~~Databases~~ 1999 QuB, Paper 6/13, notes p6q8 & 2
JKMM p13q9

B notes

a) superclass ⇒ generalisation/specialisation

b) higher order types, collections etc.

c) entity integrity via OIDs.

+ no doubt others will emerge

~~Transport business:~~

we can identify (at least) 4 different categories of employee at once —

a) administrative staff;

b) drivers;

c) loading/unloading;

d) transport maintenance.

That's enough to be going on with.

OK

B notes, ctd) ODMG Database Standard

Defines a Standard Object Model with base types and constructors, including collection types. User defined types can refer to supertypes. Behaviour is specified to method signatures only, except for the existence of standard operations on base types and collection types.

A standard meta-object format is defined for representing object schemata, which are specified via the language ODL.

Methods must be defined by a particular implementation class, in one of the OOPL for which an ODMG binding is defined.

Note that relations are simply representable as bag < struct < .... >>

SQL3 defines object extensions to the Relational Model of the SQL 1992 standard.

The extensions take two different forms:

a) the ability to manipulate rows of relations as if they are objects. One can declare CREATE ROW TYPE (<components>) to specify a tuple type, and refer to the type either to form a sub-component of other ROWs or relations or via a reference variable in (some other) ROW or relation. This gives facilities equivalent to the ODL bag<struct<...

b) manipulation of abstract data types.

This offers functionality as expected, with the exception of supertypes. There are also NO collection type constructors.

B Notes ctd) In SQL3 methods can be specified in two ways:

a) INTORNAL — a simple procedural language for accessing ADTs;

b) EXTORNAL — signature only, with a reference to an explicit programming langua

---

In the employee example, SQL3 only does part of the job. A ROW type could be created for the common employee data, with separate extensions for the specific subtypes. This is a less satisfactory solution than can be reached via a superclass in the ODMG Object Model. Maintenance staff are qualified for a set of vehicle types; again, the collection types of ODMG are.

QuB Notes, ctd)

The main drawbacks associated with the use of an OODB come from loss of data independence. OIDs provide conveniently for ~~data~~ entity integrity in a centralised system, but they are hopeless under federation. But in fact there's nothing to preclude the inclusion of application level keys ...