

1. ER diagram for the bank database.

Answers:

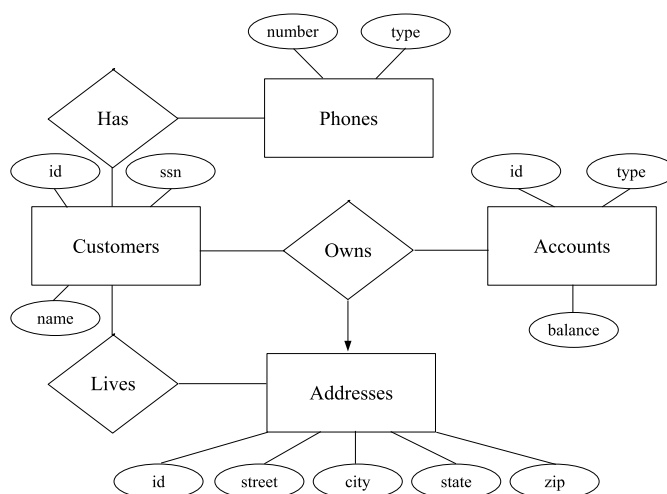


Figure 1: Bank ER Diagram

For some reason Open Office did not export the underlines of the key attributes, but the key attribute(s) for each entity set in the diagram should be fairly obvious.

2. Compare Figure 1 to Figure 2.14 in the textbook.

Answers:

The biggest problem about the design in Figure 2.14 is the **AcctSets** entity set, which is redundant because all the information about accounts are already stored in the **Accounts** entity set. Note that **AcctSets** does have an extra attribute **owner-address**, but it is again redundant because the address information is already stored in the **Addresses** entity set. The design in Figure 1 is much better in the sense that it keeps information about different entities (customers, accounts, and addresses) in their respective entity sets, and handles the ownership of the accounts using one relationship **Owns** instead of two relationships and one extra entity set as in Figure 2.14 of the textbook.

The design in Figure 2.14 also has a few other problems. In particular,

- it does not identify key attribute(s) in entity set.
- various components of an address, e.g. phone number, street, city, state, and zip, are lumped together in one attribute **address**.

- Assume that each customer may have multiple phones and multiple addresses, **Lives-at** should be a many-to-many relationship, and there should be a **Phone** entity set.

3. Convert the bank ER diagram to relations.

Answers:

```
Customers( id, ssn, name )
Phone( number, type )
Addresses( id, street, city, state, zip )
Accounts ( id, type, balance )
Owns( customer_id, account_id, address_id )
Has( customer_id, phone_number )
Lives( customer_id, address_id )
```

4. Convert the ER diagram in Figure 3.15.

Answers:

This problem may look complex, but it's actually quite straightforward if you realize two things: a) the three conversion methods (OO, ER, and NULL) *apply to only subclass entity sets and have nothing to do with relationships*, and b) subclass entity sets get their key attributes from the superclass entity set.

So the relationships can be converted to the following relations regardless of what conversion method is used:

```
ChildOf( name1, address1, name2, address2 )
FatherOf( name1, address1, name2, address2 )
MotherOf( name1, address1, name2, address2 )
Married( name1, address1, name2, address2 )
```

As for the entity sets, here are the relations using different conversion methods:

- NULL: Person
- ER: Person, Child, Father, Mother
- OO: Person, Child, Father, Mother, ChildFather, ChildMother

Also note that the many-to-one and one-to-one relationships can be merged with the relations converted from the entity sets on the “many” side.

5. Exercise with FDs and closures.

Answers:

- $\{A, E\}^+ = \{A, E, B, D\}$
- $\{A, C\}, \{B, C\}, \{D, C\}, \{E, C\}$
- $D \rightarrow A$ and $AB \rightarrow D$,