ERD modeling levels

Models can be created at different levels of abstraction, entity-relationship models too. The following modeling levels can be distinguished:

- conceptual (descriptive) level
- logical (analytical) level
- physical (executive) level

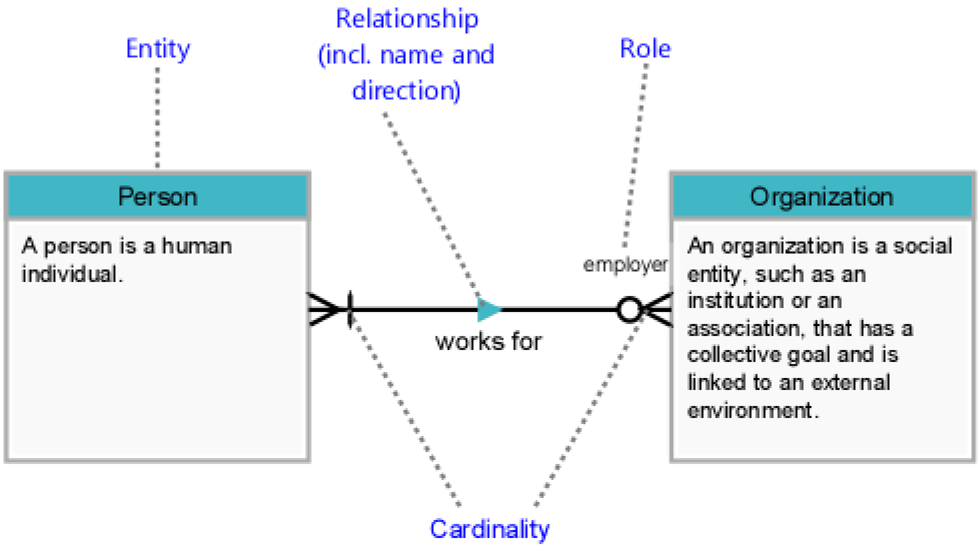
A conceptual model captures the business scope of the problem, the logical model captures the business solution, and the physical model the technical solution.
Enterprise Studio supports creating ER models on the conceptual and logical level.

Conceptual level

A model on the conceptual level captures the business scope of the problem. It displays the most important entities in business terms, as well as their interrelationships.

In a conceptual view, a definition of the entities can be included, and relations can have a cardinality. The cardinality of a relation describes the entity's involvement in the relation. The cardinality is expressed via a combination of participation (optionality) and multiplicity and is expressed in terms of one-on-one, one-to-many, many-to-one, or many-to-many.

Relations are defined by giving them a short description, with an arrow on the line indicating the direction of the relation. At both ends of the relation line, the two roles of the relation can be displayed.

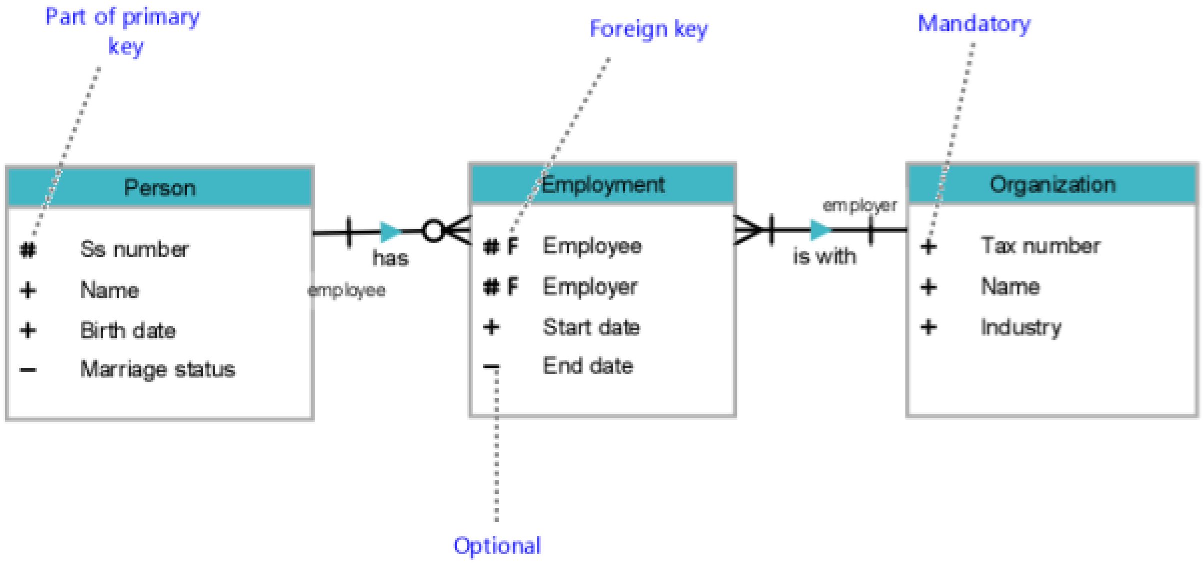

Entity
Relationship (incl. name and direction)
Role
Person
A person is a human individual.
employer
Organization
An organization is a social entity, such as an institution or an association, that has a collective goal and is linked to an external environment.
works for
Cardinality

Example of a conceptual model

In the example model above two entities are shown: *Person* and *Organization*. Between them is the relation *works for* going from *Person* to *Organization*. For the organization, the role name *employer* is included. The following can be read by the cardinality: "Person works for zero or more Organizations" and "Organization employs one or more Persons".

Logical level

A model on a logical level is a further elaboration of the conceptual model. In addition to the information recorded in a conceptual model, other information is recorded. The entities can be further elaborated by adding attributes. For attributes the optionality is indicated; are they mandatory or optional. It is also indicated whether they are part of the *primary key* of the entity, or that they are a *foreign key*.



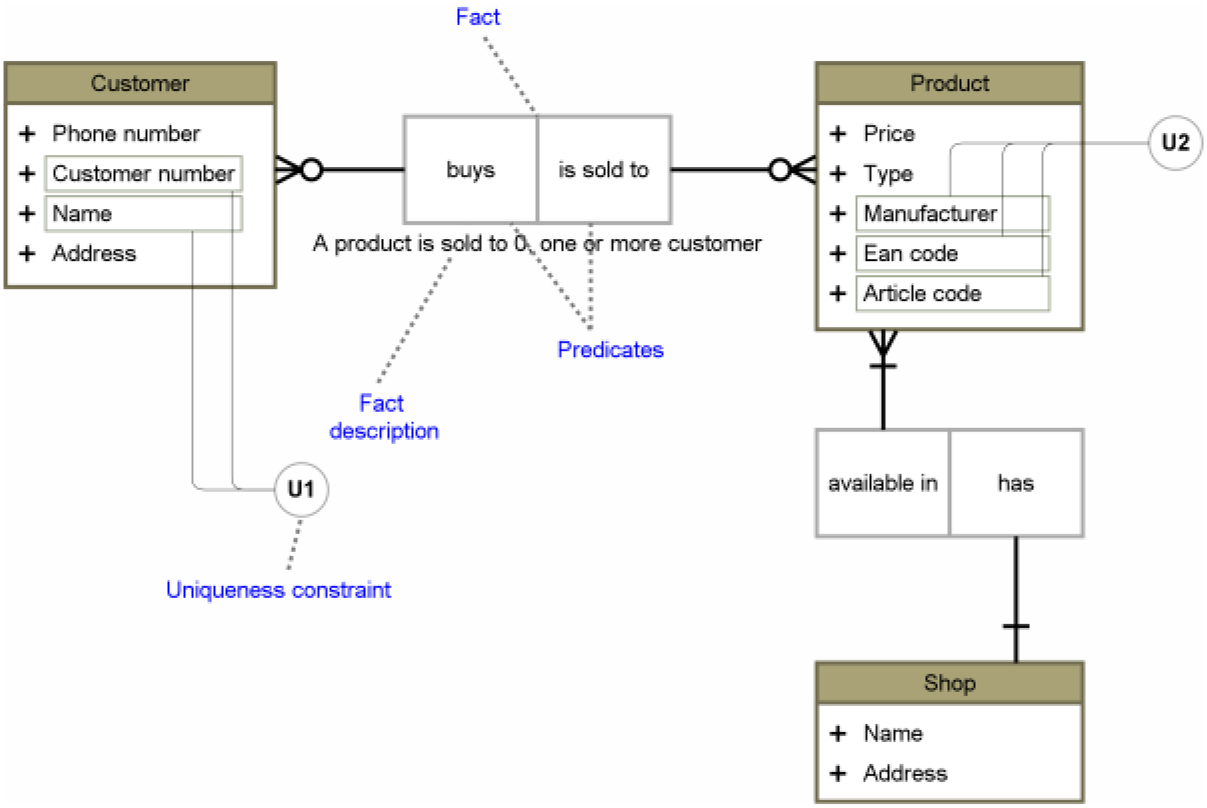
Example of a logical model

In the example model above, an extra entity *Employment* had been included with respect to the conceptual example model. With this entity, the relation between *Person* and *Organization* is further elaborated. Not only the cardinality is more detailed, but the properties (attributes) can also be recorded in detail by indicating the optionality and key for each attribute.

NIAM-inspired modeling

As an addition to modeling on a logical level, a NIAM-like (Natural language Information Analysis Method) view is available for entity-relationship modeling. It is a variation of the Crow's Foot notation inspired by NIAM (*Conceptual Schema and Relational Database Design* by Sjir Nijssen & Terry Halpin (1989)).

The notion of fact (type) is used to model relationships. A fact consists of two predicates, each defining the role of one of the entities involved in the relationship.



Example of a NIAM-inspired logical model

In the above example model the top most fact defines the role "buys" for the customer in relation to the product, and the role "is sold to" for the product in relation to the customer. This gives the following predicates "A customer buys 0, one or more product" (displayed below) and "A product is sold to 0, one or more customer".

Additionally, different types of constraints (uniqueness, exclusion, completeness) can be modeled to indicate limitations in the model.

Introductory eLearning course

Check out the free Enterprise Studio introductory eLearning course to learn about ERD modeling in Enterprise Studio.

View course