Everything on Relations

Atoms are primitive entities. They are indivisible, immutable and uninterpreted

A relation is a structure that relates atoms. A relation table consists of a set of tuples (rows). The <u>size</u> is the number of tuples. Each tuple is sequence of atoms.

All tuples must have the same length; arity

Order of tuples and order of atoms doesn’t matter

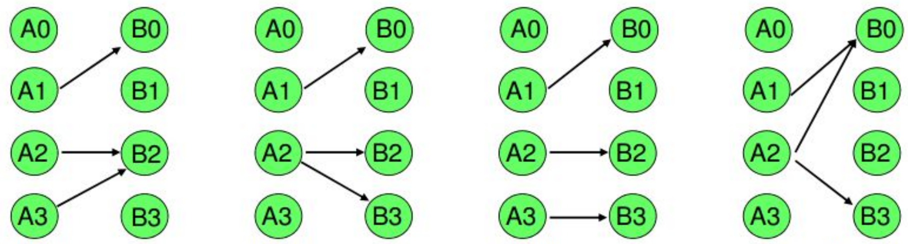
Relations

Unary relation	Arity = 1
Binary relation	Arity = 2
Ternary relation	Arity = 3
Scalar	Single values; unary relation with only one tuple

A field in a signature is a relation from atoms of that signature to atoms of the type indicated in the field

Functions and Injective Relations

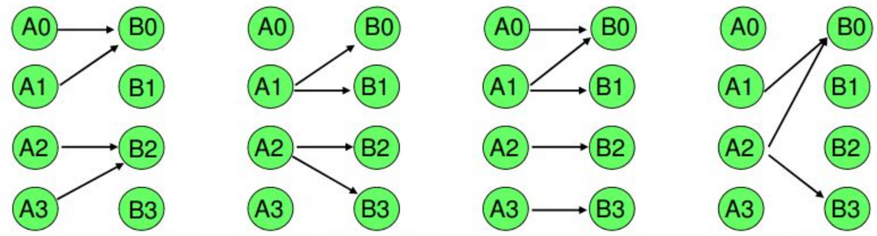
Function	A binary relation that maps each A to at most one B
Injective	A binary relation that maps at most one A to each B



function, but not injective — *injective, but not a function* — *function, and injective* — *neither a function, nor injective*

Total and Surjective Relations

Total	A binary relation that maps each A to at least one B
Surjective	A binary relation that maps at least one A to each B



total, but not surjective — *surjective, but not total* — *total and surjective* — *neither total nor surjective*

Multiplicities and Relations

```
sig A {
    r1 : B          Total and functional (default)
    r2 : one B      Total and functional
    r3 : lone B     Functional
    r4 : some B     Total
    r5 : set B      No constraints
}
```

When multiplicity is omitted; one by default
A field in a signature can be a relation itself.

```
sig A {
    r1 : B -> C        // no multiplicity constraints
    r2 : B -> some C   // each B maps to at least one C (total)
    r3 : B -> lone C   // each B maps to at most one C (functional)
    r4 : B -> one C    // each B maps to exactly one C (total and functional)
    r5 : B some -> C   // at least one B maps to every C (surjective)
    r6 : B lone -> C   // at most one B maps to each C (injective)
    r7 : B one -> C    // exactly one B maps to each C (surjective and injective)
}
```

Left and right multiplicities can be combined; rel : A one -> one B

The tuple A → B can be written as (A,B)

domain	Set of atoms in first column
range	Set of atoms in last column
iden	Identity relation; relates each element to itself

Relational Operations

→	Arrow (product)
.	Dot (join)
[]	Box (join)
^	Transitive closure
*****	Reflexive transitive closure
~	Transpose (Inverse)
<:	Domain restriction
:>	Range restriction
++	override

Product	The product of p → q of two relations p and q is the relation consisting of all possible combinations of tuples from p and q
Dot Join	P.Q contains concatenations of tuples from p and q where the values of the last column of p and first column of q agree. Dot join navigation can be forwards or backwards, and can be used to compose multiple relations
Box Join	Same as dot join : P[Q] same as q.p. Lower precedence than dot operator. More readable
Transpose	The transpose ~r of a binary relation r is the relation formed by reversing the order of atoms in each tuple in r
Transitive Closure	A binary relation is <u>transitive</u> if whenever it contains the tuples a → b and b → c it also contains a → c. The transitive closure ^r of a binary r is the smallest relation that contains r and is transitive <u>How it works</u> **-** You can compute the transitive closure by taking the relation, adding the join of the relation with itself, then adding the join of the relation with that, and so on, until adding another .r doesn't change anything The transitive closure of r can also be described as the relation that characterises the atoms reachable from each element in the domain r in <u>one or more steps</u> through r
Reflexive Transitive Closure	A binary relation is reflexive, if it contains the tuple a → a for every atom a in univ The reflexive transitive closure *r of a binary relation r is the smallest relation that contains r and is both reflexive and transitive The reflexive transitive closure of r can also be described as the relation that characterises the atoms reachable via the relation from each element in univ in <u>zero or more steps</u>