2.3 Relations

A binary relation $R(A, B)$ where A and B are sets is a subset of $A \times B$: i.e. $R \subseteq A \times B$. The domain of the relation is A and the codomain of the relation is B. The notation aRb signifies that $(a, b) \in R$.

A binary relation $R(A, A)$ is a relation between A and A. This type of relation may always be composed with itself, and its inverse is also a binary relation on A. The identity relation on A is defined by $a\ i_A a$ for all $a \in A$.

Example 2.4 There are many examples of relations

(i) The relation on a set of students in a class where $(a, b) \in R$ if the height of a is greater than the height of b.

(ii) The relation between A and B where $A = \{0, 1, 2\}$ and $B = \{3, 4, 5\}$ with R given by

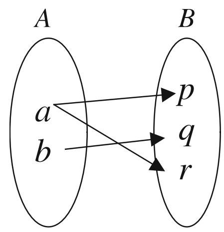
$$R = \{(0,3), (0,4), (1,4)\}$$

(iii) The relation less than (<) between and $\mathbb{R}$ and $\mathbb{R}$ is given by

$$\{(x, y) \in \mathbb{R}^2 : x < y\}$$

(iv) A bank may represent the relationship between the set of accounts and the set of customers by a relation. The implementation of a bank account will often be a positive integer with at most eight decimal digits.
The relationship between accounts and customers may be done with a relation $R \subseteq A \times B$, with the set A chosen to be the set of natural numbers, and the set B chosen to be the set of all human beings alive or dead. The set A could also be chosen to be $A = \{n \in \mathbb{N}: n < 10^8\}$

A relation $R(A, B)$ may be represented pictorially. This is referred to as the graph of the relation, and it is illustrated in the diagram below. An arrow from x to y is drawn if (x, y) is in the relation. Thus for the height relation R given by $\{(a, p), (a, r), (b, q)\}$ an arrow is drawn from a to p, from a to r and from b to q to indicate that (a, p), (a, r) and (b, q) are in the relation R.



The pictorial representation of the relation makes it easy to see that the height of a is greater than the height of p and r; and that the height of b is greater than the height of q.

An n-ary relation R $(A_1, A_2, \ldots A_n)$ is a subset of $(A_1 \times A_2 \times \ldots \times A_n)$. However, an n-ary relation may also be regarded as a binary relation $R(A, B)$ with $A = A_1 \times A_2 \times \ldots \times A_{n-1}$ and $B = A_n$.

2.3.1 Reflexive, Symmetric and Transitive Relations

There are various types of relations including reflexive, symmetric and transitive relations.

(i) A relation on a set A is *reflexive* if $(a, a) \in R$ for all $a \in A$.
(ii) A relation R is *symmetric* if whenever $(a, b) \in$ R then $(b, a) \in R$.
(iii) A relation is *transitive* if whenever $(a, b) \in R$ and $(b, c) \in$ R then $(a, c) \in R$.

A relation that is reflexive, symmetric and transitive is termed an *equivalence relation*.

Example 2.5 (**Reflexive Relation**) A relation is reflexive if each element possesses an edge looping around on itself. The relation in Fig. 2.2 is reflexive.

Example 2.6 (**Symmetric Relation**) The graph of a symmetric relation will show for every arrow from a to b an opposite arrow from b to a. The relation in Fig. 2.3 is symmetric: i.e. whenever $(a, b) \in R$ then $(b, a) \in R$.

Example 2.7 (**Transitive relation**) The graph of a transitive relation will show that whenever there is an arrow from a to b and an arrow from b to c that there is an arrow from a to c. The relation in Fig. 2.4 is transitive: i.e. whenever $(a, b) \in R$ and $(b, c) \in R$ then $(a, c) \in R$.

Example 2.8 (**Equivalence relation**) The relation on the set of integers $\mathbb{Z}$ defined by $(a, b) \in R$ if $a - b = 2\,k$ for some $k \in \mathbb{Z}$ is an equivalence relation, and it partitions the set of integers into two equivalence classes: i.e. the even and odd integers.

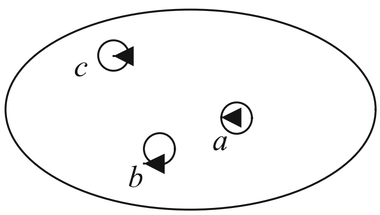


Fig. 2.2 Reflexive relation

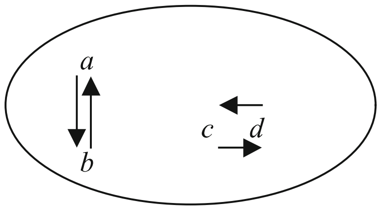


Fig. 2.3 Symmetric relation

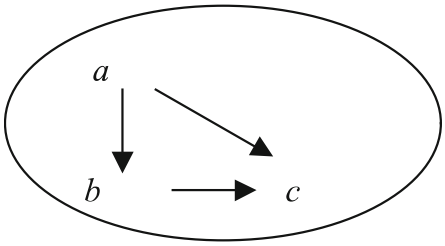


Fig. 2.4 Transitive relation

Domain and Range of Relation

The domain of a relation R (A, B) is given by $\{a \in A \mid \exists b \in B$ and $(a, b) \in R\}$. It is denoted by **dom** R. The domain of the relation R = $\{(a, p), (a, r), (b, q)\}$ is $\{a, b\}$.

The range of a relation R (A, B) is given by $\{b \in \mathrm{B} \mid \exists a \in A$ and $(a, b) \in \mathrm{R}\}$. It is denoted by **rng** R. The range of the relation $R = \{(a, p), (a, r), (b, q)\}$ is $\{p, q, r\}$.

Inverse of a Relation

Suppose $R \subseteq A \times B$ is a relation between A and B then the inverse relation $R^{-1} \subseteq B \times A$ is defined as the relation between B and A and is given by

$$b\ R^{-1}\ a \text{ if and only if } a\ R\ b$$

That is

$$R^{-1} = \{(b, a) \in B \times A: (a, b) \in R\}$$

Example 2.9 Let R be the relation between $\mathbb{Z}$ and $\mathbb{Z}^+$ defined by mRn if and only if $m^2 = n$. Then $R = \{(m, n) \in \mathbb{Z} \times \mathbb{Z}^+: m^2 = n\}$ and $R^{-1} = \{(n, m) \in \mathbb{Z}^+ \times \mathbb{Z}: m^2 = n\}$.

For example, $-3\ R\ 9$, $-4\ R\ 16$, $0\ R\ 0$, $16\ R^{-1} - 4$, $9\ R^{-1} - 3$, etc.

Partitions and Equivalence Relations

An equivalence relation on A leads to a partition of A, and vice versa for every partition of A there is a corresponding equivalence relation.

Let A be a finite set and let $A_1, A_2, \ldots, A_n$ be subsets of A such $A_i \neq \varnothing$ for all i, $A_i \cap A_j = \varnothing$ if $i \neq j$ and $A = \cup_i^n A_i = A_1 \cup A_2 \cup \ldots \cup A_n$. The sets A_i partition the set A, and these sets are called the classes of the partition (Fig. 2.5).

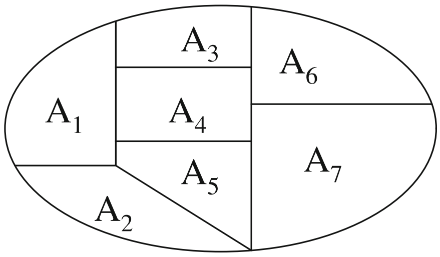


Fig. 2.5 Partitions of A

Theorem 2.2 (Equivalence Relation and Partitions) *An equivalence relation on A gives rise to a partition of A where the equivalence classes are given by Class (a) = {x | x ∈ A and (a, x) ∈ R}. Similarly, a partition gives rise to an equivalence relation R, where (a, b) ∈ R if and only if a and b are in the same partition.*

Proof Clearly, $a \in$ Class(a) since R is reflexive and clearly the union of the equivalence classes is A. Next, we show that two equivalence classes are either equal or disjoint.

Suppose Class(a) ∩ Class(b) ≠ ∅. Let $x \in$ Class(a) ∩ Class(b) and so (a, x) and (b, x) ∈ R. By the symmetric property (x, b) ∈ R and since R is transitive from (a, x) and (x, b) in R we deduce that (a, b) ∈ R. Therefore $b \in$ Class(a). Suppose y is an arbitrary member of Class (b) then (b, y) ∈ R therefore from (a, b) and (b, y) in R we deduce that (a, y) is in R. Therefore since y was an arbitrary member of Class(a) we deduce that Class(b) ⊆ Class(a). Similarly, Class(a) ⊆ Class(b) and so Class(a) = Class(b).

This proves the first part of the theorem and for the second part we define a relation R such that (a, b) ∈ R if a and b are in the same partition. It is clear that this is an equivalence relation.

2.3.2 Composition of Relations

The composition of two relations $R_1(A, B)$ and $R_2(B, C)$ is given by R_2 o R_1 where $(a, c) \in R_2$ o R_1 if and only there exists $b \in$ B such that $(a, b) \in R_1$ and $(b, c) \in R_2$. The composition of relations is associative: i.e.

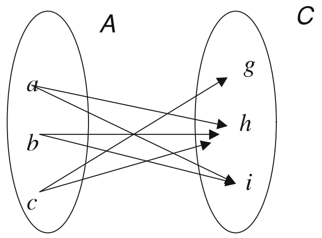
$$(R_3 \, \mathrm{o} \, R_2) \, \mathrm{o} \, R_1 = R_3 \, \mathrm{o} \, (R_2 \, \mathrm{o} \, R_1)$$

Example 2.10 (**Composition of Relations**) Consider a library that maintains two files. The first file maintains the serial number s of each book as well as the details of the author a of the book. This may be represented by the relation $R_1 = sR_1a$. The second file maintains the library card number c of its borrowers and the serial number s of any books that they have borrowed. This may be represented by the relation $R_2 = c\ R_2 s$.

The library wishes to issue a reminder to its borrowers of the authors of all books currently on loan to them. This may be determined by the composition of R_1 o R_2: i.e. $c\ R_1$ o $R_2\ a$ if there is book with serial number s such that $c\ R_2\ s$ and $s\ R_1\ a$.

Example 2.11 (**Composition of Relations**) Consider sets $A = \{a, b, c\}$, $B = \{d, e, f\}$, $C = \{g, h, i\}$ and relations $R(A, B) = \{(a, d), (a, f), (b, d), (c, e)\}$ and $S(B, C) = \{(d, h), (d, i), (e, g), (e, h)\}$. Then we graph these relations and show how to determine the composition pictorially.

S o R is determined by choosing $x \in A$ and $y \in C$ and checking if there is a route from x to y in the graph (Fig. 2.6). If so, we join x to y in S o R. For example, if we consider a and h we see that there is a path from a to d and from d to h and therefore (a, h) is in the composition of S and R.



The union of two relations $R_1(A, B)$ and $R_2(A, B)$ is meaningful (as these are both subsets of $A \times B$). The union $R_1 \cup R_2$ is defined as $(a, b) \in R_1 \cup R_2$ if and only if $(a, b) \in R_1$ or $(a, b) \in R_2$.

Similarly, the intersection of R_1 and R_2 ($R_1 \cap R_2$) is meaningful and is defined as $(a, b) \in R_1 \cap R_2$ if and only if $(a, b) \in R_1$ and $(a, b) \in R_2$. The relation R_1 is a subset of R_2 ($R_1 \subseteq R_2$) if whenever $(a, b) \in R_1$ then $(a, b) \in R_2$.

The inverse of the relation R was discussed earlier and is given by the relation R^{-1} where $R^{-1} = \{(b, a) \mid (a, b) \in R\}$.

The composition of R and R^{-1} yields: R^{-1} o $R = \{(a, a) \mid a \in \text{dom } R\} = i_A$ and R o $R^{-1} = \{(b, b) \mid b \in \textbf{dom } R^{-1}\} = i_B$.

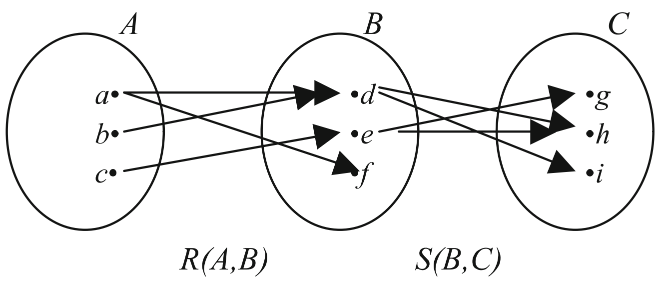


Fig. 2.6 Composition of relations

2.3.3 Binary Relations

A binary relation R on A is a relation between A and A, and a binary relation can always be composed with itself. Its inverse is a binary relation on the same set. The following are all relations on A:

$$R^2 = R \circ R$$
$$R^3 = (R \circ R) \circ R$$
$$R^0 = i_A(\text{identity relation})$$
$$R^{-2} = R^{-1} \circ R^{-1}$$

Example 2.12 Let R be the binary relation on the set of all people P such that $(a, b) \in R$ if a is a parent of b. Then the relation R^n is interpreted as

R is the parent relationship
R^2 is the grandparent relationship
R^3 is the great grandparent relationship.
R^{-1} is the child relationship.
R^{-2} is the grandchild relationship.
R^{-3} is the great grandchild relationship

This can be generalized to a relation R^n on A where $R^n = R \circ R \circ \ldots \circ R$ (n-times). The transitive closure of the relation R on A is given by

$$R^* = \cup_{i=0}^{\infty} R^i = R^0 \cup R^1 \cup R^2 \cup \ldots R^n \cup \ldots$$

where R^0 is the reflexive relation containing only each element in the domain of R: i.e. $R^0 = i_A = \{(a, a) \mid a \in \mathbf{dom}\ R\}$.

The positive transitive closure is similar to the transitive closure except that it does not contain R^0. It is given by

$$R^+ = \cup_{i=1}^{\infty} R^i = R^1 \cup R^2 \cup \ldots \cup R^n \cup \ldots$$

$a\, R^+\, b$ if and only if $a\, R^n\, b$ for some $n > 0$: i.e. there exists $c_1, c_2 \ldots c_n \in A$ such that

$$aRc_1, c_1Rc_2, \ldots, c_nRb$$

Parnas[7] introduced the concept of the limited domain relation (LD-relation), and a LD relation L consists of an ordered pair (R_L, C_L) where R_L is a relation and C_L is a subset of Dom R_L. The relation R_L is on a set U and C_L is termed the competence

[7]Parnas made important contributions to software engineering in the 1970s. He invented information hiding which is used in object-oriented design.

set of the LD relation L. A description of LD relations and a discussion of their properties are in Chap. 2 of [1].

The importance of LD relations is that they may be used to describe program execution. The relation component of the LD relation L describes a set of states such that if execution starts in state x it may terminate in state y. The set U is the set of states. The competence set of L is such that if execution starts in a state that is in the competence set then it is guaranteed to terminate.

2.3.4 Applications of Relations

A relational database management system (RDBMS) is a system that manages data using the relational model, and examples of such systems include RDMS developed at MIT in the 1970s; Ingres developed at the University of California, Berkeley in the mid-1970s; Oracle developed in the late 1970s; DB2; Informix; and Microsoft SQL Server.

A relation is defined as a set of tuples and is usually represented by a table. A table is data organized in rows and columns, with the data in each column of the table of the same data type. Constraints may be employed to provide restrictions on the kinds of data that may be stored in the relations. Constraints are Boolean expressions which indicate whether the constraint holds or not, and are a way of implementing business rules in the database.

Relations have one or more keys associated with them, and the *key uniquely identifies the row of the table*. An index is a way of providing fast access to the data in a relational database, as it allows the tuple in a relation to be looked up directly (using the index) rather than checking all of the tuples in the relation.

The Structured Query Language (SQL) is a computer language that tells the relational database what to retrieve and how to display it. A stored procedure is executable code that is associated with the database, and it is used to perform common operations on the database.

The concept of a relational database was first described in a paper '*A Relational Model of Data for Large Shared Data Banks*' by Codd [2]. A relational database is a database that conforms to the relational model, and it may be defined as a set of relations (or tables).

Codd (Fig. 2.7) developed the *relational data base model* in the late 1960s, and today, this is the standard way that information is organized and retrieved from computers. Relational databases are at the heart of systems from hospitals' patient records to airline flight and schedule information.

A binary relation R(A, B) where A and B are sets is a subset of the Cartesian product (A × B) of A and B. The domain of the relation is A, and the codomain of the relation is B. The notation $a\mathrm{R}b$ signifies that there is a relation between a and b and that $(a, b) \in \mathrm{R}$. An n-ary relation R $(A_1, A_2, \ldots A_n)$ is a subset of the Cartesian product of the n sets: i.e. a subset of $(A_1 \times A_2 \times \ldots \times A_n)$. However, an

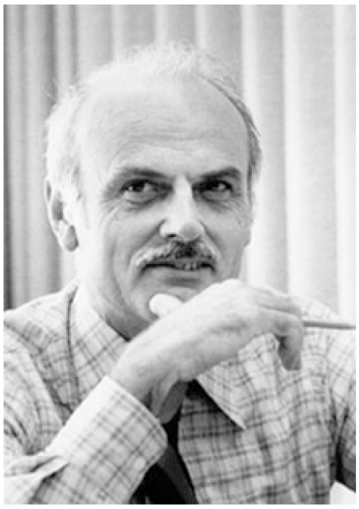


Fig. 2.7 Edgar Codd

n-ary relation may also be regarded as a binary relation R(A, B) with $A = A_1 \times A_2 \times \ldots \times A_{n-1}$ and $B = A_n$.

The data in the relational model are represented as a mathematical n-ary relation. In other words, a relation is defined as a set of n-tuples, and is usually represented by a table. A table is a visual representation of the relation, and the data are organized in rows and columns. The data stored in each column of the table are of the same data type.

The basic relational building block is the domain or data type (often called just type). Each row of the table represents one n-tuple (one tuple) of the relation, and the number of tuples in the relation is the cardinality of the relation. Consider the PART relation taken from [3], where this relation consists of a heading and the body. There are five data types representing part numbers, part names, part colours, part weights, and locations in which the parts are stored. The body consists of a set of n-tuples, and the PART relation given in Fig. 2.8 is of cardinality six.

For more information on the relational model and databases see [4]