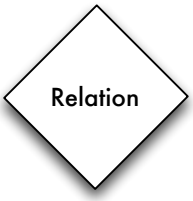
Overview of the (E)ER Model Notation

Based on the book by Saake, Sattler, Heuer "Datenbanken: Konzepte und Sprachen" 5.Auflage, mitp-Verlag, 2013

Basic Notation Elements

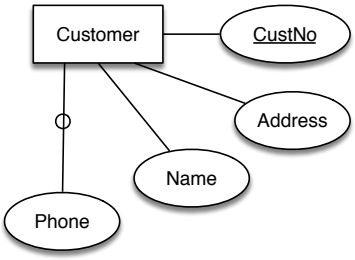
Entity types are the units of information represented by a database.
Example: A database stores information about customers *and* products.



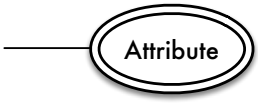
Relationship types describe relationships between entities.
Example: Customers buy *products.*

Attributes represent properties of entities or relationships.
Example: A customer has a name.

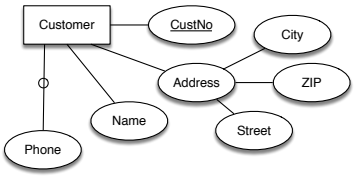
Primary Keys (Key Attributes) are attributes that uniquely identify an entity of an entity type.
Example: The customer ID *uniquely identifies each customer.*



Optional Attributes represent properties of entities or relationships that may exist, but don't need to.
Example: A customer may specify his phone number*, but he doesn't have to.*



Multi-Value Attributes may be specified multiple times.
Example: A customer is able to have multiple phone numbers.

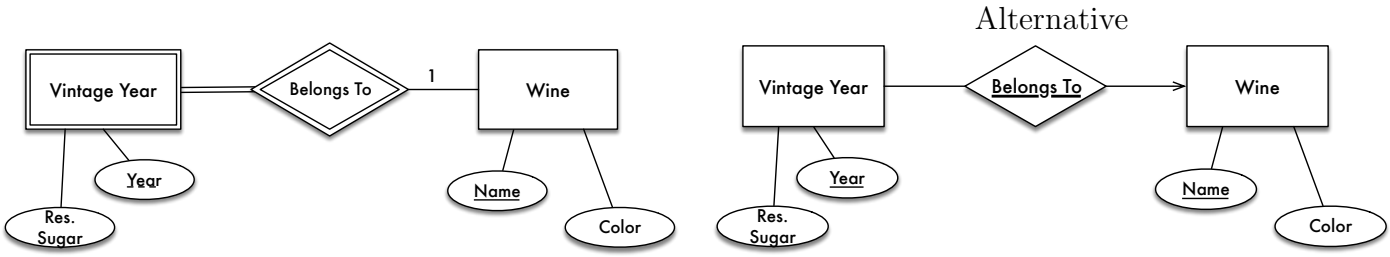


Structured Attributes consist of multiple sub-attributes.
Example: The customer address *consists of street, zip code and city.*

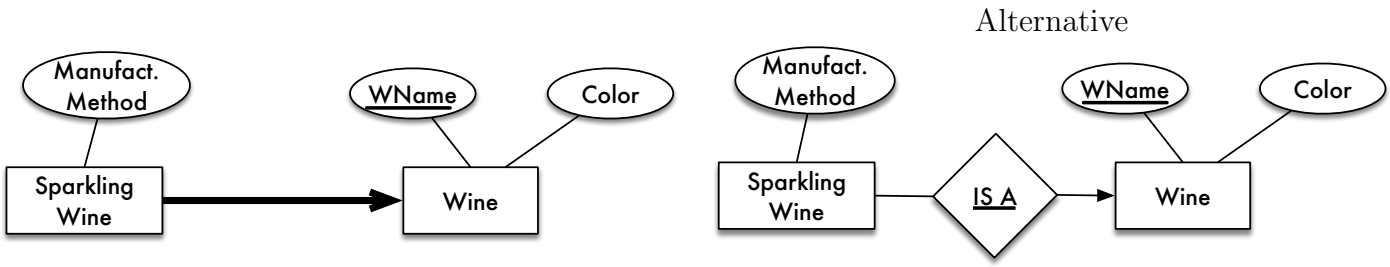
Derived Attributes can be derived from other attributes.
Example: The retail price of a product is calculated from the net price and additional taxes.

Authors: David Broneske, Sebastian Dorok, Florian Koch

Depending or Weak Entities are functionally determined by another entity. The dependent entity type doesn't have any attributes that may be used to uniquely identify an entity of this type. A primary key, or respectively a set of attributes forming one, always consists of the key attributes of the entity type it depends on, too.
Example: A vintage always belongs to a wine, since it has a specific residual sweetness. If the wine doesn't exists, the vintage doesn't exist, either. But, since there are multiple wines with a vintage in the same year, only the combination of wine name and vintage year are able to uniquely identify a vintage.



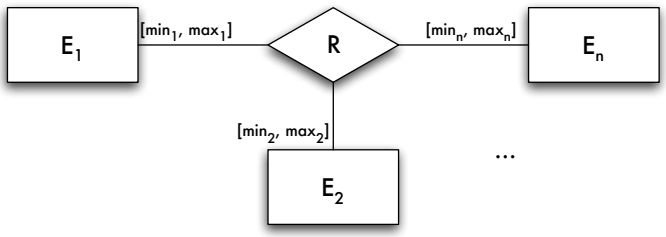
IS-Relationships, also called specialization-/generalization-relationships, semantically correspond to injective functional relationships.
Example: A sparkling wine is a special wine. Through this definition the sparkling wine inherits the attributes name and color of a wine.



Cardinalities

Cardinalities describe, how often entities of an entity type may have a relation to another entity.
Example: A specific product is delivered by a specific supplier. Therefore, each product has a relationship to exactly one supplier. On the other hand, a supplier may offer multiple or no products at all. In the course of this, a supplier can have a relationship with multiple products.

$[min, max]$ notation: Minimal and maximum quantity is specified.

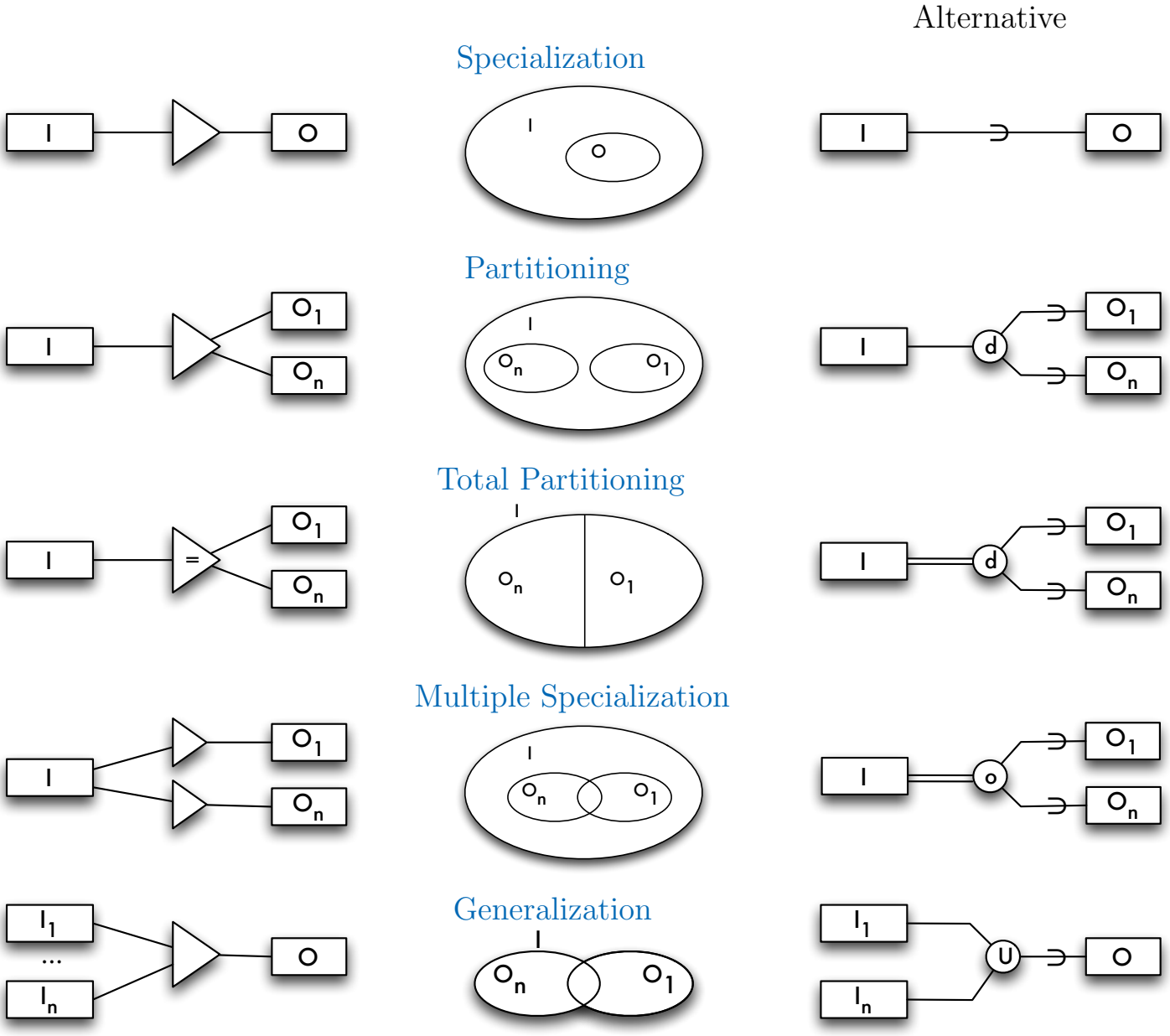


Beispiel

Authors: David Broneske, Sebastian Dorok, Florian Koch

Alternative representation: *Note the "reversed" assignment of the cardinalities!*

Enhanced/Extended Notation Elements



Authors: David Broneske, Sebastian Dorok, Florian Koch