MonkeDB

Data and Applications Project Phase 3

Team WeDontKnown

Gaurav Singh, 2020111014

Pratyay Suvarnapathaki, 2020111016

Yash Mehan, 2020111020

Our Mini-World

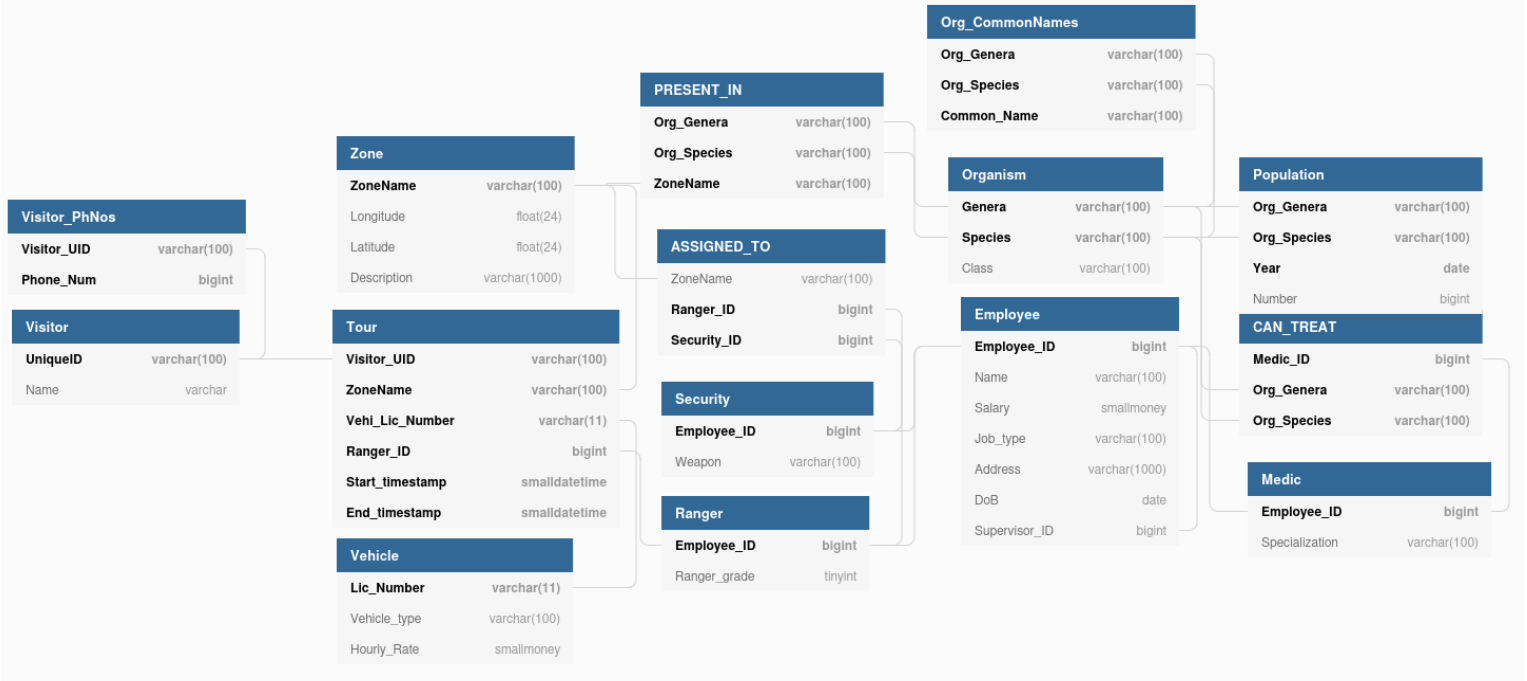
Mini World: A **National Park**

Converting from ER to Relational Model

The conversion from the ER model as finalized in Phase 2 to a relational model was carried out in the following manner:

- ➔ Each **strong entity** has been represented as its own **entity relationship**.
- ➔ **Derived attributes** have not been explicitly mentioned in the relational model, as is the convention.
- ➔ **Composite attributes** have been represented as their distinct sub-component.
- ➔ **Multivalued attributes** have been represented as their own separate relation, involving a foreign key from the entity whose attribute they are.
- ➔ Each **weak entity** has been represented as an entity relationship with a **foreign key** corresponding to each strong entity that is involved in its identifying relationship. Consequently, the **identifying relationships** themselves have hence been mapped too.
- ➔ **1:N and N:1 relationships** have been represented as foreign key attributes in the ‘N-side’ relation.
- ➔ **M:N and n>2-ary relationships** have been represented as their own **relationship relations** consisting of foreign keys from participating relations.
- ➔ The **subclasses** (of the Employee type) have been implemented as separate relations.
- ➔ The **value sets** of the attributes have been updated to match **SQL data types**. Consequently, some **changes have been made** for optimizing data storage:
    1. Address is now of type varchar(1000) instead of a ‘100-character string’.
    2. The start and end times of tours are now of type smalldatetime instead of UNIX timestamps.
    3. Ranger_grade is now of type tinyint instead of being a string.

Relational Diagram



Normalization

Criteria for 1NF

For each relation in the database:

- ➔ Each cell needs to be single valued (or **atomic**). It is clear from the above relational diagram that no tuple is having a set of values, a tuple of values, or a combination of both as an attribute value.
- ➔ In one column type, all entries are of the same form: clear from the relational database
- ➔ Rows are uniquely identifiable: check

∴ 1NF verified

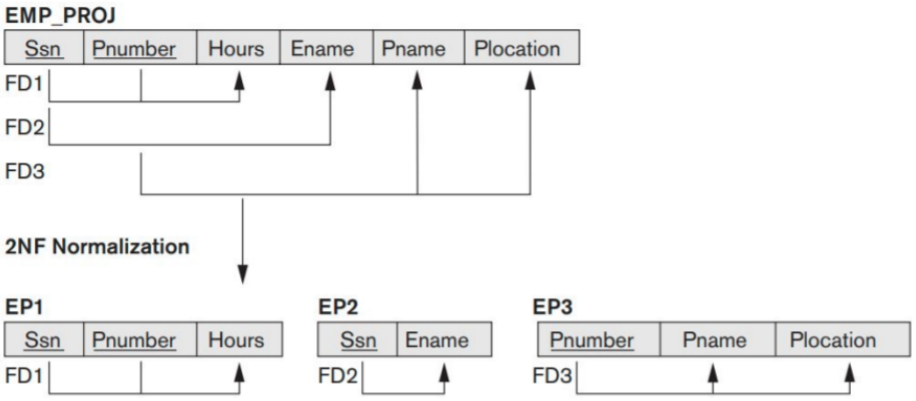
Criteria for 2NF

- ➔ Satisfies 1NF criteria: already seen in the previous point.

Definition: A relation schema R is in 2nd normal form if every non prime attribute A in R is **fully functionally dependent** on the primary key of R. Thus,

- ➔ For each relation in the database, every non-prime attribute should be dependent on the entire primary key and not a part of it.

Checking each relation in the database:



(fig. 14.11a from 'Fundamentals of Database Systems' by Elmasri and Navathe)

Relation	Form
Visitor	EP2
Visitor_PhNos	No non-prime attributes
Zone	EP3
Tour	No non-prime attributes
Vehicle	EP3
Organism	EP1
Org_CommonNames	No non-prime attributes
Population	EP1
PRESENT_IN	No non-prime attributes
Employee	EP3
Medic	EP2
CAN_TREAT	No non-prime attributes
Security	EP2
Ranger	EP2
ASSIGNED_TO	EP1

Hence, every relation visually corresponds to some kind of valid 2NF form.

∴ 2NF verified

Criteria for 3NF

Definition. According to Codd's original definition, a relation schema R is in 3NF if it satisfies 2NF and **no non prime attribute** of R is **transitively dependent** on the primary key.

- ➔ Satisfies 2NF criteria: already seen in the previous point.
- ➔ For all relations in the database, no non-prime attribute should depend on another non-prime attribute.

This does check out for the relations in the database, because owing to the forms as listed above, for any relation with non-prime attributes, only primary keys are on the LHS of any FD.
However, we did notice that in the table **Zone,** it *seems* that the composite of attributes *Longitude* and *Latitude* might describe any given zone and it seems that it is a violation of the second rule. But we note that both of them combined form a **candidate key** (thus making it no longer a non-prime attribute set). Hence the relational diagram satisfies all conditions of 3NF as well.

No changes had to be made to the original database proposition, it turns out our database was in 3rd normal form since the beginning.
∴ 3NF verified

Get to know the amazing music community and collaborate for covers.