Dr Greg Wadley

INFO 90002 Database Systems & Information Modelling

Week 06

Data Dictionaries

Two meanings of ‘data dictionary’

- manually-maintained documentation
- system-maintained repository

Both store metadata about the database.
Manually-maintained documentation can explain:

- the meaning of terms in the ER diagram
- business rules that are not documented in the ER diagram
- sources of data

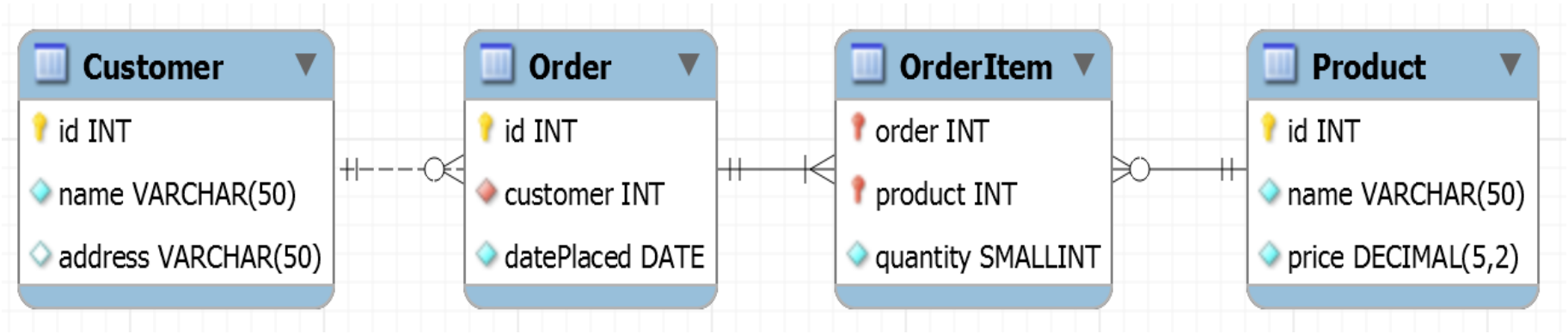
As a data modeller, you will mostly be concerned with manual documentation. You need to write one in Assignment One.

As an administrator of developer, you will sometimes need to use the system-maintained repository.

Therefore in this lecture we discuss both types.

Manually-maintained data dictionary

Example basic dictionary structure



- recall our *Orders* example (ER diagram shown above)
- many business rules are represented (e.g. keys, types …)
- some are not, e.g. min/max of quantity, price
- meanings are not described, e.g. customer.name
- sources of data are not described, e.g. datePlaced

Examples of manual documentation

- on LMS / Resources

Example fragment of Data Dictionary for TV database

Entity 'CHANNEL'

A TV station which is broadcasting programs on the channel in question.

Attribute	Description
channel_id	While this primary key is a unique integer, it is also the number by which the TV station is best known on a TV tuner. E.g. '9' for Channel Nine, '2' for the ABC, etc.
channel_name	The name of the TV station itself e.g. 'Channel NINE' or 'ABC 1' or 'SBS'.

Entity 'PROGRAM'

A program (movie, news broadcast, episode of a series etc) to be broadcast. There will be one row for each episode of an ongoing program (e.g. a soap opera or the nightly news), but not for 'repeat' broadcasts'. The entity PROGRAM holds 'static' data about a program: those attributes which do not varying from showing to showing.

Attribute	Description
program_id	This is the primary-key of PROGRAM, automatically generated by the DBMS. If a movie is shown three times during the year, there will only be one record in this entity, the details of which will be used by all three broadcasts.
title	Holds the actual title of the program e.g. the name of a movie, or 'ABC News' a regularly timetabled news program.
description	A description of the program. It may be up to 255 characters, which is large enough to hold the small descriptions published in TV guides.
country	The name of the country where the program was produced. If the programs is a collaboration between companies or organisations across two or more countries, this field will hold just the country of the primary collaborator.
year_made	The four-digit year that the program was produced in. e.g. '2006', for a movie released in 2006.
classification	The formal classification assigned to the Program reflecting community standards regarding what children, teenagers and adults should see. Can hold one of the following values: 'G', 'PG', 'M', 'MA15+', 'AV15+'.
high_definition	This field holds either a 'T' for True, or 'F' for False- regarding whether or not the program was filmed in High-Definition. Television programs can be filmed at different image qualities (scan-lines and pixels per line). High Definition shows more scan-lines and more pixels per line..
sub_titled	Holds either True or False, indicating whether or not the program comes with sub-titles (in English)
episode_no	In the case of a series which has multiple episodes, each episode will have its own record in this PROGRAM entity, and therefore, this attributes tells us the number of the episode this record actually represents. If the program doesn't have multiple episodes, it should be left as NULL.

Entity Class/Data Item	Constraints
Student Absence	No date/time overlaps between records for the same Student
be for Student	Mandatory; Student must already exist
Start Date	Mandatory; must be valid date; must be within reasonable range
End Date	If entered: must be valid date; must be not be before Start Date; must be within reasonable range
First Timetable Period No	Mandatory; integer; must be between 1 and maximum timetable period no inclusive
Last Timetable Period No	If entered: integer; must be between 1 and maximum timetable period no inclusive; must not be less than First Timetable Period No
be classified by Student Absence Reason	Mandatory; Student Absence Reason must already exist
Notification Date	If entered: must be valid date; must be within reasonable range
Absence Approved Flag	If entered: must be Yes or No
Student Absence Reason	
Absence Reason Code	Mandatory; must be unique
Description	Mandatory; must be unique

Figure 14.1 Some data validation rules.

- from Simsion textbook

System-maintained data dictionary

System-maintained data dictionary

- Each RDMS automatically maintains a data dictionary or repository of meta data about the databases stored on that server, their tables, contents and usage.

- The repository consists of the following typical information:
  - Names of the tables in the database
  - Names, type and other information on the columns for each table
  - Constraints of a table. Keys, Relationships, etc.
  - Owner and authorised users of the table
  - Last accessed information of objects
  - Last updated information of objects
  - Engines, character types

the MySQL repository

- The `mysql` system schema contains information required by the MySQL server as it runs. This database contains data dictionary tables and system tables.
- Data dictionary tables are protected and may only be accessed in debug builds of MySQL.
- Thus, data dictionary tables are invisible. They cannot be read with SELECT, do not appear in the output of SHOW TABLES and so forth.
- However, MySQL provides INFORMATION_SCHEMA views that show equivalent information. Conceptually, the INFORMATION_SCHEMA provides a view through which MySQL exposes data dictionary metadata.

information_schema views

- The information schema (information_schema) is an ANSI-standard set of read-only views which provide information about all of the tables, columns, constraints, etc in a database management system.

- Some important examples:
  - COLUMNS – Return one row for each column the current user has access to use in the current database. This view can be used to determine the data type and table the column is defined for use in.
  - TABLES – Return one row for each table the users has access to use within the current database. Note, both tables and views are returned using the TABLES view.
  - The REFERENTIAL_CONSTRAINTS table provides information about foreign keys.
  - The KEY_COLUMN_USAGE table describes which key columns have constraints and information about those constraints.

information_schema examples

- The `mysql` dictionary is not directly accessible

```
mysql> SELECT * FROM mysql.schemata;
ERROR 3554 (HY000): Access to data dictionary table 'mysql.schemata' is rejected.
```

- Instead, to browse tables/schemata, start with "SELECT * FROM INFORMATION_SCHEMA.SCHEMATA"

CATALOG_NAME	SCHEMA_NAME	DEFAULT_CHARACTER_SET_	DEFAULT_COLLATION_NAME	SQL_PATH
def	information_schema	utf8	utf8_general_ci	NULL
def	aware_test	utf8	utf8_general_ci	NULL
def	bank	utf8	utf8_general_ci	NULL
def	mysql	utf8	utf8_general_ci	NULL
def	orders	utf8	utf8_general_ci	NULL
def	performance_schema	utf8	utf8_general_ci	NULL
def	socialmedia	utf8	utf8_general_ci	NULL
def	sys	utf8	utf8_general_ci	NULL

information_schema examples

- SELECT * FROM `TABLES` WHERE TABLE_SCHEMA = 'orders'

TABLE_CATALOG	TABLE_SCHEMA	TABLE_NAME	TABLE_TYPE	ENGINE	VERSION	ROW_FORMAT	TABLE_ROWS	AVG_I
def	orders	customer	BASE TABLE	InnoDB	10	Dynamic	2	8192
def	orders	order	BASE TABLE	InnoDB	10	Dynamic	5	3276
def	orders	orderitem	BASE TABLE	InnoDB	10	Dynamic	11	1489
def	orders	product	BASE TABLE	InnoDB	10	Dynamic	4	4096

- SELECT TABLE_NAME, COLUMN_NAME, CONSTRAINT_NAME, REFERENCED_TABLE_NAME, REFERENCED_COLUMN_NAME FROM INFORMATION_SCHEMA.KEY_COLUMN_USAGE WHERE CONSTRAINT_SCHEMA = 'orders';

TABLE_NAME	COLUMN_NAME	CONSTRAINT_NAME	REFERENCED_TABLE_NAME	REFERENCED_COLUMN_NAM
customer	id	PRIMARY	NULL	NULL
order	id	PRIMARY	NULL	NULL
order	customer	fk_Order_Customer	customer	id
orderitem	order	PRIMARY	NULL	NULL
orderitem	product	PRIMARY	NULL	NULL
orderitem	order	fk_OrderItem_Order1	order	id
orderitem	product	fk_OrderItem_Product1	product	id
product	id	PRIMARY	NULL	NULL