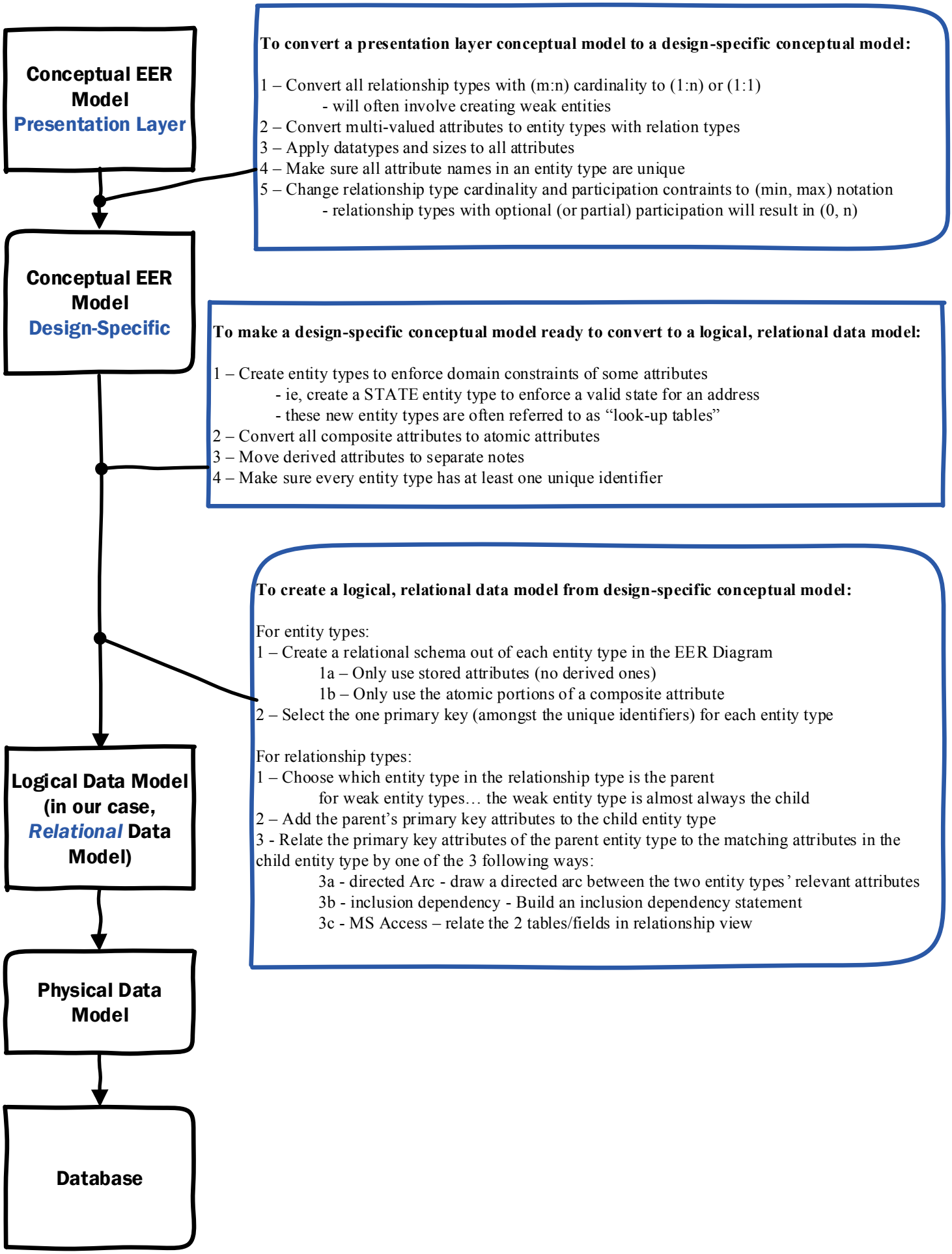
Conceptual EER Model Presentation Layer
To convert a presentation layer conceptual model to a design-specific conceptual model:
1 – Convert all relationship types with (m:n) cardinality to (1:n) or (1:1)
- will often involve creating weak entities
2 – Convert multi-valued attributes to entity types with relation types
3 – Apply datatypes and sizes to all attributes
4 – Make sure all attribute names in an entity type are unique
5 – Change relationship type cardinality and participation contraints to (min, max) notation
- relationship types with optional (or partial) participation will result in (0, n)
Conceptual EER Model Design-Specific
To make a design-specific conceptual model ready to convert to a logical, relational data model:
1 – Create entity types to enforce domain constraints of some attributes
- ie, create a STATE entity type to enforce a valid state for an address
- these new entity types are often referred to as "look-up tables"
2 – Convert all composite attributes to atomic attributes
3 – Move derived attributes to separate notes
4 – Make sure every entity type has at least one unique identifier
To create a logical, relational data model from design-specific conceptual model:
For entity types:
1 – Create a relational schema out of each entity type in the EER Diagram
1a – Only use stored attributes (no derived ones)
1b – Only use the atomic portions of a composite attribute
2 – Select the one primary key (amongst the unique identifiers) for each entity type
For relationship types:
1 – Choose which entity type in the relationship type is the parent
for weak entity types… the weak entity type is almost always the child
2 – Add the parent's primary key attributes to the child entity type
3 - Relate the primary key attributes of the parent entity type to the matching attributes in the child entity type by one of the 3 following ways:
3a - directed Arc - draw a directed arc between the two entity types' relevant attributes
3b - inclusion dependency - Build an inclusion dependency statement
3c - MS Access – relate the 2 tables/fields in relationship view
Logical Data Model (in our case, Relational Data Model)
Physical Data Model
Database