Fungi Database.

Ioannis Valasakis[1]

[1]Birkbeck, University of London

March 5, 2018

Abstract

The following coursework coursework involves designing a database to manage selected data from an environmental risk assessment concerning the growth of fungi within buildings, such as heritage buildings owned by the National Trust.

Introduction

Schemata

Logical Schema

The required database will have the above shown logical schema (describing integrity constraints, views, and tables).

Physical Schema

The physical schema shows how data exist physically on a storage system in terms of files and indices.

Relational Table Definitions

They are included in the attached `iv_cw_db.sql`file together with some dummy data for testing the database.

Decisions on the database design

Cardinality is the quantity of elements that interact between two specific tables that are related. The identification of cardinality helps to ensure that the data have been divided into tables most efficiently. Looking at the schema, the cardinality as well as some certain decisions taken can be observed.

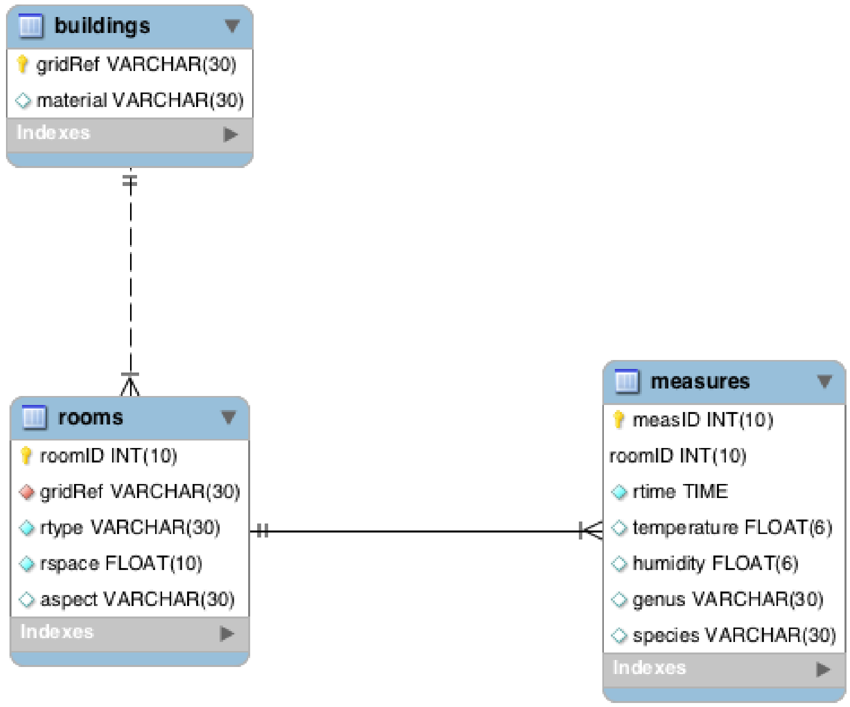

Figure 1: Entity Relationship Hierarchical Model of Fungi Database

Online analytical processing (OLAP) databases which favour analysis and reporting might fare better with a degree of denormalisation since the emphasis is on the speed of calculation. These include decision support applications in which data needs to be analysed quickly but not changed, which is the case given in our fungi database.

In this aspect, I choose to denormalise the *material* from the table *buildings*. Still, kept the rule of the second normal form ($2NF$).

The entity integrity rule says that the primary key can't be NULL, which is the case in the current design as well. In addition, if the key is made up of multiple columns, none of them can be NULL. Otherwise, it could fail to uniquely identify the record. The referential integrity rule requires each foreign key listed in one table to be matched with one primary key in the table it references, as shown above.

In a summary, for the logical schema, there are three tables *buildings*, *rooms*, *measures* (short for measurements). There is a one to many relationship between *buildings* and *rooms* (as a building can have multiple rooms). Same between *measurements* and *rooms,* as multiple *measurements* can be performed on the same room (defined by the *measID* and *roomID*).

The described schema is concurrent with the following requirements (as much as possible) in order to ensure seamless transition: *Extended overlap preservation* (as elements associated with overlapping elements are copied to the database schema), *Normalisation* (Independent relationships between entities are kept in seperate tables), *Minimality* (none of the elements in any of the sources are lost).

Finally, as genus and species (as well as room *types* and room *aspect*) are going to be frequently looked up, there was the choice of indexing them.

Database queries

The answers to the specific database queries are included in the attached file named `iv_cw_dq.sql`