

1999

p12q9.

A notes)External  
level

VIEW A

VIEW B

Sub-schema

Conceptual  
levelDATA  
MODELSCHEMA  
DDLInternal  
(storage) levelSTORAGE  
DATAStorage  
mapping

The specific model representing an enterprise DB is defined in the Schema DDL. This enterprise model can then be realised in any environment that supports the same Schema DDL (DBTG, SQL, ODL or whatever).

A notes, ctd)

There is another way in which the ~~architecture~~ supports data independence.

Applications access the database through views. If the data model is extended / modified, changes not visible in a particular view will not affect associated applications.

---

Codd's Relational Model of 1970. A relation is defined over data domains  $D_i$ , each a set of values; each relation has some fixed finite degree  $n$ , so that

$$R \subseteq (D_1 \times D_2 \times D_3 \times \dots \times D_n)$$

is a set of  $n$ -ary tuples.

A specific data model is defined by giving the form of a finite set of relations  $\{R_i\}_{i=1}^k$ , which may of course be of different degrees.

A notes.cdd)

Codd required that relations

should be in First Normal Form (1NF), that is that each  $D_i$  should be a simple scalar set of data values

---

Strengths a) DATA INDEPENDENCE

b) NO ACCESS PATH SPECIFICATION

$\Rightarrow$  OPPORTUNITY FOR OPTIMISATION

c) CLEAN SEMANTICS FOR DBPL (can be based on relational algebra / calculus)

d) RELATION  $\equiv$  PREDICATE, consistent with DEDUCTIVE DB approach

+ no doubt others!

---

Weaknesses e) TABULAR SEMANTICS are inappropriate for lots of real data instances:

— repeating groups

— conditionally defined attributes

A notes, etc)

- f) FLAT SCALAR domains are hopeless
- specialisation / generalisation
  - partial information
  - higher order data
- g) "relationally complete" DBPL have weak semantics (lack control structures, cf. DATALOG & fixed-point semantics)
- + no doubt others!

---

Fundamentally, SQL as a Standard. NOT nice, cf embedded SQL & cursor variables.

BUT universally accepted throughout the industry amenable to optimisation / parallel exec<sup>ns</sup> suitable target for e.g. SQL controlled by ANSI (1986, 1992)

notes, ctd) Relational model was the first  
to introduce TRUE data independence, enabling  
DBMS to be constructed that supported complex  
enterprise modelling.

But it's SQL that has kept ~~OODB~~  
at bay.

---