Database Management System (DBMS)

Lecture-4

Dharmendra Kumar

August 9, 2020

Network model

A network database consists of a collection of records connected to one another through links. A record is in many respects similar to an entity in the E-R model. Each record is a collection of fields (attributes), each of which contains only one data value. A link is an association between precisely two records. Thus, a link can be viewed as a restricted (binary) form of relationship in the sense of the E-R model.

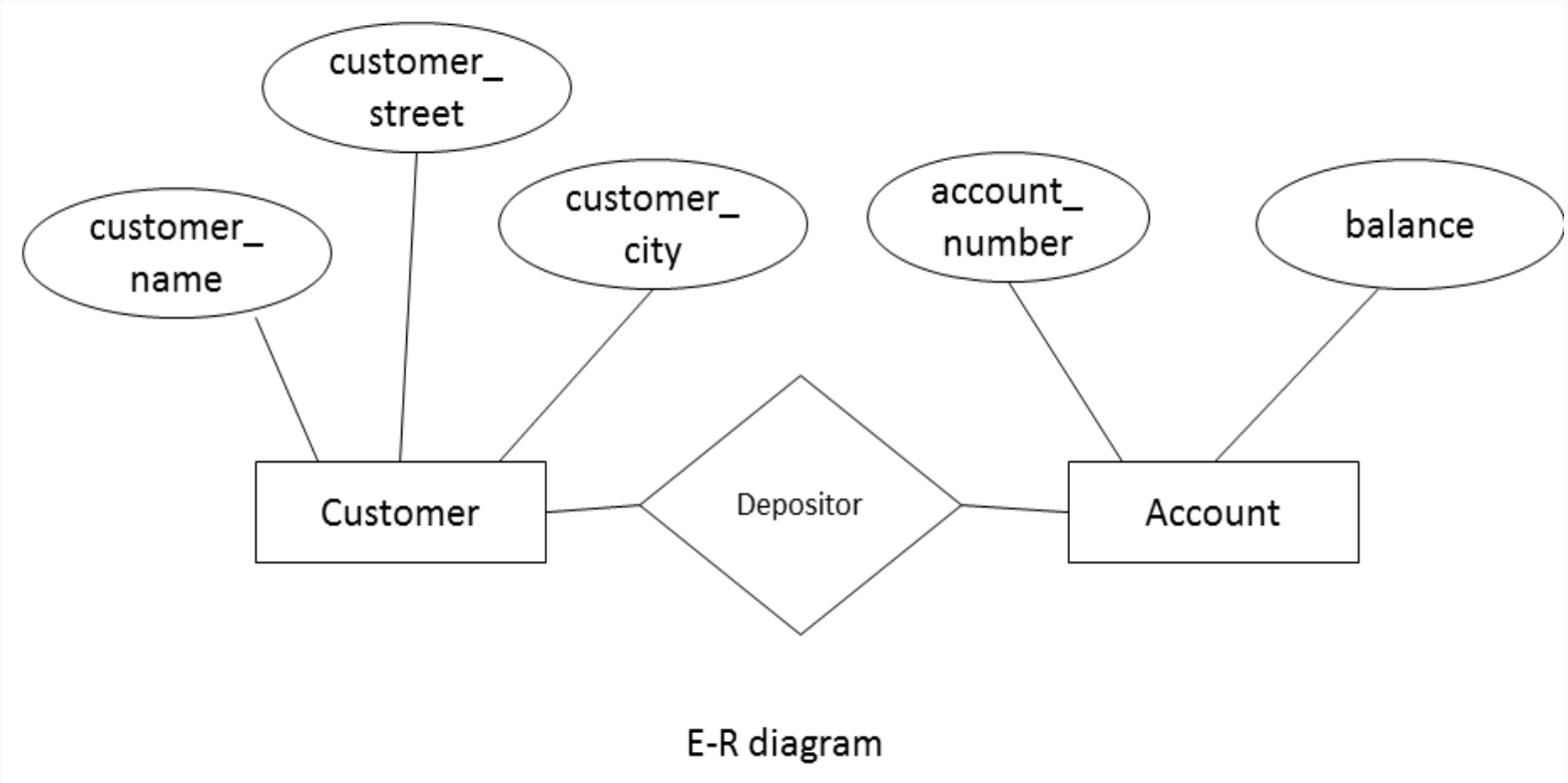
A schema representing the design of a network database is said to be Data-structure diagram. Such a diagram consists of two basic components:

1. Boxes, which correspond to record types
2. Lines, which correspond to links

A data-structure diagram serves the same purpose as an E-R diagram; namely, it specifies the overall logical structure of the database. Every E-R diagrams can be transformed into their corresponding data-structure diagrams.

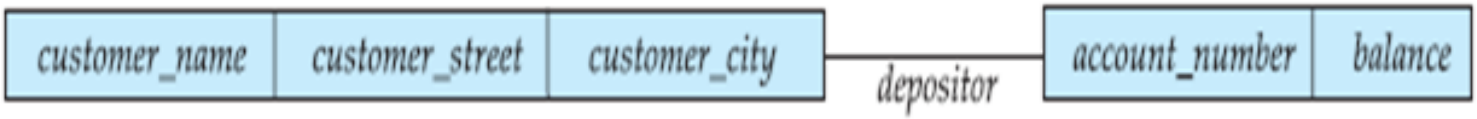
Example: Consider the following E-R diagram:-

This E-R diagram is consisting of two entity sets, customer and account, related through a binary, many-to-many relationship depositor, with no descriptive attributes. This diagram specifies that a customer may have several accounts, and that an account may belong to several different customers.



E-R diagram

The data-structure diagram for the corresponding E-R diagram is the following:-

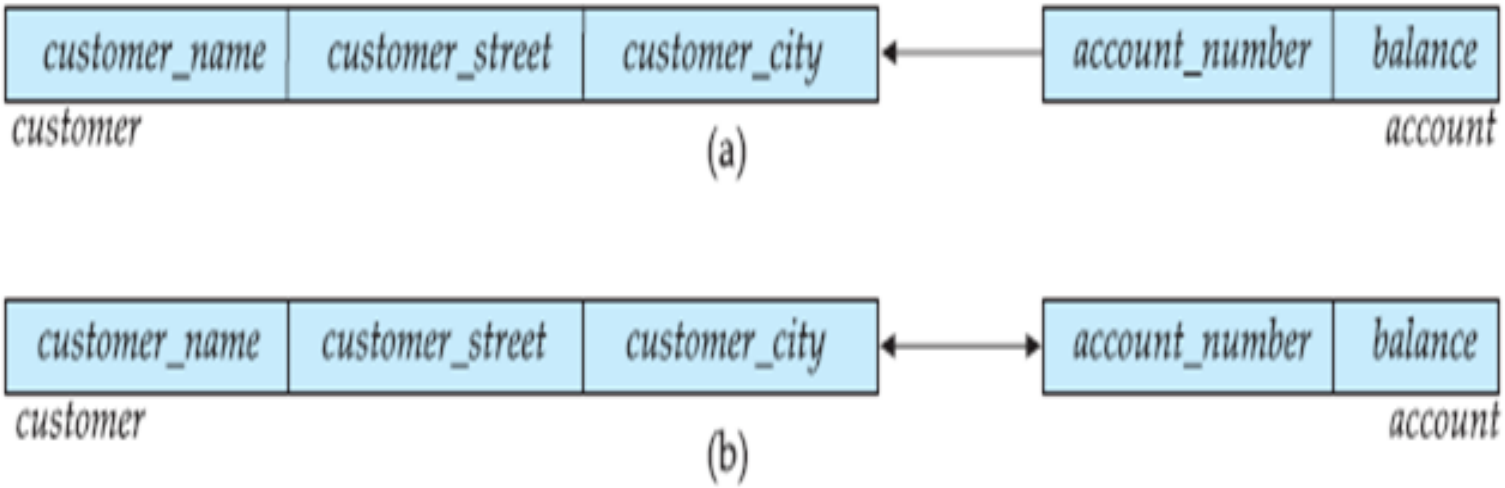


Data-structure diagram

Figure 1: Data

The record type customer corresponds to the entity set customer. It includes three fields— customer_name, customer_street, and customer_city. Similarly, account is the record type corresponding to the entity set account. It includes the two fields account_number and balance. Finally, the relationship depositor has been replaced with the link depositor.

Here, the relationship depositor is many to many. If the relationship depositor were one to many from customer to account, then the link depositor would have an arrow pointing to customer record type. Similarly, if the relationship depositor were one to one, then the link depositor would have two arrows: one pointing to account record type and one pointing to customer record type.



Two data-structure diagram

A sample database corresponding to the data-structure diagram of Figure 1 is shown in Figure 2.

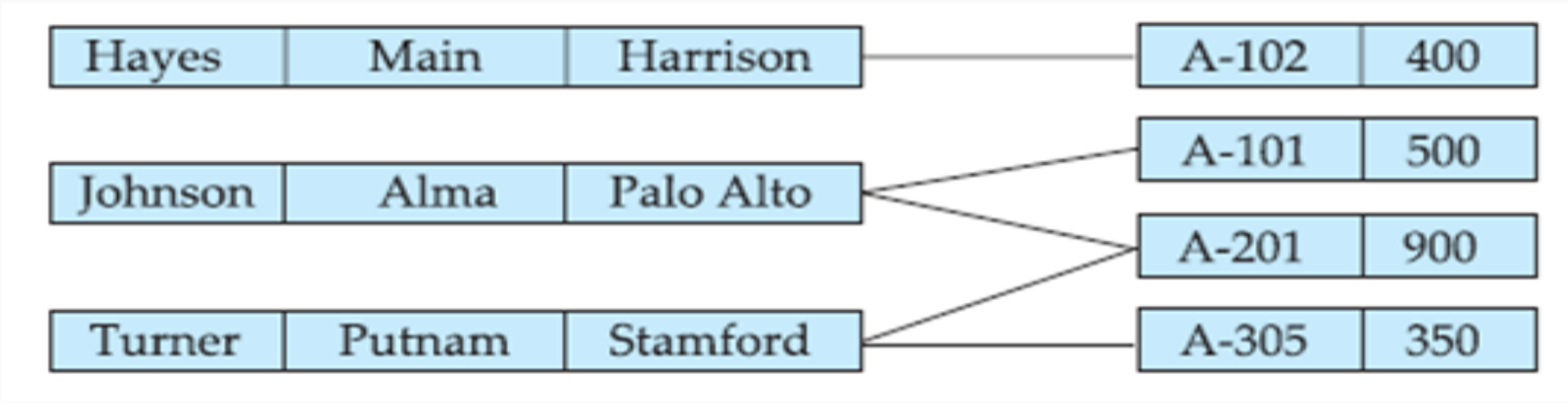
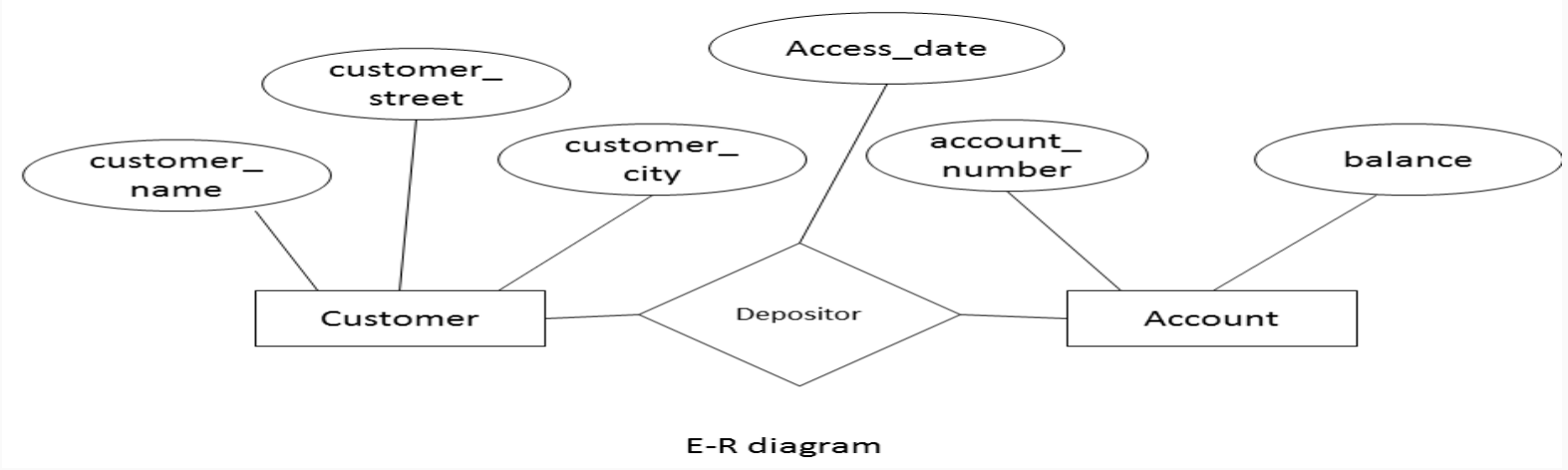


Figure 2: Sample database for data-structure diagram shown in figure 1

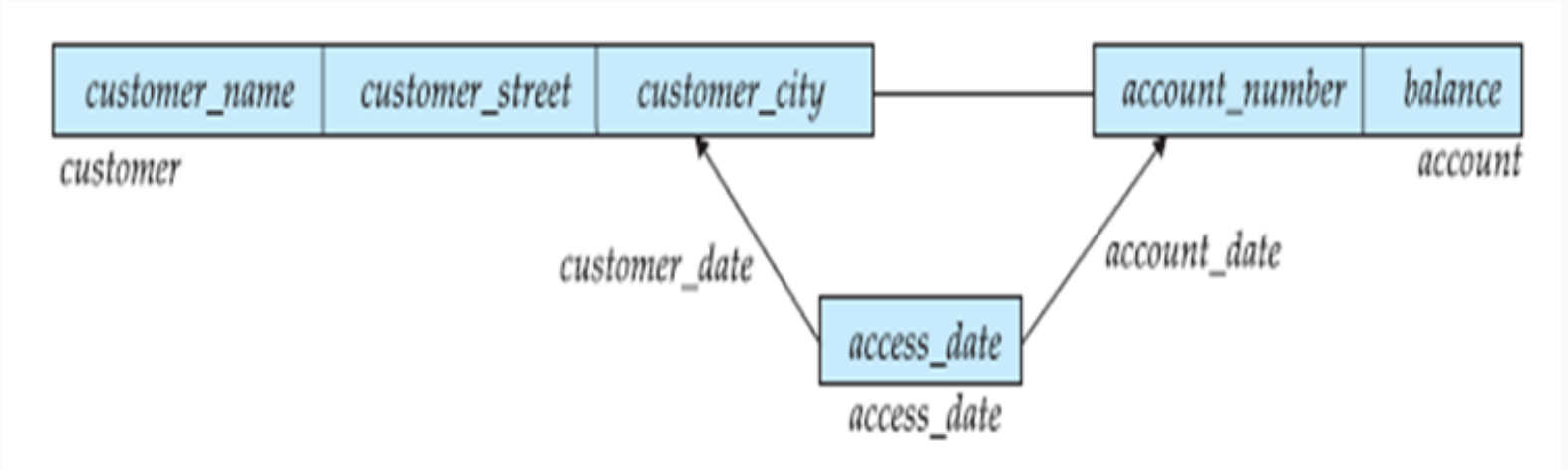
If a relationship includes descriptive attributes, the transformation from an E-R diagram to a data-structure diagram is more complicated. A link cannot contain any data value, so a new record type needs to be created and links need to be established.

Example: Consider the following E-R diagram.

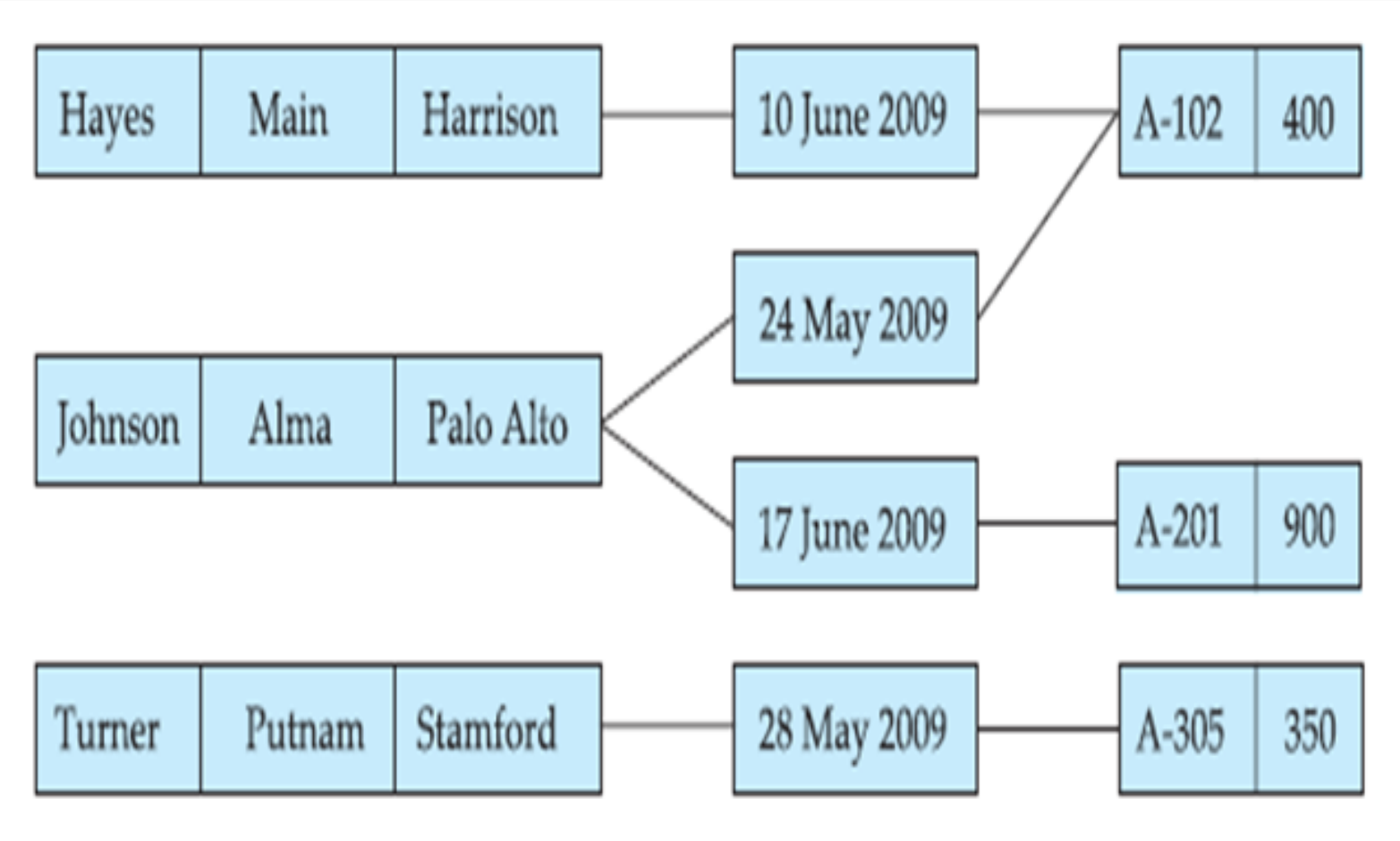


E-R diagram

The data structure diagram of this E-R diagram is shown in the following figure:-



Sample database corresponding to above data-structure diagram is the following:-



Hierarchical model

A hierarchical database consists of a collection of records that are connected to each other through links. A record is similar to a record in the network model. Each record is a collection of fields (attributes), each of which contains only one data value. A link is an association between precisely two records. Thus, a link here is similar to a link in the network model.

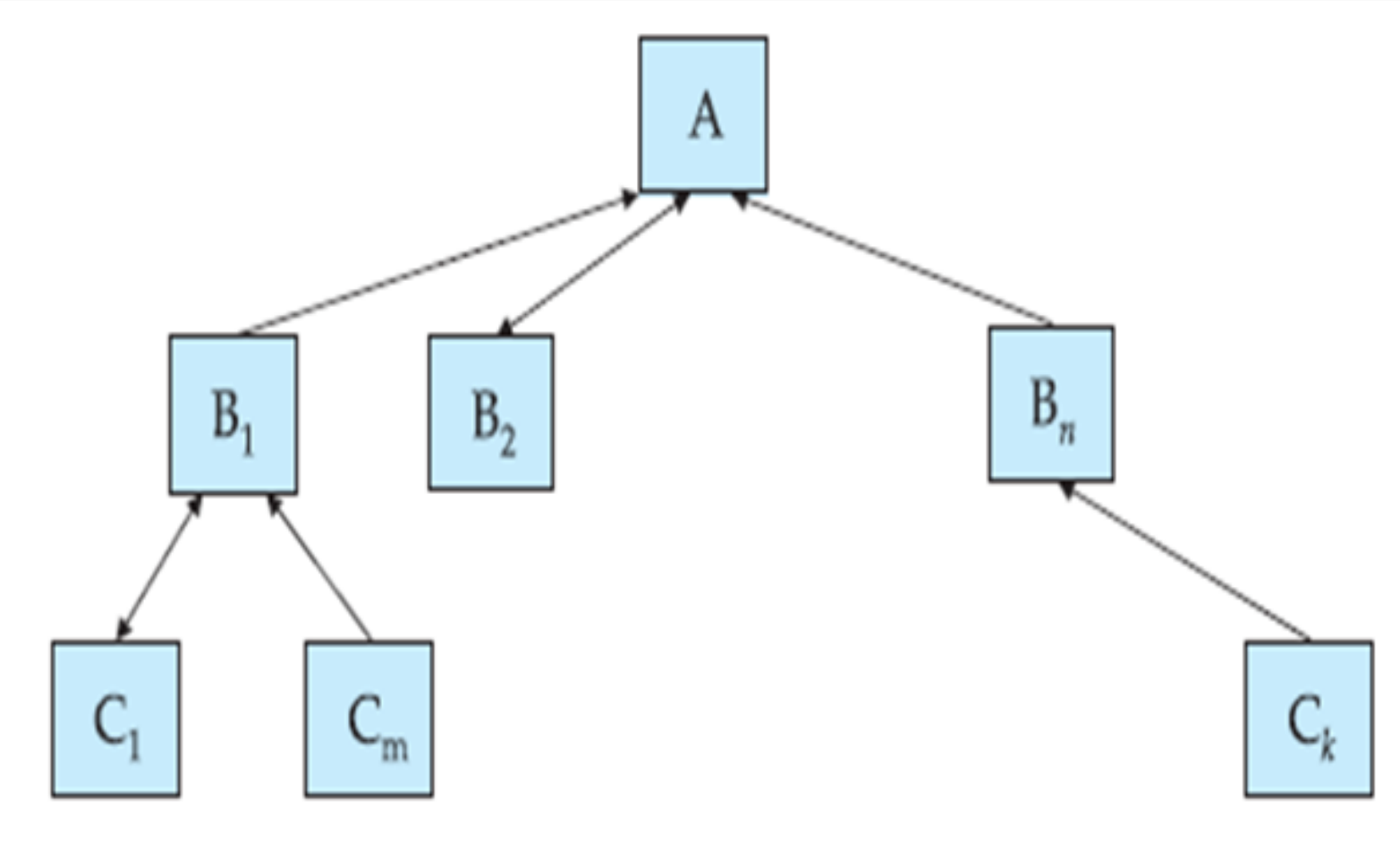
Consider a database that represents a customer-account relationship in a banking system. There are two record types: customer and account. The set of all customer and account records is organized in the form of a rooted tree, where the root of the tree is a dummy node. A hierarchical database is a collection of such rooted trees, and hence forms a forest. We shall refer to each such rooted tree as a database tree.

The schema for a hierarchical database is represented by tree-structured diagram. Such a diagram consists of two basic components:

1. Boxes, which correspond to record types
2. Lines, which correspond to links

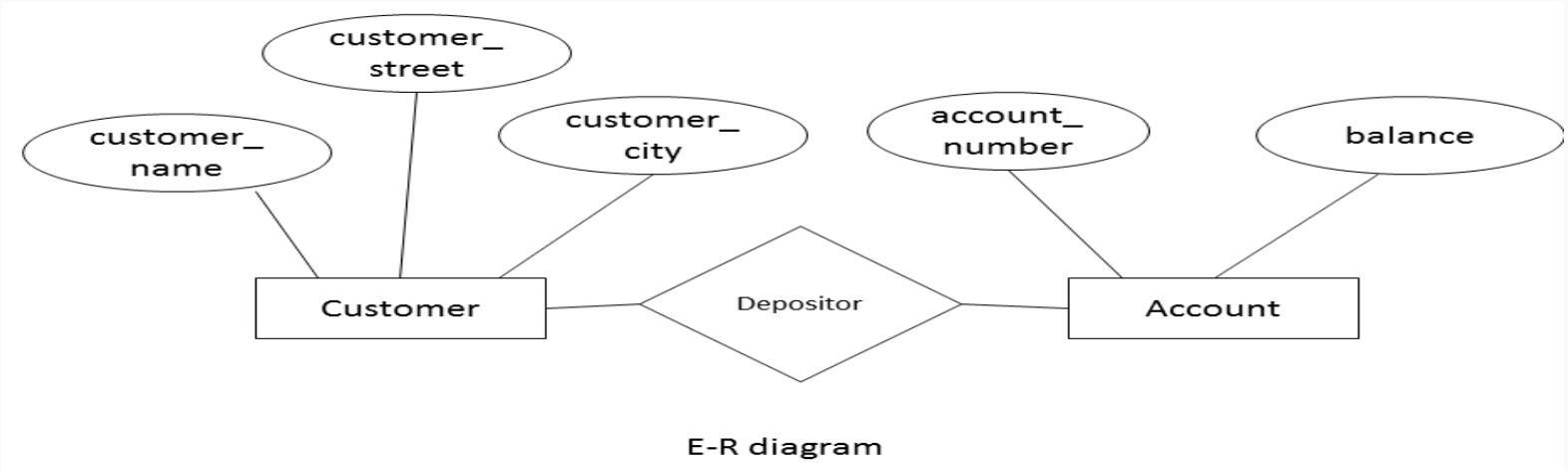
A tree-structure diagram serves the same purpose as an entity–relationship (E-R) diagram; namely, it specifies the overall logical structure of the database. A tree structure diagram is similar to a data-structure diagram in the network model. The main difference is that, in the latter, record types are organized in the form of an arbitrary graph, whereas in the former, record types are organized in the form of a rooted tree. The relationships formed in the tree-structure diagram must be such that only one-to-many or one-to-one relationships exist between a parent and a child.

The general form of a tree-structure diagram is shown in the following figure:-



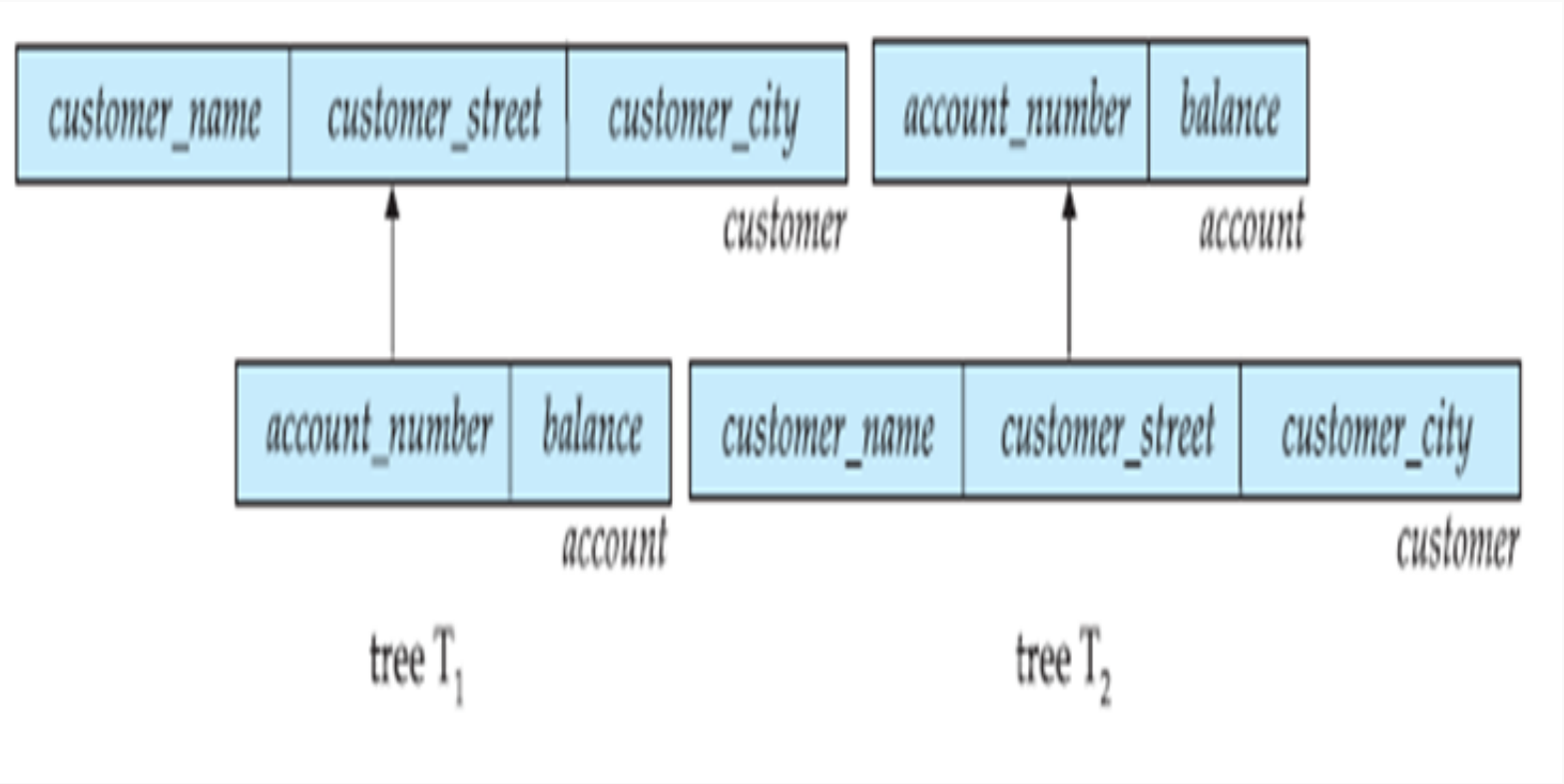
The database schema is represented as a collection of tree-structure diagrams. For each such diagram, there exists one single instance of a database tree. The root of this tree is a dummy node. The children of the dummy node are instances of the root record type in the tree-structure diagram. Each record instance may, in turn, have several children, which are instances of various record types, as specified in the corresponding tree-structure diagram. Every E-R diagram can be transformed into tree-structure diagram.

Example: Consider the following E-R diagram:-



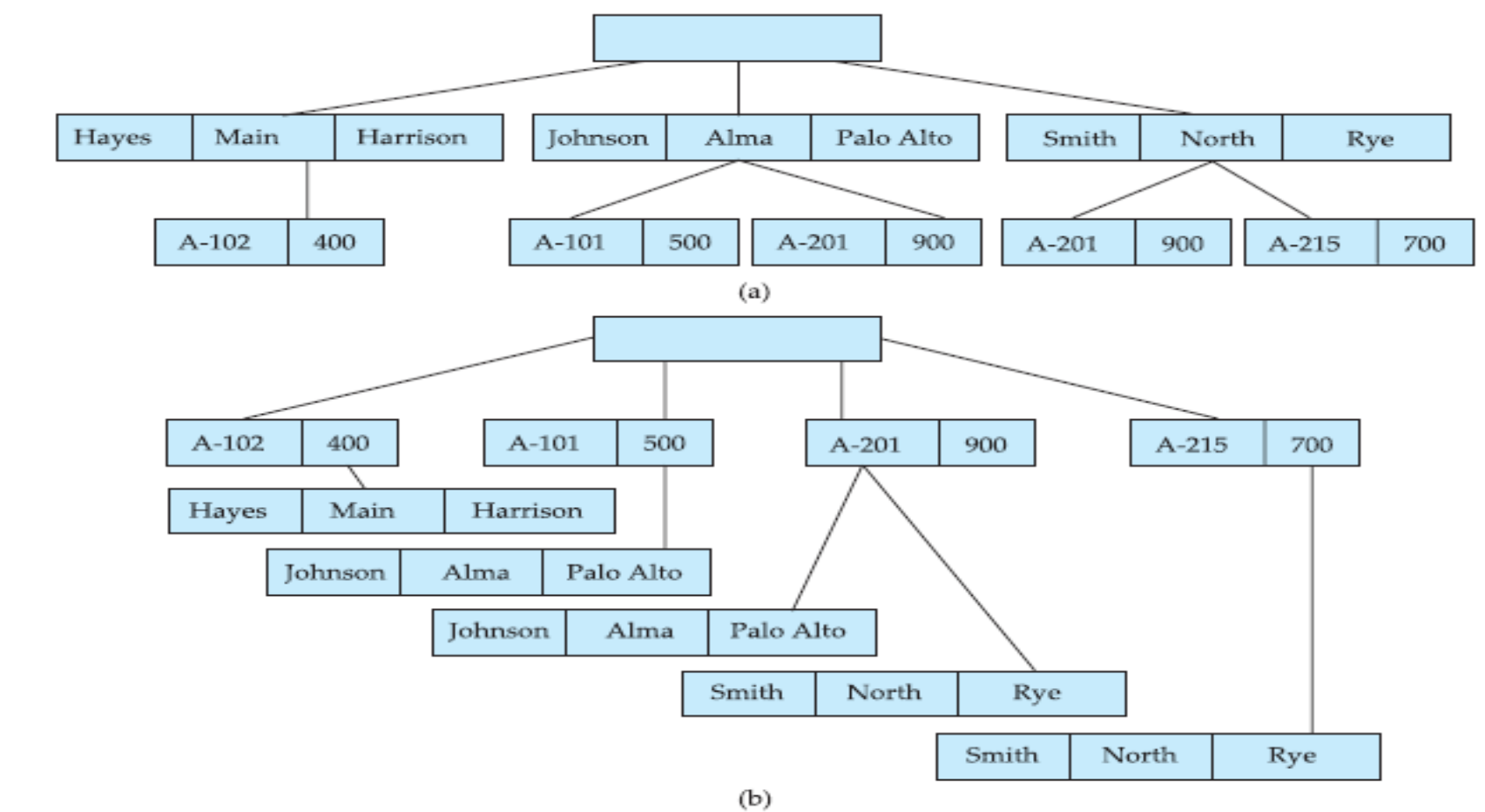
E-R diagram

Tree structure diagram for this E-R diagram is the following:-

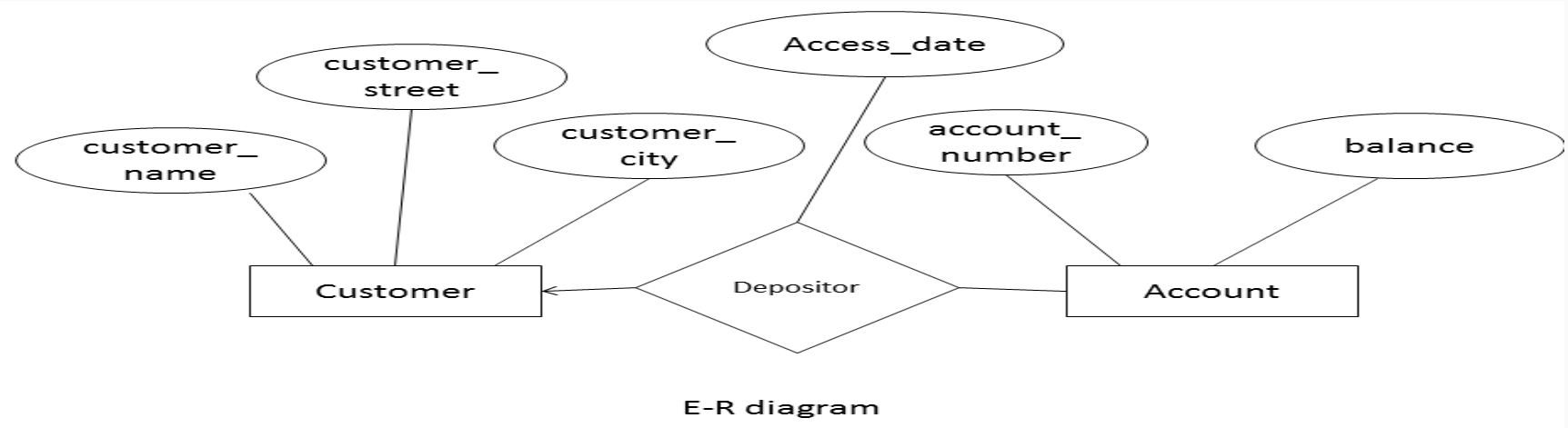


There are two database trees. The first tree (Figure a) corresponds to the tree-structure diagram T1; the second tree (Figure b) corresponds to the tree-structure diagram T2.

A sample database corresponding to the above tree-structure diagram is shown in the following figure:-

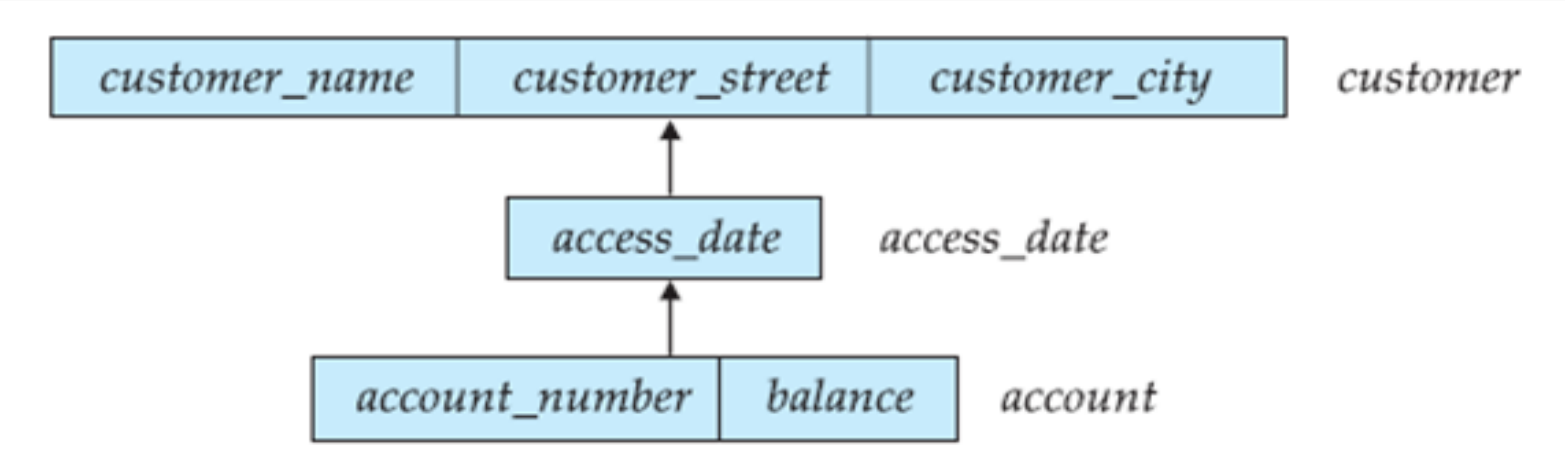


Example: Consider the following E-R diagram:-



E-R diagram

Tree structure diagram for this E-R diagram is the following:-



A sample database corresponding to the above tree-structure diagram is shown in the following figure:-