Why

We can generalize the product of two sets to a product of a family of sets.

Discussion for pairs

Let A and B be sets. There is a natural correspondence between $A \times B$ (see *Cartesian Products*) and a particular set of families. The particular set of families $z : \{i, j\} \to (A \cup B)$ with $z_i \in A$ and $z_b \in B$. The family z corresponds with the pair (z_i, z_i). The pair (a, b) corresponds to the family $z : \{i, j\} \in (A \cup B)$ defined by $z(i) = a$ and $z(j) = b$. In other words, we can think about ordered pairs as special families. The generalization of Cartesian products to families generalizes the notion for families.

Direct Products

Let X be a set. Let $A : I \to X$ be a family of subsets of X. The *direct product* or *family Cartesian product* of A is the set of all families $a : I \to X$ which satisfy $a_i \in A_i$ for every $i \in I$. A function on a product is called a *function of several variables* and, in particular, a function on the product $X \times Y$ is called a *funciton of two variables.*

Notation

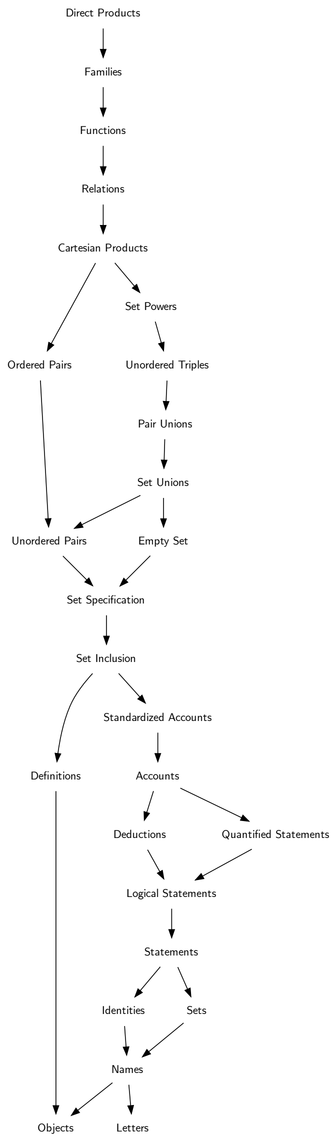
We denote the product of the family $\{A_i\}$ by

$$\prod_{i \in I} A_i$$

We read this notation as "product over i in I of A sub-i."

Projections

The word "projection" is used in two senses with families. Let I be a set, and let $\{A_i\}$ be a family of sets. Define $A = \prod_{i \in I} A_i$. Then if $J \subset I$, there is a natural correspondence between the elements of X and those of $\prod_{j \in J} A_j$. To each element $x \in X$, we restrict x to J and this is an element of $\prod_{j \in J} A_j$. The correspondence is called the *projection* of X onto $\prod_{i \in J} X_i$ Also, the value of x at j is called the *projection of x onto index* j or the j*-coordinate* of x.


Direct Products
Families
Functions
Relations
Cartesian Products
Set Powers
Ordered Pairs
Unordered Triples
Pair Unions
Set Unions
Unordered Pairs
Empty Set
Set Specification
Set Inclusion
Standardized Accounts
Definitions
Accounts
Deductions
Quantified Statements
Logical Statements
Statements
Identities
Sets
Names
Objects
Letters