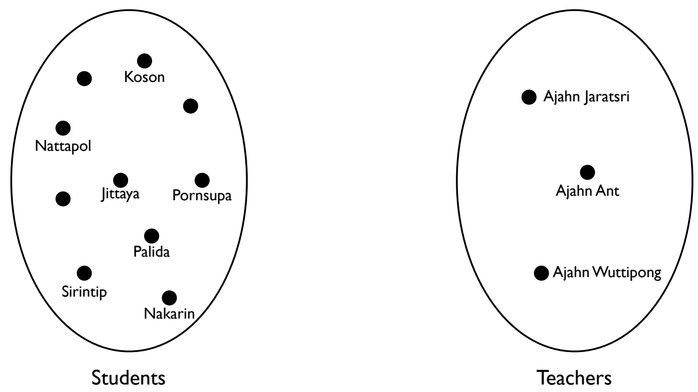
3. Relations

Discrete Mathematics

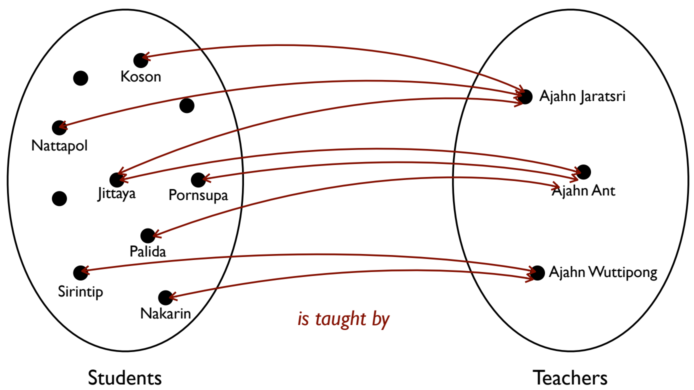
Introduction

Consider two sets...



Introduction

...and a relationship



Introduction

The relationship ‘is taught by’ can be written as *a set of ordered pairs*:

R = { (Koson, Aj.Jaratsri), (Nattapol, Aj.Jaratsri), (Jittaya, Aj.Jaratsri), (Jittaya, Aj.Ant), (Pornsupa, Aj.Ant), (Palida, Aj.Ant), ...}

(Remember: an ordered pair is a sequence of length 2)

Introduction

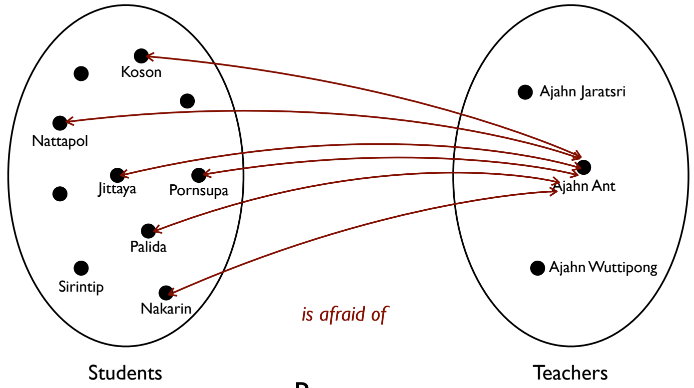
The relationship 'is taught by' is

a subset of the cartesian product of Students and Teachers:

$R \subseteq$ Students × Teachers

Introduction

There are many possible relationships between Students and Teachers



$R_{afraidof}$

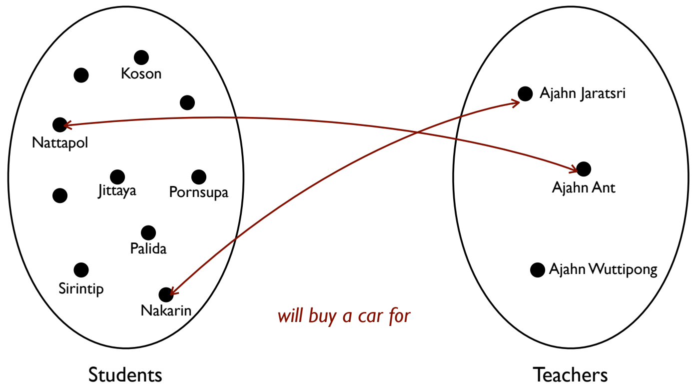
Introduction

$R_{afraidof}$: Students ↔ Teachers =

{ (x,y) ∈ Students × Teachers | x is afraid of y }

Introduction

There are many possible relationships between Students and Teachers

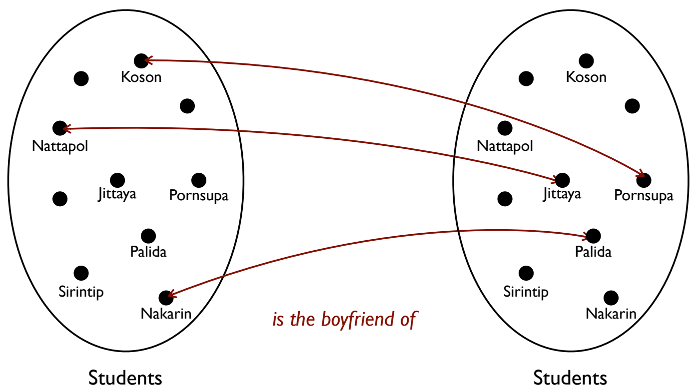


Introduction

$R_{buyacar}$: Students ↔ Teachers =
{ (x,y) ∈ Students × Teachers | x will buy a car for y }

Introduction

And there can be relationships between Students and Students



Introduction

$R_{boyfriend}$: Students ↔ Students =
{ (x,y) ∈ Students × Students | x is the boyfriend of y }

Relations

Definition:

A relation between sets A, B is a subset of A × B, written as:

$$R_p : A \leftrightarrow B$$

where $R_p \subseteq A \times B$

Example:

$R_{\leq} = \{ (a,b) \in \mathbf{N} \times \mathbf{N} \mid a \leq b \}$

(The cartesian product of all natural numbers where the first number is less than or equal to the second number.)

Relations

Examples of relations:

$R_<, R_\leq, R_>, R_\geq : \mathbf{N} \leftrightarrow \mathbf{N}$

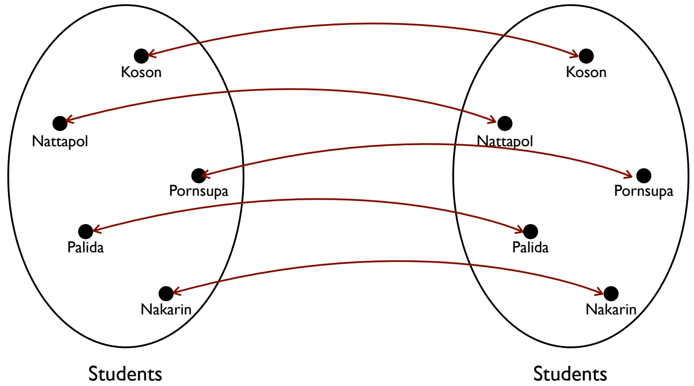
$R_| : \mathbf{Z} \leftrightarrow \mathbf{Z}$ $\quad m \mid n \;\Leftrightarrow\;$ m divides n $\;\Leftrightarrow\; \exists k \in \mathbf{Z} : km = n$

R_q : People $\leftrightarrow$ People $\quad x\ q\ y \;\Leftrightarrow\;$ x is a child of y (q = ‘is a child of’)

R_t : People $\leftrightarrow$ Animals $\quad x\ t\ y \;\Leftrightarrow\;$ x has a pet y

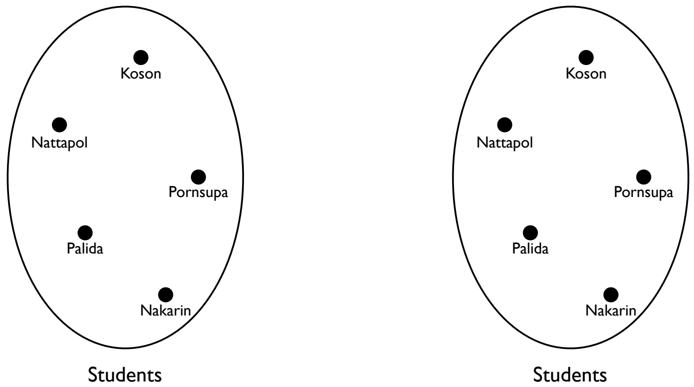
Relations

Question 1: What is this relation?



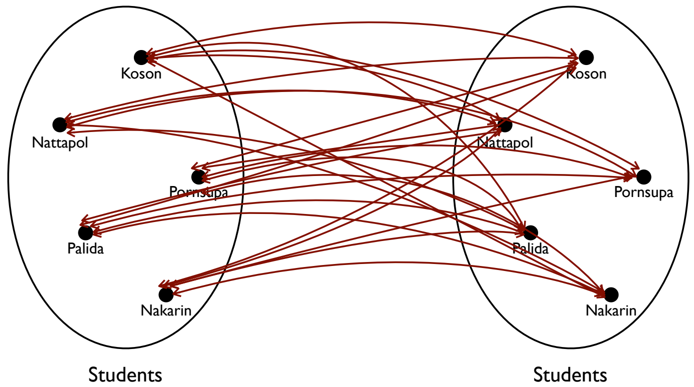
Relations

Question: 2 What is this relation?



Relations

Question 3: What is this relation?

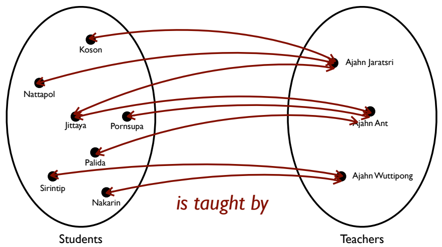


Relations

Common relations:

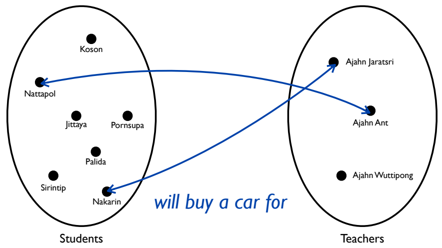
Equality relation	$R_= : A \leftrightarrow A$	$R_= = \{ (a,a) \mid a \in A \}$
Empty relation	$\varnothing : A \leftrightarrow A$	
Complete relation	$A^2 : A \leftrightarrow A$	

Operations on relations



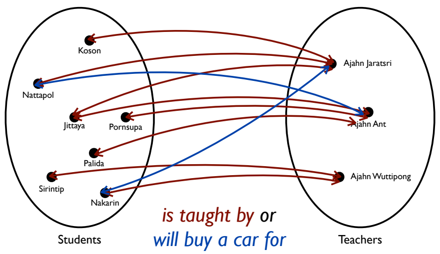
$= R_a$

$R_b =$



Operations on relations

$R_a \cup R_b =$

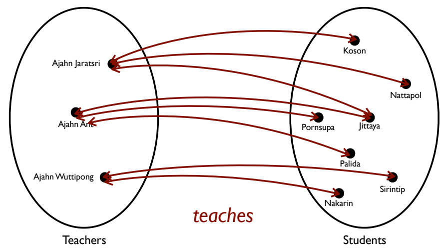


Also $R_a \cap R_b$ and $R_a \setminus R_b$

Operations on relations

$R_{a-1} =$

(inverse)



Operations on relations

Let $R_p, R_{p'} : A \leftrightarrow B$ and $R_q : B \leftrightarrow C$

The *intersection*, *union* and *difference* of p and p′:

$$R_p \cap R_{p'},\quad R_p \cup R_{p'},\quad R_p \setminus R_{p'} : A \leftrightarrow B$$

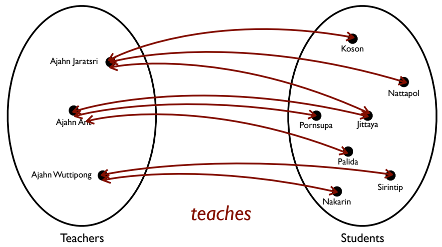
The *inverse* of p:

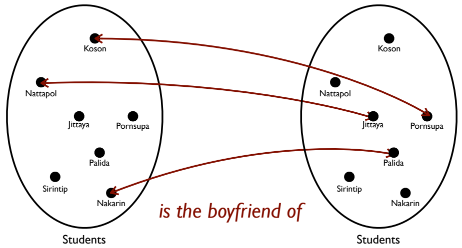
$$R_{p^{-1}} : B \leftrightarrow A$$

$$\forall (b, a) \in B \times A : b\ (p^{-1})\ a \Leftrightarrow a\ p\ b$$

Operations on relations

$R_a =$

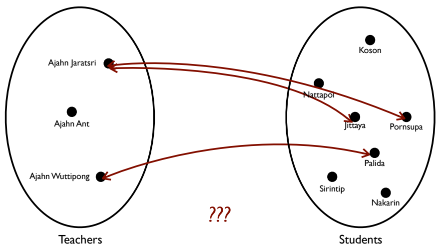




$= R_b$

Operations on relations

$R_{a \circ b} =$



Question: What is this relation?

Operations on relations

Let $R_p : A \leftrightarrow B$ and $R_q : B \leftrightarrow C$

The *composition* of p and q:

$$R_{p \circ q} : A \leftrightarrow C$$

$$\forall (a, c) \in A \times C : a\ (p \circ q)\ c$$

$$\forall (a, c) \in A \times C : (\exists b \in B : (a\ p\ b) \wedge (b\ q\ c))$$

Pause... and wake up!

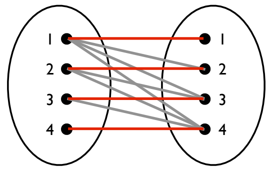
Properties of relations

Relation $R_p : A \leftrightarrow A$ is *reflexive*, if $\forall a \in A : a \; p \; a$

or...

R_p is reflexive if and only if $R_= \subseteq R_p$

Example: $R_\leq$



Relation must have (1,1), (2,2), etc.

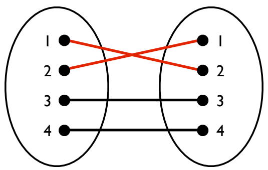
Properties of relations

Relation $R_p : A \leftrightarrow A$ is *symmetric*, if $\forall a,b \in A : a \; p \; b \Rightarrow b \; p \; a$

or...

R_p symmetric if and only if $R_{p^{-1}} = R_p$

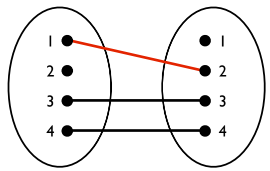
Example:



If the relation has (1,2) then it must have (2,1)

Properties of relations

Example of *not* symmetric:



R = { (1,2), (3,3), (4,4) }

Because in the relation we have (1,2), but we do not have (2,1).

Properties of relations

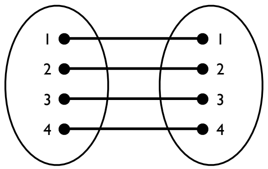
Relation $R_p : A \leftrightarrow A$ is *antisymmetric*, if $\forall a,b \in A : (a\ p\ b \wedge b\ p\ a) \Rightarrow a = b$

or...

R_p antisymmetric if and only if $R_p \cap R_{p-1} \subseteq R_=$

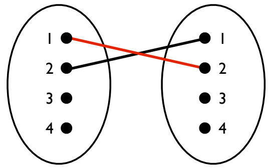
(Note: a relation can be symmetric and antisymmetric)

Example: $R_=$



Properties of relations

Example of *not* antisymmetric:



R = { (1,2), (2,1) }

Because in the relation we have (1,2) and (2,1), but... 1 != 2 .

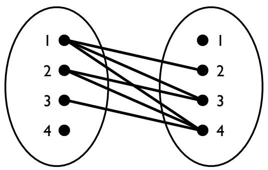
Properties of relations

Relation $R_p : A \leftrightarrow A$ is *transitive*, if $\forall a,b,c \in A : (a\ p\ b \wedge b\ p\ c) \Rightarrow a\ p\ c$

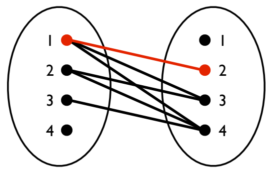
or...

R_p is reflexive if and only if $R_{p \circ p} \subseteq R_p$

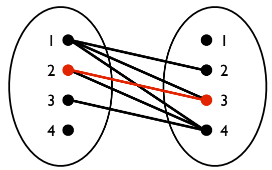
Example: $R_<$



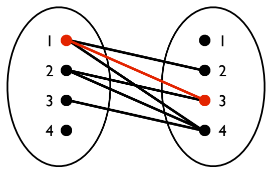
Properties of relations



and



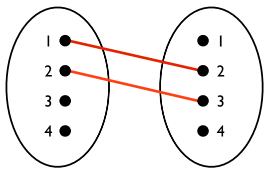
then...



$\forall a,b,c \in A : (a\ p\ b \wedge b\ p\ c) \Rightarrow a\ p\ c$ (transitive relation!)

Properties of relations

Example of *not* transitive:



R = { (1,2), (2,3) }

Because in the relation we have (1,2) and (2,3), but we do not have (1,3).

Equivalence relations

Definition:

A relation is an *equivalence relation*, if it is

- reflexive
- symmetric
- transitive

Equivalence relations

Consider the relation R_{age} : People ↔ People

where x age y ⇔ x is the same age in years as y.

It is reflexive because a person is the same age as himself.

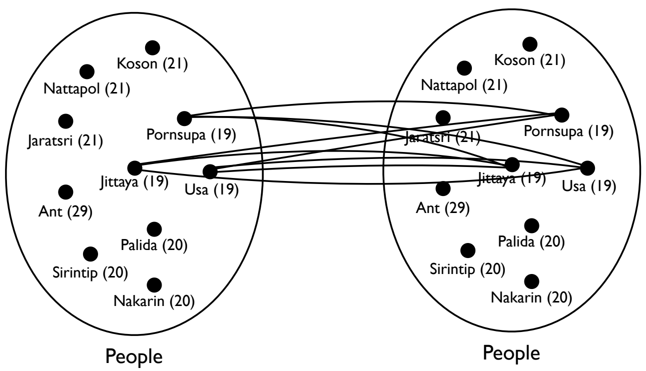
It is symmetric because if person A is the same age as persion B then person B is also the same age as person A.

It is transitive because if person A is the same age as person B and person B is the same age as person C, then person A is also the same age as person C.

It is reflexive, symmetric and transitive, so it is an *equivalence relation*.

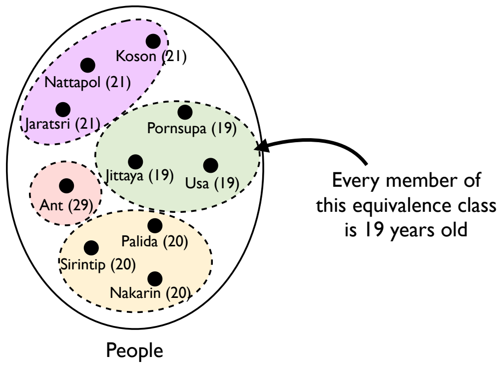
Equivalence relations

Equivalence relations have a special property that you can group the elements into *equivalence classes*.



Equivalence relations

Equivalence classes for R_{age} :



Partial orderings

Definition:

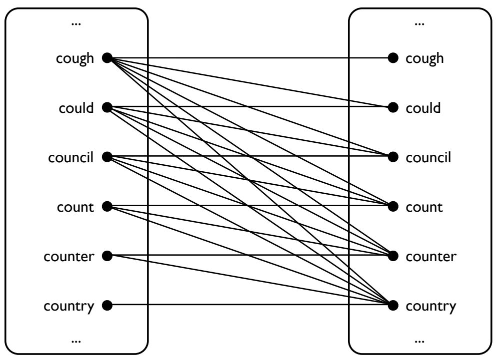
A relation is a *partial order,* if it is

- reflexive
- antisymmetric
- transitive

Partial orderings

Consider the relation $R_{\leq}$: Words ↔ Words

where $a \leq b \Leftrightarrow b$ does not come before a in the English dictionary.



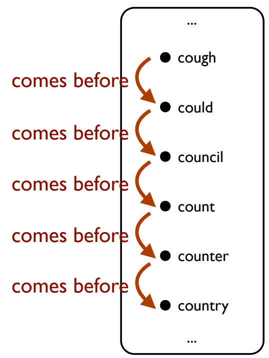
✓ reflexive

✓ antisymmetric

✓ transitive

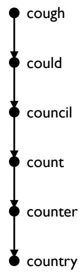
Partial orderings

We can think of the relation as an order on the set of Words...



Partial orderings

A partial order is often drawn using a Hasse diagram:



Partial orderings

A relation R_p is a *total order* if is a partial order and all the elements are comparable (i.e. for all a, b $\in$ A, either a p b or b p a).

The relation $R_{\leq}$: Words $\leftrightarrow$ Words is a total order.

Vocabulary

Relation

Composition

Reflexive

Symmetric

Antisymmetric

Transitive

Equivalence relation

Partial order

Total order

?

Relations

If we consider the "not equals" relation on natural numbers:

$$R_{!=} : N \leftrightarrow N$$

Which of the following properties does it have?

- reflexive
- symmetric
- antisymmetric
- transitive

Relations

Example:

3 != 3   FALSE

∴ not reflexive

Relations

Example:

3 != 5   TRUE

and

5 != 3   TRUE

∴ symmetric

Relations

Example:

4 != 6   TRUE

and

6 != 4   TRUE

∴ not antisymmetric

Relations

Example:

7 != 8   TRUE

and

8 != 9   TRUE

hence

7 != 9   TRUE

∴ ???

Relations

Example:

7 != 8 TRUE

and

8 != 7 TRUE

hence

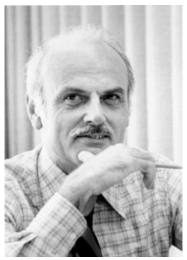
7 != 7 FALSE

∴ not transitive

Applications of relations

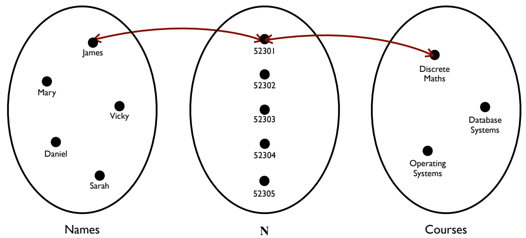
Relational databases

Relational databases (or Relational Database Management Systems - RDBMS) are based on relational theory.



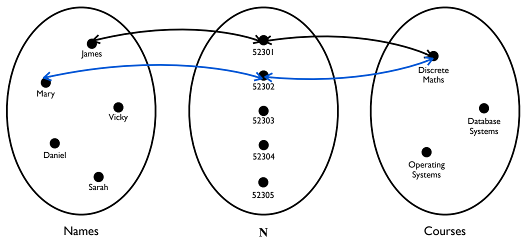
E.F. Codd (1923-2003) was a British computer scientist who invented the relational model for database systems.

Relational databases



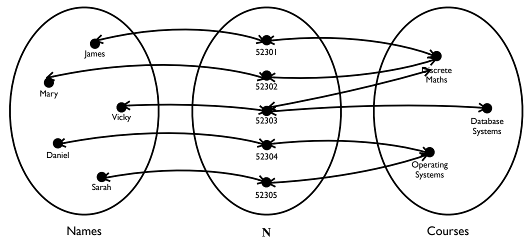
R = { (James, 52301, Discrete Maths) }

Relational databases



R = { (James, 52301, Discrete Maths),
(Mary, 52302, Discrete Maths) }

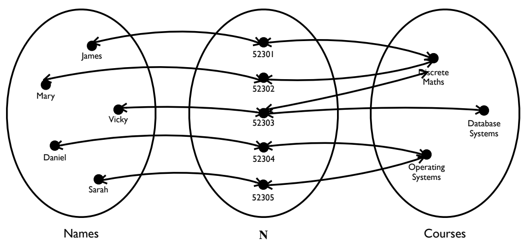
Relational databases



R : Names ↔ $\mathbb{N}$ ↔ Courses

R is the relationship between a name, a student number and a course that the student is taking.

Relational databases



R ⊆ Names × $\mathbb{N}$ × Courses

R is a set of sequences of length 3.

E.g. R = { (x_1, x_2, x_3), (y_1, y_2, y_3), ... }

Relational databases

R = {

(James,	52301,	Discrete Maths),
(Mary,	52302,	Discrete Maths),
(Vicky,	52303,	Discrete Maths),
(Vicky,	52303,	Database Systems),
(Daniel,	52304,	Operating Systems),
(Sarah,	52305,	Operating Systems)

}

Relational databases

Name	StudentId	Course
James	52301	Discrete Maths
Mary	52302	Discrete Maths
Vicky	52303	Discrete Maths
Vicky	52303	Database Systems
Daniel	52304	Operating Systems
Sarah	52305	Operating Systems

A relation between 3 sets

A sequence (or tuple)

Members of the set 'Courses'

Database languages

The most common language for databases is SQL (Structured Query Language). But there are others!

ISBL (Information Systems Base Language) was an early language developed at IBM. Not used now, but it is interesting because the syntax is similar to the set theory notation.

Relational operations

ISBL operation	Name	Equivalent set operation
A + B	Union	$A \cup B$
A . B	Intersection	$A \cap B$
A - B	Difference	$A \setminus B$
A % a, b, c	Projection	-
A : f	Restriction	$\{ x \in A \mid f(x) \}$
A * B	Join	Similar to $A \circ B$