database schema
- Do not leave the integrity to the application developers
- Data checked before travelling
  - On both sides
- (Use the checklist)