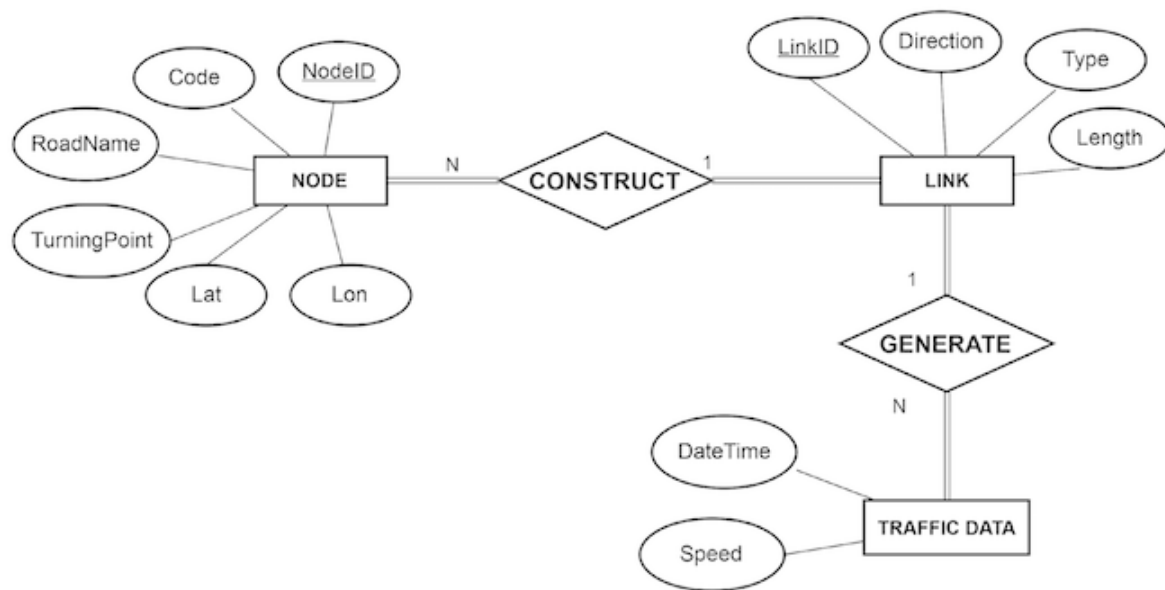


2.3 Interacting With Data (RDB)

Database

- Computer based record keeping system
 - Record keeping system: A manual or automated system that collects, organise and categorise records, facilitating their preservation, retrieval, use and disposition
- Collection of data to meet an organisation's information needs
- Computer files that are optimised to store data in a structured way
- Organisation of files helps to ensure that data is accurate
- Properties of a relational table:
 1. Values are atomic: columns are non-repeating
 2. Each row is unique
 3. Column values are of the same kind: all values come from the same domain (i.e. integers/strings)
 4. Sequence of columns is insignificant
 5. Sequence of columns is insignificant
 6. Each column has a unique name
- Examples:
 - Library system:
 - ◆ Location of books
 - ◆ Students' and teachers' details
 - ◆ Loaning history
 - Student system
 - ◆ Students' particulars
 - ◆ Results and grades
 - Resource booking system
 - ◆ Booking of venues and facilities
 - ◆ Details of booking: location, time slot
 - Human resource system
 - ◆ Workers' details
 - ◆ Leave information
 - ◆ Payroll
 - Inventory system
 - ◆ Items in warehouse
 - ◆ Stock level
- **Flat-file Database:** Holds all data in an unstructured table

- Each file consists a table of related information defined by Rows and Columns
 - ◆ Rows: "Records"
 - ◆ Columns: "Fields"
- Drawbacks:
 - ◆ Duplicated data is unnecessarily entered
 - ◆ Database space is wasted with this duplicated data
 - ◆ Duplicated data takes a long time to enter and update (unnecessary)
 - ◆ Data redundancy
- **Relational Database:** Stores data and how the data are related
 - Organises data into one or more tables
 - Each table represents one entity type
 - Reduces redundancy
 - Types of relationship:
 - ◆ One to one
 - ◆ One to many
 - ◆ Many to many
 - Entity Relationship Diagram (ER):
 - ◆ Visual representation of different data using conventions that describe how these data are related
 - ◆ Shows structure of data in a database
 - ◆ Building blocks:
 - ◆ Entity: Rectangle
 - ◆ Relationship: Diamond
 - ◆ Fields: Circles (not required)
 - ◆ Shows 1 to 1, 1 to many, many to many relationships between entities



○ Shorthand Notation:

- ◆ E.g. Student (Student ID, Name, NRIC, Address, Telephone)
- ◆ Key Field / Primary Key:
 - ◆ Underlined (Student ID)
 - ◆ Unique, used to identify a particular Record
- ◆ Composite Key:
 - ◆ Primary Key that consists more than one field
 - ◆ E.g. Name + Surname
- ◆ Candidate Keys:
 - ◆ Table may have one or more choices for primary key
 - ◆ Collectively known as candidate key
 - ◆ Table can choose which key to be used as primary key, other key becomes secondary key
- ◆ Foreign key:
 - ◆ A column/columns in a table that draws values from a primary key in another table
 - ◆ Assists in ensuring the data integrity of a table
 - ◆ Not required to be unique

○ Normalisation:

- ◆ To process data to more efficiently manage data
- ◆ To isolate data so that additions/deletions/modifications of a field can be made in just one table and then propagated through the rest of the database using defined relationships

- ◆ First Normal Form:
 - ◆ Make sure that any attributes with multiple values are removed so that records are of the same length
- ◆ Second Normal Form (assume relation in first normal form):
 - ◆ All attributes in an entity must be functionally dependent (unique association) with the primary key for the purpose of identification
 - ◆ Each field is unique to its key field
- ◆ Third Normal Form (assume relation in second normal form):
 - ◆ There is no transitive functional dependency
 - ◆ All non-key attributes are fully functional dependent only on the primary key
 - ◆ There isn't an extra table to describe relationship between the non-key attributes