DBMS - Ch.3 Enhanced ER Model

★ Supertype & Subtype Representation

SUBTYPE - a subgrouping of the entities in an entity type.

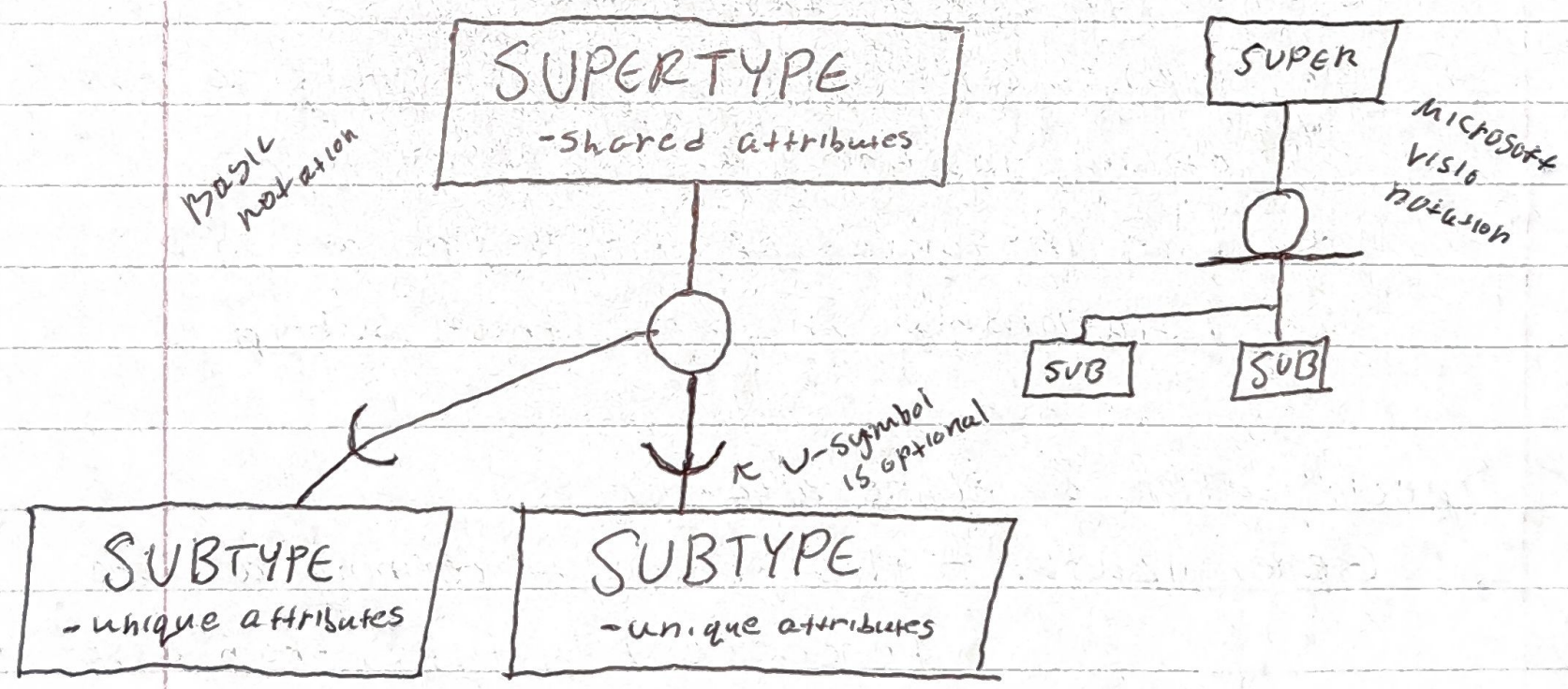
Ex. entity type: STUDENT

subtypes: GRADUATE, UNDERGRAD

SUPERTYPE - a generic entity that has a relationship with 1+ subtypes.

Ex. STUDENT is a supertype in the ~~previous~~ previous example.

★ Basic Concepts & Notation



☆ Attribute Inheritance

- An entity instance of a subtype represents the same entity instance of the supertype.

EMPLOYEE —O→ CONSULTANT

- If Jonas Daniels is an occurrence of CONSULTANT, it's also an occurrence of EMPLOYEE

Attribute Inheritance- subtype entities inherit values of all attributes & instances of all relationships of the supertype.

☆ When to Use SUPER/SUBTYPE Relationships

Either or both conditions...

1. There are attributes that apply to *some*, but not all, instances of an entity type.
2. The instances of a subtype are in a relationship unique to that subtype

☆ Specialization & Generalization

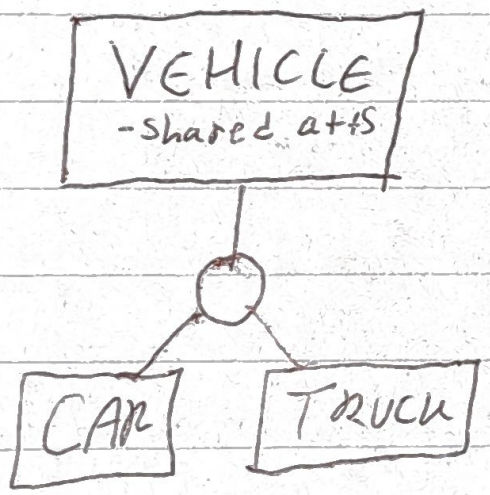
Generalization - the process of defining a more general entity type from a set of more specialized entity types.

- bottom-up process

DBMS - Ch. 3 pt. 2 Enhanced ER Model

★ Generalization & ~~Spec~~ Specialization pt. 2

SU Supertypes, CAR, TRUCK & MOTORCYCLE can be connected to the general entity type VEHICLE if they share common attributes.



Specialization - process of defining a subtype of the supertype & forming super/subtype relationships.

- top-down process

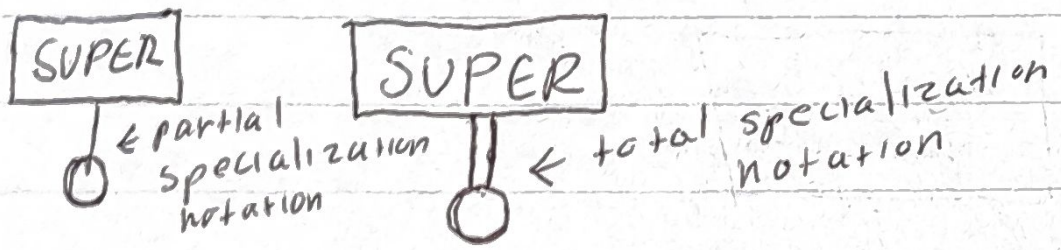
★ Specifying Constraints

Completeness Constraint - addresses whether an instance of a supertype should also be an instance of it subtypes.

- 2 rules: partial & total ~~spec~~ specialization

partial specialization- an entity instance of a supertype is allowed not to be an instance of any subtype.

total specialization- each entity instance of a supertype must be an instance of some subtype in the relationship.



Disjointness Constraint- addresses whether an instance of a supertype may simultaneously be a member of more than 1 subtype.

- 2 rules: disjoint rule & overlap rule

Disjoint Rule- an instance can only belong to 1 subtype.

Overlap Rule- an instance can belong to multiple subtypes.