**Cardinality & Ordinality**

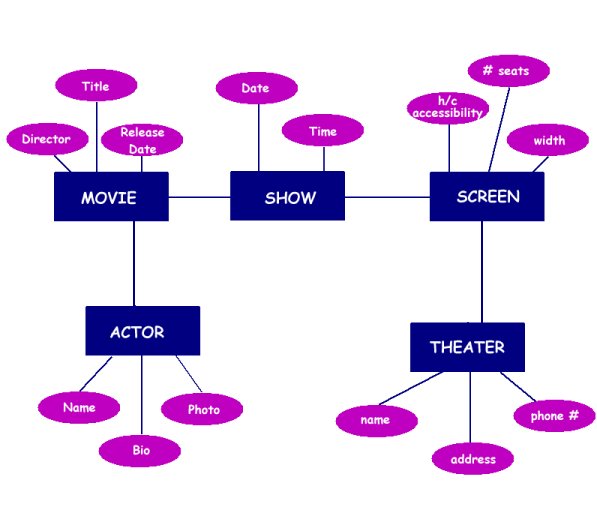
***Entity-relationship diagram*** also called an *entity-relationship model*, a graphical representation of entities and their relationships to each other, typically used in computing in regard to the organization of [data](http://www.webopedia.com/TERM/e/data.html) within [databases](http://www.webopedia.com/TERM/e/database.html) or information systems. An entity is a piece of data—an [object](http://www.webopedia.com/TERM/e/object.html) or concept about which data is stored. A relationship is how the data is shared between entities. There are three types of relationships between entities :

* **One-to-one**: one instance of an entity (A) is associated with one other instance of another entity (B). For example, in a database of employees, each employee name (A) is associated with only one social security number (B).
* **One-to-many**: one instance of an entity (A) is associated with zero, one or many instances of another entity (B), but for one instance of entity B there is only one instance of entity A. For example, for a company with all employees working in one building, the building name (A) is associated with many different employees (B), but those employees all share the same singular association with entity A.
* **Many-to-many**: one instance of an entity (A) is associated with one, zero or many instances of another entity (B), and one instance of entity B is associated with one, zero or many instances of entity A. For example, for a company in which all of its employees work on multiple projects, each instance of an employee (A) is associated with many instances of a project (B), and at the same time, each instance of a project (B) has multiple employees (A) associated with it.

These three relationships are also called **cardinality.**

**Ordinality** is also closely linked to cardinality. While cardinality specifies the occurrences of a relationship, ordinality describes the relationship as either mandatory or optional. In other words, cardinality specifies the maximum number of relationships and ordinality specifies the absolute minimum number of relationships.

**Generally, ERD's looks like this**:

****

There are three basic elements in ER models:

* Entities are the "things" about which we seek information.
* Attributes are the data we collect about the entities.
* Relationships provide the structure needed to draw information from multiple entities.

### Entity Relationship Diagram Notations

[Peter Chen](http://www.csc.lsu.edu/~chen/chen.html) developed ERDs in 1976. Since then Charles Bachman and James Martin have added some slight refinements to the basic ERD principles.

##### **Entity**

An entity is an object or concept about which you want to store information.

Entity

##### **Weak Entity**

A weak entity is an entity that must defined by a foreign key relationship with another entity as it cannot be uniquely identified by its own attributes alone.

Weak Entity

##### **Key attribute**

A key attribute is the unique, distinguishing characteristic of the entity. For example, an employee's social security number might be the employee's key attribute.

Key attribute

##### **Multi-valued attribute**

A multi-valued attribute can have more than one value. For example, an employee entity can have multiple skill values.

Multivalued attribute

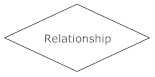
##### **Derived attribute**

A derived attribute is based on another attribute. For example, an employee's monthly salary is based on the employee's annual salary.

Derived attribute

##### **Relationships**

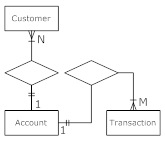
Relationships illustrate how two entities share information in the database structure.



##### **Cardinality**

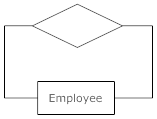
Cardinality specifies how many instances of an entity relate to one instance of another entity.

Ordinality is also closely linked to cardinality. While cardinality specifies the occurences of a relationship, ordinality describes the relationship as either mandatory or optional. In other words, cardinality specifies the maximum number of relationships and ordinality specifies the absolute minimum number of relationships.



##### **Recursive relationship**

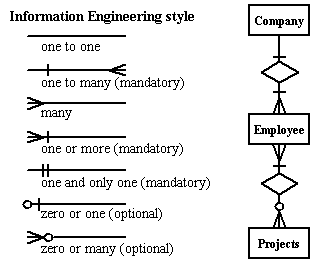
In some cases, entities can be self-linked. For example, employees can supervise other employees.



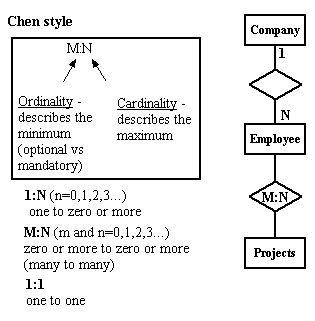
Top of Form

## There are many notation styles that express cardinality

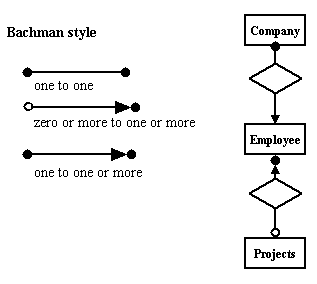
##### Information Engineering



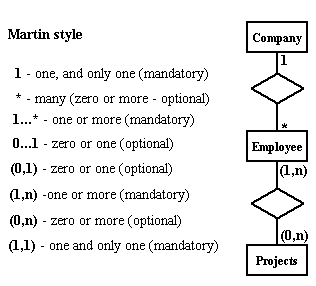
##### Chen



##### Bachman

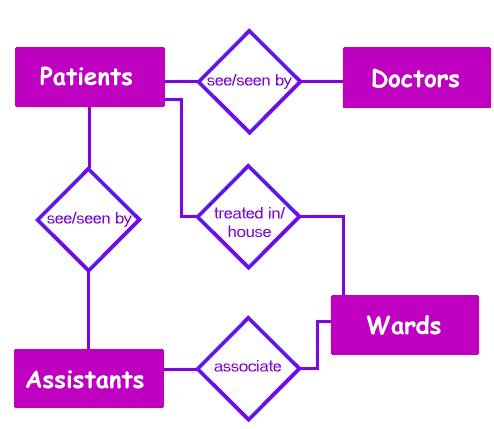


##### Martin



**Reading an ERD**: It takes some practice reading an ERD, but they can be used with clients to discuss business rules. These allow us to represent the information from above such as the E-R Diagram below:

Cardinality Notations - Entity Relationship Diagrams - SmartDraw Tutorials/resources/tutorials/Cardinality-NotationsEntity\_Relationship\_Di...Cardinality-Notations.ascx

****

**How do we start an ERD?**

* Define Entities: these are usually nouns used in descriptions of the system, in the discussion of business rules, or in documentation; identified in the narrative (see highlighted items above).
* Define Relationships: these are usually verbs used in descriptions of the system or in discussion of the business rules (entity \_\_\_\_\_\_ entity); identified in the narrative (see highlighted items above).
* Add attributes to the relations; these are determined by the queries,and may also suggest new entities, e.g. grade; or they may suggest the need for keys or identifiers.

**What questions can we ask?**

* Which doctors work in which wards?
* How much will be spent in a ward in a given week?
* How much will a patient cost to treat?
* How much does a doctor cost per week?
* Which assistants can a patient expect to see?
* Which drugs are being used?

# Add cardinality to the relations

* Many-to-Many must be resolved to two one-to-manys with an additional entity
* Usually automatically happens
* Sometimes involves introduction of a link entity (which will be all foreign key)