**INTRODUCTION**

The hierarchical data model is the most traditional type of data model. It uses a tree-like structure to arrange data. Nodes in a hierarchical model are linked by branches. The topmost node is referred to as the root node. If several nodes occur at the top level, these are referred to as root segments. Each node has a single parent. A single parent can have multiple children.

The network data model is a more sophisticated variant of the hierarchical data model. Instead of a tree-structure, it uses guided graphs to arrange results. This child could have two or more parents. It employs the concepts of two data structures: records and collections.

The relational data model physical links as they are in the hierarchical data model. Only tables are used to display results. It is just concerned with the details and not with the physical structure. It contains metadata-related material. The tuple would have just one value at the intersection of row and column. It allows you to easily manage the questions.

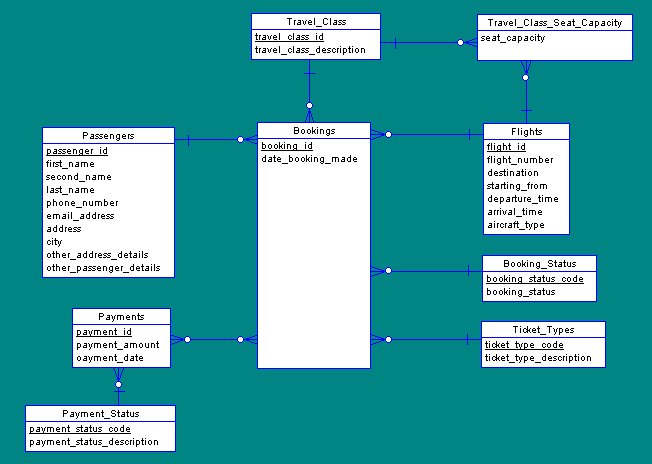
The Structured Query Language (SQL) is the industry standard for working with relational databases. Tables describe relationships in a relational database. SQL programming can be used to efficiently insert, browse, edit, and erase database information. That doesn't mean SQL can't do anything else. It is capable of a wide variety of functions, including, but not limited to, managing and sustaining databanks.

NoSQL is basically a non-relational database management scheme that does not need a static structure, does not require joins, and is easy to scale. NoSQL databases are typically used in distributed data stores with large data space requirements. NoSQL is utilized in big data and real-time smartphone devices. Every day, organizations like twitter, facebook, and google generate large amounts of user data. The acronym NoSQL stands for "Not Just SQL" or "Not SQL." Traditional relational database management systems (RDBMS) use SQL syntax to store and retrieve data for further analysis. A NoSQL database architecture, on the other hand, incorporates a broad variety of database technology capable of storing structured, semi-structured, unstructured, and polymorphic data.

**MODELLING**

**CONCEPTUAL**

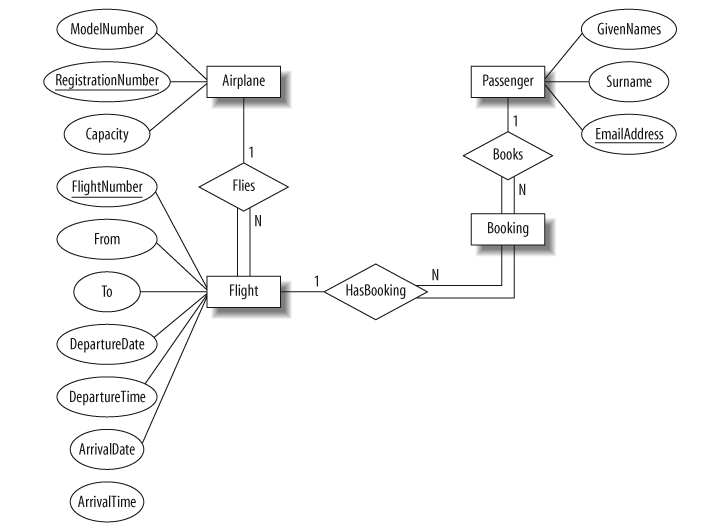
The entities in the in the airline scenario are Passenger, Flight, Employee, Job Description, Salary, Plane/Aircraft, Payment and Book Ticket/Ticket



**LOGICAL**

Normalization as it is a good database technique of coming up with a good relation database. For the case of the Mars Airline, the entities are split into smaller tables that are able to accommodate the various relationships. The primary requirements become the UNF. Further, 1NF is obtained by making the data atomic and a primary key assigned. Foreign key is also used to link other related tables in the scenario. 2NF is gotten by making sure that all the redundant attributes and fields are link by a primary key and foreign key. For the case of 3NF, the deal is to make sure that the information contained has no transitive dependencies.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **UNF** | **1NF** | **2NF** | **3NF** | **Entities** |
| **Passenger** | **Passenger** | **Passenger** | **Passenger** | **Passenger** |
| name | name | passportid PK | passportid PK |  |
| Address | address | name | name |  |
| phone\_number | phone\_number | address | address |  |
| email\_address | email\_address | phone\_number | phone\_number |  |
| dateofbirth | dateofbirth | email\_address | email\_address |  |
| **Flight** | passportid | dateofbirth | dateofbirth |  |
| flight number | nationality | nationality | nationality |  |
| origin | gender | gender | gender |  |
| destination | **Flight** | **Flight** | flight number FK |  |
| departure time | flight number | flight number PK | Planeid FK |  |
| arrival time | origin | origin | Paymentid FK |  |
| capacity | destination | destination | Ticketid FK |  |
| flightclass | boarding time | boarding time | Chekinid FK |  |
| **Employee** | departure time | departure time | **Flight** | **Flight** |
| name | arrival time | arrival time | flight number PK |  |
| address | flight status | flight status | origin |  |
| salary | flight length | flight length | numberofpassengers |  |
| identification\_number | flightclass | flightclass | numberof flights |  |
| flight information | flyingnumber | Employee | destination |  |
| Jobdescription | **Employee** | identification\_number PK | boarding time |  |
| Qualification | name | name | departure time |  |
| **Plane/Aircraft** | address | address | arrival time |  |
| Model | identification\_number | **JobDescription** | flight status |  |
| manufacturers | **JobDescription** | Jobid PK | flight length |  |
| planetype | jobrole | jobrole | flightclass |  |
| **Payment** | Jobdescription | Jobdescription | passportid FK |  |
| Paymenttype | jobqualification | jobqualification | identification\_number FK |  |
| Paymentdate | **Salary** | **Salary** | Jobid FK |  |
| Currency | Currency | Salaryid PK | Salaryid FK |  |
| Amount | Salary payment method | Currency | Planeid FK |  |
| **Book\_Ticket/Ticket** | Amount | Salary payment method | Paymentid FK |  |
| Ticketdate | Frequency | Amount | Ticketid FK |  |
| Ticketdescription | Taxcode | Frequency | Chekinid FK |  |
| Seat | Paymentdate | Taxcode | **Employee** | **Employee** |
| Flyingnumber | Holidaypayment | Paymentdate | identification\_number PK |  |
| Tickettype | bonus | Holidaypayment | **name** |  |
| Luggageweight | **Plane/Aircraft** | bonus | **address** |  |
|  | Model | **Plane/Aircraft** | flight number FK |  |
|  | manufacturers | Planeid PK | Jobid FK |  |
|  | planetype | Model | Salaryid FK |  |
|  | capacity | manufacturers | Planeid FK |  |
|  | **Payment** | planetype | **JobDescription** | **JobDescription** |
|  | Paymenttype | capacity | Jobid PK |  |
|  | Paymentdate | **Payment** | jobrole |  |
|  | Currency | Paymentid PK | Jobdescription |  |
|  | Amount | Paymenttype | Jobqualification |  |
|  |  | Paymentdate | flight number FK |  |
|  |  | Currency | identification\_number FK |  |
|  |  | Amount | Salaryid FK |  |
|  |  | **Book\_Ticket/Ticket** | Planeid FK |  |
|  |  | Ticketid PK | **Salary** | **Salary** |
|  |  | Ticketdate | Salaryid PK |  |
|  |  | Ticketdescription | Currency |  |
|  |  | seatreservation | Salary payment method |  |
|  |  | Tickettype | Amount |  |
|  |  | Quantity | Frequency |  |
|  |  | **CheckIn Table** | Taxcode |  |
|  |  | Chekinid PK | Paymentdate |  |
|  |  | checkindate | Holidaypayment |  |
|  |  |  | Bonus |  |
|  |  |  | flight number FK |  |
|  |  |  | identification\_number FK |  |
|  |  |  | **Plane/Aircraft** | **Plane/Aircraft** |
|  |  |  | Planeid PK |  |
|  |  |  | Model |  |
|  |  |  | manufacturers |  |
|  |  |  | planetype |  |
|  |  |  | capacity |  |
|  |  |  | passportid FK |  |
|  |  |  | flight number FK |  |
|  |  |  | identification\_number FK |  |
|  |  |  | Jobid FK |  |
|  |  |  | Salaryid FK |  |
|  |  |  | Paymentid FK |  |
|  |  |  | Ticketid FK |  |
|  |  |  | Chekinid FK |  |
|  |  |  | **Payment** | **Payment** |
|  |  |  | Paymentid PK |  |
|  |  |  | Paymenttype |  |
|  |  |  | Paymentdate |  |
|  |  |  | Currency |  |
|  |  |  | Amount |  |
|  |  |  | Passportid FK |  |
|  |  |  | flight number FK |  |
|  |  |  | Planeid FK |  |
|  |  |  | Ticketid FK |  |
|  |  |  | Chekinid FK |  |
|  |  |  | **Book\_Ticket/Ticket** | **Book\_Ticket/Ticket** |
|  |  |  | Ticketid PK |  |
|  |  |  | Ticketdate |  |
|  |  |  | Ticketdescription |  |
|  |  |  | seatreservation |  |
|  |  |  | Tickettype |  |
|  |  |  | Quantity |  |
|  |  |  | Passportid FK |  |
|  |  |  | flight number FK |  |
|  |  |  | Planeid FK |  |
|  |  |  | Paymentid FK |  |
|  |  |  | Chekinid FK |  |
|  |  |  | CheckIn Table |  |
|  |  |  | Chekinid PK |  |
|  |  |  | Checkindate |  |
|  |  |  | Passportid FK |  |
|  |  |  | flight number FK |  |
|  |  |  | Planeid FK |  |
|  |  |  | Paymentid FK |  |
|  |  |  | Ticketid FK |  |



**PHYSICAL**

