Data modelling in software engineering is the process of creating a conceptual representation of data structures that are used in a software system. These representations help in understanding the organization, relationships, and constraints of the data within the system. Here's a breakdown of data modeling:

1. **Conceptual Data Model**: This represents high-level concepts and relationships between different data entities without delving into technical details. It provides a broad overview of the data elements and their associations. Common techniques for conceptual data modeling include Entity-Relationship Diagrams (ERDs) and Unified Modeling Language (UML) class diagrams.
2. **Logical Data Model**: The logical data model translates the conceptual model into a more detailed representation that is closer to the actual implementation. It defines the structure of the data, including entities, attributes, relationships, and constraints. The logical data model is often expressed using a standardized notation such as SQL (Structured Query Language) or UML class diagrams.
3. **Physical Data Model**: The physical data model represents how the data will be stored and organized in the underlying database management system (DBMS). It includes details such as data types, indexing, partitioning, and optimization considerations. Physical data modeling is closely tied to the specific characteristics and requirements of the chosen DBMS.
4. **Normalization**: Normalization is a technique used to minimize redundancy and dependency in a database schema. It involves organizing the data into multiple tables and establishing relationships between them to reduce data duplication and improve data integrity. Normal forms, such as First Normal Form (1NF), Second Normal Form (2NF), and Third Normal Form (3NF), are commonly used guidelines for normalization.
5. **Denormalization**: Denormalization is the opposite of normalization and involves intentionally introducing redundancy into the database schema to improve performance, especially for read-heavy workloads. It can be used judiciously in situations where the overhead of normalization outweighs the benefits.
6. **Data Modeling Tools**: Various software tools are available to assist in the process of data modeling. These tools provide features for creating, visualizing, and managing data models, as well as generating database schemas and documentation. Popular data modeling tools include ERwin, Microsoft Visio, Lucidchart, and MySQL Workbench.

Data modeling is a critical aspect of software engineering, as it lays the foundation for the design, implementation, and maintenance of database systems. By creating clear and well-defined data models, software engineers can ensure that the software system effectively captures, stores, and retrieves the necessary data to support its functionality and business requirements.