WTF with SQL

It’s just not right!

Kennie Nybo Pontoppidan

Cambridge, September 12th 2015

Agenda

- Sets
- Relational algebra
- Three Valued Logic
- SQL WTFs
  - NULLs
  - Count
  - Union
  - PK and FK
  - …

Sets

The axioms of Zermelo-Fraenkel set theory with choice ZFC

In principle all of mathematics can be derived from these axioms

Extensionality	$\forall X \forall Y [X = Y \Leftrightarrow \forall z(z \in X \Leftrightarrow z \in Y)]$
Pairing	$\forall x \forall y \exists Z \forall z [z \in Z \Leftrightarrow z = x \text{ or } z = y]$
Union	$\forall X \exists Y \forall y [y \in Y \Leftrightarrow \exists Z(Z \in X \text{ and } y \in Z)]$
Empty set	$\exists X \forall y [y \notin X]$ (this set X is denoted by $\emptyset$)
Infinity	$\exists X [\emptyset \in X \text{ and } \forall x(x \in X \Rightarrow x \cup \{x\} \in X)]$
Power set	$\forall X \exists Y \forall Z [Z \in Y \Leftrightarrow \forall z(z \in Z \Rightarrow z \in X)]$
Replacement	$\forall x \in X \exists! y P(x, y) \Rightarrow [\exists Y \forall y (y \in Y \Leftrightarrow \exists x \in X (P(x, y)))]$
Regularity	$\forall X [X \neq \emptyset \Rightarrow \exists Y \in X (X \cap Y = \emptyset)]$
Axiom of choice	$\forall X [\emptyset \notin X \text{ and } \forall Y, Z \in X(Y \neq Z \Rightarrow Y \cap Z = \emptyset) \Rightarrow \exists Y \forall Z \in X \exists! z \in Z (z \in Y)]$

Set (vague definition)

- A set consists of it's members
  - No duplicates

- If A and B are sets, so are
  - A∪B (union)
  - A∩B (intersection)
  - A\B (difference)

RDBMS WTF

- Database tables are not sets, they are multisets

- Demo script
  - 01 - multisets.sql

Relations

- A relation R consists of a header H and a body B
- The header H is a set of pairs (name, type)
- The body B is a set of tuples T
- A tuple T is a set of values conforming to the specification in H

RDBMS WTF

- Database tables are not relations
  - Header is not a set
  - Tubles are not sets
- Demo script
  - 02 - relation.sql

Relational algebra

- On the set of relations, we can define some operators
  - π (projection)
  - σ (selection)
  - ⋈ (natural join)

SQL naming WTF

Relational operator	SQL syntax
π (projection)	SELECT
σ (selection)	WHERE
⋈ (natural join)	NATURAL JOIN

Three valued logic

AND	TRUE	FALSE	NULL
TRUE	TRUE	FALSE	NULL
FALSE	FALSE	FALSE	FALSE
NULL	NULL	FALSE	NULL

=	TRUE	FALSE	NULL
TRUE	TRUE	FALSE	NULL
FALSE	FALSE	TRUE	NULL
NULL	NULL	NULL	NULL

TVL is hard

- Examples taken from
  - http://www.techrepublic.com/article/oracle-tip-understand-how-nulls-affect-in-and-exists/
- Demo script
  - 03 nulls.sql

Nulls are hard as well

From Codd's 12 rules

Rule 3: Systematic treatment of null values

- The DBMS must allow each field to remain null (or empty).
- Specifically, it must support a representation of "missing information and inapplicable information" that is *systematic, distinct* from all regular values (for example, "distinct from zero or any other number", in the case of numeric values), and independent of data type.
- It is also implied that such representations must be manipulated by the DBMS in a systematic way.

- https://en.wikipedia.org/wiki/Codd%27s_12_rules

Sum vs. + WTF

- Aggregate sum operator (sum column values) and the + operator (sum values in the row) have different semantics regarding null values

- Demo
  - 04 - sum.sql

Let's count…

- SQL has different semantic rules for equality
  - Unique operator
  - Duplicate elimination
- We can construct WTFs with
  - strings
  - Nulls (nulls are always more fun…)
- Demo
  - 05 count demo.sql

Null wrap up

- Until now we have seen three different semantics of the expression
  - null = null

Demo	Expression	Semantics
Demo 3	select 'any rows?‘ where null = null	Unknown
Demo 5	Select distinct count …	False
Demo 5	Select distinct …	True

Codd’s 12 rules again

Rule 3: Systematic treatment of null values:

The DBMS must allow each field to remain null (or empty). Specifically, it must support a representation of "missing information and inapplicable information" that is systematic, distinct from all regular values (for example, "distinct from zero or any other number", in the case of numeric values), and independent of data type. **It is also implied that such representations must be manipulated by the DBMS in a systematic way.**

Other places with weird semantics of =

- String comparisons are fun
  - Collations are not helping
- Demo
  - 06 count strings demo.sql
  - 07 max demo.sql

String comparison semantics for profit and fun

- Let’s make some money, then…

- Demo
  - 08 transfer money demo.sql

Primary keys and foreign keys

When

‘WTF’ is equal to ‘wtf’

we can have more fun…

- Demo
  - 09 primary key.sql
  - 10 foreign key.sql

The winner takes it all

- {1,2,3,4}\{1} = ?
- Well, not always in SQL
- Demo
  - 10 cross join demo.sql

Fun with white space

- SQL Servers default settings dealing with white space…

- Demo
  - 12 white space demo.sql
  - 13 white space insert demo.sql

Floating point numbers

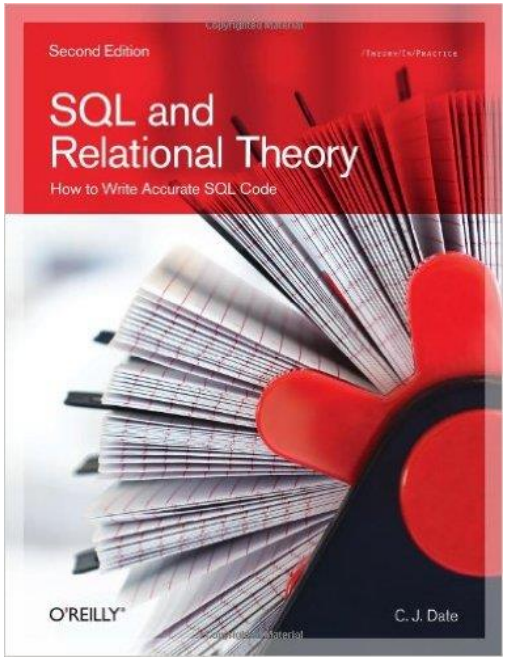
- Just be careful
  - 59.95 cannot be finitely represented in IEEE 754 base 2
- Demo
  - 14 float demo.sql

Want to know more?

Buy this book

SQL and Relational Theory: How to Write Accurate SQL Code

by C.J. Date



Or buy his video master class

http://shop.oreilly.com/product/0636920002710.do