

# Die relationale Sprache **TITUR<sub>EL</sub>** in mehreren Fallstudien

Gunther Schmidt

Institut für Softwaretechnologie

Fakultät für Informatik

[Gunther.Schmidt@UniBw.DE](mailto:Gunther.Schmidt@UniBw.DE)

# Outline

- 1. Motivation for a relational language**
2. Aims
3. Architecture
4. Substrate on which to build
5. Language details
6. Case study: Measure and Order
7. Case study: De Morgan Triples
8. Case study: Relational Integration

# Modelling with matrices by engineers

## Engineering

...

surface structures

...

steel frame structures

...

metal matrix  
composites

...

metal foams

## Finite elements

solve matrix eqs

$$Ax = b$$

determine

eigenvalues

$$\text{from } Ax = \lambda x$$

## Standard software

Mathematics + computing



## COST 274: TARSKI

Theory and Applications of Relational Structures  
as Knowledge Instruments

Chair: Gunther Schmidt

Vice Chair: Harrie de Swart

WA 1: Ewa Orłowska (Warsaw)

Algebr. and logical foundations of real world relations

WA 2: Gunther Schmidt (Munich)

Mechanization of relational reasoning

WA 3: Marc Roubens (Liège)

Relational scaling and preferences

WA 4: Tony Cohn (Leeds) / Ivo Düntsch (St. Catharines)

Relational reasoning in qualitative Physics

# Modelling with relations by application people

forest damage  
health services  
image processing  
multi-criteria  
    decision aid  
voting schemata  
knowledge  
    engineering  
spatial information  
data mining

vague  
fuzzy  
spatial  
temporal  
uncertain  
rough  
qualitative  
discrete

Private  
programs  
**Standard  
software  
unavailable  
or unknown**

Private  
programs

---

Logics and mathematics

# Situation as it deserves to be ...

## Modelling with relations by application people

forest damage  
health services  
image processing  
multi-criteria  
    decision aid  
voting schemata  
knowledge  
    engineering  
spatial information  
data mining

Common  
relational  
language

Standard  
software

---

Logics and mathematics

# Outline

1. Motivation for a relational language

## **2. Aims**

3. Architecture
4. Substrate on which to build
5. Language details
6. Case study: Measure and Order
7. Case study: De Morgan Triples
8. Case study: Relational Integration

# Aims in designing **TITUREL**

- Formulate all problems so far tackled with relational methods
- Transform relational terms and formulae in order to optimize them
- Interpret the relational constructs as boolean matrices, in RELVIEW, in the **TITUREL** substrate, or in RATH
- Prove relational formulae with system support in the style of RALF or Rasiowa-Sikorski
- Translate relational formulae into TeX-representation, or to first-order predicate logic, e.g.

# Outline

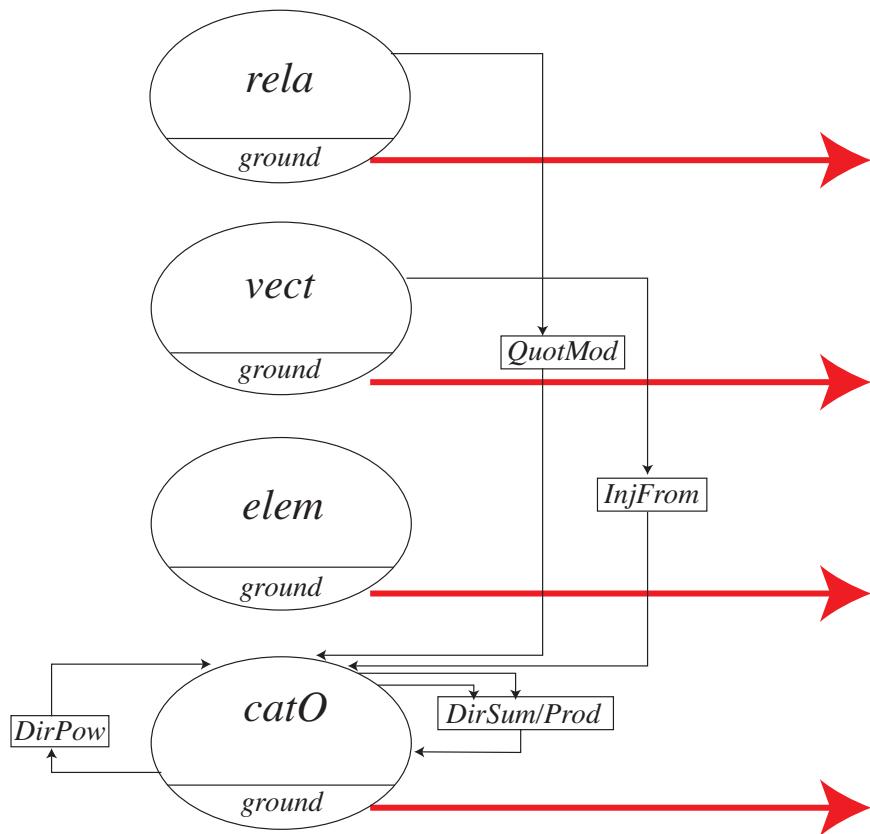
1. Motivation for a relational language
2. Aims
- 3. Architecture**
4. Substrate on which to build
5. Language details
6. Case study: Measure and Order
7. Case study: De Morgan Triples
8. Case study: Relational Integration

# *Language*

*theory*

*formulae*

*sets of rela,  
vect, elem*



# *Substrate*

*model*

*bool*

*sets of matrices,  
vectors, elements*

*bs + bs + matrix/set function/  
predicate/pairlist etc.*

*bs and predicate/marking/  
listing/powerset element etc.*

*bs with  
named/numbered element*

*finite baseset with  
named/numbered elements*

# Outline

1. Motivation for a relational language
2. Aims
3. Architecture
- 4. Substrate on which to build**
5. Language details
6. Case study: Measure and Order
7. Case study: De Morgan Triples
8. Case study: Relational Integration

**Basesets** are either **ground**,

which means given with their size

BS "8-Element-Set" 8

or with fully enumerated list (!) of element names

BSN "Politicians"

[Reagan, Clinton, Bush, Giscard, Mitterand,  
Chirac, Schmidt, Kohl, Schroeder, Thatcher,  
Major, Blair, Gonzales, Aznar, Zapatero],

or **constructed** via direct product, sum, power, or as dependent type, namely **quotient sets** or **extruded subsets**.

**Element denotation** is possible either as

## *Politicians*<sub>5</sub>

or as his name

*Mitterrand*

or marked in a vector or marked in a diagonal matrix

**Subset denotation** is possible either as

## LISTSet:

out of

## BSN Politicians =

[Bush, Chirac, Kohl, Blair]

or marked in a vector or marked in a diagonal matrix

# Relation as matrix or vector

	red	green	blue	orange
Mon	0	1	0	1
Tue	0	1	0	1
Wed	0	1	1	0
Thu	1	0	1	0
Fri	0	1	1	0
Sat	0	1	0	1
Sun	0	1	0	1

(Mon,red)	0
(Tue,red)	1
(Mon,green)	0
(Wed,red)	1
(Tue,green)	0
(Mon,blue)	1
(Thu,red)	0
(Wed,green)	1
(Tue,blue)	0
(Mon,orange)	1
(Fri,red)	1
(Thu,green)	0
(Wed,blue)	1
(Tue,orange)	0
(Sat,red)	1
(Fri,green)	0
(Thu,blue)	0
(Wed,orange)	1
(Sun,red)	1
(Sat,green)	0
(Fri,blue)	0
(Thu,orange)	1
(Sun,green)	0
(Sat,blue)	1
(Fri,orange)	0
(Sun,blue)	1
(Sat,orange)	0
(Sun,orange)	1

```

data BaseSet = BS String Int | BSN String [String]

data ElemInBaseSet = NUMBElem BaseSet Int      | MARKElem BaseSet [Bool] |
                     NAMEElem BaseSet String | DIAGElem BaseSet [[Bool]]

elemAsNUMB, elemAsMARK, elemAsNAME, elemAsDIAG

data SubSet = LISTSet BaseSet [Int]           | LINASet BaseSet [String] |
              PREDSet BaseSet (Int -> Bool) | POWESet BaseSet [Bool] |
              MARKSet BaseSet [Bool]          | DIAGSet BaseSet [[Bool]]

setAsLIST, setAsLINA, setAsPRED, setAsPOWE, setAsMARK, setAsDIAG

data Rel = MATRRel BaseSet BaseSet [[Bool]] |
           PREDRel BaseSet BaseSet (Int -> Int -> Bool) |
           SETFRel BaseSet BaseSet (Int -> [Int]) |
           SNAFRel BaseSet BaseSet (Int -> [String]) |
           PALIRel BaseSet BaseSet [(Int,Int)] |
           VECTRel BaseSet BaseSet [Bool] |
           POWERel BaseSet BaseSet [Bool]

data FuncOnBaseSets = MATRFunc BaseSet BaseSet [[Bool]] |
                      LISTFunc BaseSet BaseSet [Int]

data Fct2OnBaseSets = TABUFct2 BaseSet BaseSet BaseSet [[Int]] |
                      MATRFct2 BaseSet BaseSet BaseSet [[Bool]]

relAsMATRRel, ..., funcAsMATR, funcAsLIST, fct2AsTABU, fct2AsMATR

```

# Direct product with projection ordered rowwisely

$$\pi = \begin{array}{c} \text{Ace} \quad \text{King} \quad \text{Queen} \quad \text{Jack} \quad \text{Ten} \\ \hline (\text{Ace}, \spadesuit) & \begin{matrix} \color{red}{1} & 0 & 0 & 0 & 0 \end{matrix} \\ (\text{Ace}, \heartsuit) & \begin{matrix} \color{red}{1} & 0 & 0 & 0 & 0 \end{matrix} \\ (\text{Ace}, \diamondsuit) & \begin{matrix} \color{red}{1} & 0 & 0 & 0 & 0 \end{matrix} \\ (\text{Ace}, \clubsuit) & \begin{matrix} \color{red}{1} & 0 & 0 & 0 & 0 \end{matrix} \\ (\text{King}, \spadesuit) & \begin{matrix} 0 & \color{red}{1} & 0 & 0 & 0 \end{matrix} \\ (\text{King}, \heartsuit) & \begin{matrix} 0 & \color{red}{1} & 0 & 0 & 0 \end{matrix} \\ (\text{King}, \diamondsuit) & \begin{matrix} 0 & \color{red}{1} & 0 & 0 & 0 \end{matrix} \\ (\text{King}, \clubsuit) & \begin{matrix} 0 & \color{red}{1} & 0 & 0 & 0 \end{matrix} \\ (\text{Queen}, \spadesuit) & \begin{matrix} 0 & 0 & \color{red}{1} & 0 & 0 \end{matrix} \\ (\text{Queen}, \heartsuit) & \begin{matrix} 0 & 0 & \color{red}{1} & 0 & 0 \end{matrix} \\ (\text{Queen}, \diamondsuit) & \begin{matrix} 0 & 0 & \color{red}{1} & 0 & 0 \end{matrix} \\ (\text{Queen}, \clubsuit) & \begin{matrix} 0 & 0 & \color{red}{1} & 0 & 0 \end{matrix} \\ (\text{Jack}, \spadesuit) & \begin{matrix} 0 & 0 & 0 & \color{red}{1} & 0 \end{matrix} \\ (\text{Jack}, \heartsuit) & \begin{matrix} 0 & 0 & 0 & \color{red}{1} & 0 \end{matrix} \\ (\text{Jack}, \diamondsuit) & \begin{matrix} 0 & 0 & 0 & \color{red}{1} & 0 \end{matrix} \\ (\text{Jack}, \clubsuit) & \begin{matrix} 0 & 0 & 0 & \color{red}{1} & 0 \end{matrix} \\ (\text{Ten}, \spadesuit) & \begin{matrix} 0 & 0 & 0 & 0 & \color{red}{1} \end{matrix} \\ (\text{Ten}, \heartsuit) & \begin{matrix} 0 & 0 & 0 & 0 & \color{red}{1} \end{matrix} \\ (\text{Ten}, \diamondsuit) & \begin{matrix} 0 & 0 & 0 & 0 & \color{red}{1} \end{matrix} \\ (\text{Ten}, \clubsuit) & \begin{matrix} 0 & 0 & 0 & 0 & \color{red}{1} \end{matrix} \end{array}$$

$$\rho = \begin{array}{c} \clubsuit \quad \spadesuit \quad \heartsuit \quad \diamondsuit \\ \hline (\text{Ace}, \spadesuit) & \begin{matrix} \color{red}{1} & 0 & 0 & 0 \end{matrix} \\ (\text{Ace}, \heartsuit) & \begin{matrix} 0 & \color{red}{1} & 0 & 0 \end{matrix} \\ (\text{Ace}, \diamondsuit) & \begin{matrix} 0 & 0 & \color{red}{1} & 0 \end{matrix} \\ (\text{Ace}, \clubsuit) & \begin{matrix} 0 & 0 & 0 & \color{red}{1} \end{matrix} \\ (\text{King}, \spadesuit) & \begin{matrix} \color{red}{1} & 0 & 0 & 0 \end{matrix} \\ (\text{King}, \heartsuit) & \begin{matrix} 0 & \color{red}{1} & 0 & 0 \end{matrix} \\ (\text{King}, \diamondsuit) & \begin{matrix} 0 & 0 & \color{red}{1} & 0 \end{matrix} \\ (\text{King}, \clubsuit) & \begin{matrix} 0 & 0 & 0 & \color{red}{1} \end{matrix} \\ (\text{Queen}, \spadesuit) & \begin{matrix} \color{red}{1} & 0 & 0 & 0 \end{matrix} \\ (\text{Queen}, \heartsuit) & \begin{matrix} 0 & \color{red}{1} & 0 & 0 \end{matrix} \\ (\text{Queen}, \diamondsuit) & \begin{matrix} 0 & 0 & \color{red}{1} & 0 \end{matrix} \\ (\text{Queen}, \clubsuit) & \begin{matrix} 0 & 0 & 0 & \color{red}{1} \end{matrix} \\ (\text{Jack}, \spadesuit) & \begin{matrix} \color{red}{1} & 0 & 0 & 0 \end{matrix} \\ (\text{Jack}, \heartsuit) & \begin{matrix} 0 & \color{red}{1} & 0 & 0 \end{matrix} \\ (\text{Jack}, \diamondsuit) & \begin{matrix} 0 & 0 & \color{red}{1} & 0 \end{matrix} \\ (\text{Jack}, \clubsuit) & \begin{matrix} 0 & 0 & 0 & \color{red}{1} \end{matrix} \\ (\text{Ten}, \spadesuit) & \begin{matrix} \color{red}{1} & 0 & 0 & 0 \end{matrix} \\ (\text{Ten}, \heartsuit) & \begin{matrix} 0 & \color{red}{1} & 0 & 0 \end{matrix} \\ (\text{Ten}, \diamondsuit) & \begin{matrix} 0 & 0 & \color{red}{1} & 0 \end{matrix} \\ (\text{Ten}, \clubsuit) & \begin{matrix} 0 & 0 & 0 & \color{red}{1} \end{matrix} \end{array}$$

# Projection relations diagonally

	Ace	King	Queen	Jack	Ten		♠	♡	♦	♣
(Ace, ♠)	1	0	0	0	0	(Ace, ♠)	1	0	0	0
(King, ♠)	0	1	0	0	0	(King, ♠)	1	0	0	0
(Ace, ♡)	1	0	0	0	0	(Ace, ♡)	0	1	0	0
(Queen, ♠)	0	0	1	0	0	(Queen, ♠)	1	0	0	0
(King, ♡)	0	1	0	0	0	(King, ♡)	0	1	0	0
(Ace, ♦)	1	0	0	0	0	(Ace, ♦)	0	0	1	0
(Jack, ♠)	0	0	0	1	0	(Jack, ♠)	1	0	0	0
(Queen, ♡)	0	0	1	0	0	(Queen, ♡)	0	1	0	0
(King, ♦)	0	1	0	0	0	(King, ♦)	0	0	1	0
(Ace, ♣)	1	0	0	0	0	(Ace, ♣)	0	0	0	1
(Ten, ♠)	0	0	0	0	1	(Ten, ♠)	1	0	0	0
(Jack, ♡)	0	0	0	1	0	(Jack, ♡)	0	1	0	0
(Queen, ♦)	0	0	1	0	0	(Queen, ♦)	0	0	1	0
(King, ♣)	0	1	0	0	0	(King, ♣)	0	0	0	1
(Ten, ♡)	0	0	0	0	1	(Ten, ♡)	0	1	0	0
(Jack, ♦)	0	0	0	1	0	(Jack, ♦)	0	0	1	0
(Queen, ♣)	0	0	1	0	0	(Queen, ♣)	0	0	0	1
(Ten, ♦)	0	0	0	0	1	(Ten, ♦)	0	0	1	0
(Jack, ♣)	0	0	0	1	0	(Jack, ♣)	0	0	0	1
(Ten, ♣)	0	0	0	0	1	(Ten, ♣)	0	0	0	1

## Direct sum with injection — alternating order

	<	Ace	<	King	<	Queen	<	Jack	<	Ten
♠	♣	♦	♥	♦	♣	♦	♥	♣	♦	♥
1	0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0	0

	<	Ace	<	King	<	Queen	<	Jack	<	Ten
♠	♣	♦	♥	♦	♣	♦	♥	♣	♦	♥
Ace	0	1	0	0	0	0	0	0	0	0
King	0	0	0	1	0	0	0	0	0	0
Queen	0	0	0	0	0	1	0	0	0	0
Jack	0	0	0	0	0	0	0	1	0	0
Ten	0	0	0	0	0	0	0	0	1	0

## Direct sum with injection — normal order

	<	<	<	<	>	Ace	<	King	<	Queen	<	Jack	<	Ten
♠	♣	♦	♥	♦	♣	♦	♥	♦	♣	♦	♥	♦	♣	♦
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0	0	0	0

	<	<	<	<	>	Ace	<	King	<	Queen	<	Jack	<	Ten
♠	♣	♦	♥	♦	♣	♦	♥	♦	♣	♦	♥	♦	♣	♦
Ace	0	0	0	0	1	0	0	0	0	0	0	0	0	0
King	0	0	0	0	0	1	0	0	0	0	0	0	0	0
Queen	0	0	0	0	0	0	1	0	0	0	0	0	0	0
Jack	0	0	0	0	0	0	0	1	0	0	0	0	0	0
Ten	0	0	0	0	0	0	0	0	1	0	0	0	0	0

The relation between a baseset and its powerset is generated **generically**.

$$\begin{array}{l}
 \text{Bush} \rightarrow \begin{pmatrix} 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix} \\
 \text{Chirac} \rightarrow \begin{pmatrix} \{\} & \{\text{Bush} \rightarrow\} & \{\text{Chirac} \rightarrow\} & \{\text{Bush} \rightarrow, \text{Chirac} \rightarrow\} & \{\text{Kohl} \rightarrow\} & \{\text{Bush} \rightarrow, \text{Kohl} \rightarrow\} & \{\text{Chirac} \rightarrow, \text{Kohl} \rightarrow\} & \{\text{Bush} \rightarrow, \text{Chirac} \rightarrow, \text{Kohl} \rightarrow\} & \{\text{Blair} \rightarrow\} & \{\text{Bush} \rightarrow, \text{Blair} \rightarrow\} & \{\text{Chirac} \rightarrow, \text{Blair} \rightarrow\} & \{\text{Bush} \rightarrow, \text{Chirac} \rightarrow, \text{Blair} \rightarrow\} & \{\text{Kohl} \rightarrow, \text{Blair} \rightarrow\} & \{\text{Bush} \rightarrow, \text{Kohl} \rightarrow, \text{Blair} \rightarrow\} & \{\text{Chirac} \rightarrow, \text{Kohl} \rightarrow, \text{Blair} \rightarrow\} & \{\text{Bush} \rightarrow, \text{Chirac} \rightarrow, \text{Kohl} \rightarrow, \text{Blair} \rightarrow\} \end{pmatrix} \\
 \text{Kohl} \rightarrow \\
 \text{Blair} \rightarrow
 \end{array}$$

When a vector term is given, one may construct another **dependent** baseset via **subset extrusion**

BSN Politiciansinjected =  
[Bush->, Chirac->, Kohl->, Blair->]

Its element denotations are slightly different, namely marked with →

There is also the **natural injection**

	Reagan	Clinton	Bush	Giscard	Mitterand	Chirac	Schmidt	Kohl	Schroeder	Major	Thatcher	Blair	Gonzales	Aznar	Zapatero
Bush→	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
Chirac→	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
Kohl→	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
Blair→	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0

When an equivalence is given, one may construct another **dependent** baseset as a **quotient**

$\{[\text{Reagan}],[\text{Giscard}],[\text{Schmidt}],[\text{Thatcher}],[\text{Gonzales}]\}$

There is also the **natural projection**

	Reagan	Clinton	Bush	Giscard	Mitterand	Chirac	Schmidt	Kohl	Schroeder	Thatcher	Major	Blair	Gonzales	Aznar	Zapatero
Reagan	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
Clinton	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
Bush	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
Giscard	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0
Mitterand	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0
Chirac	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0
Schmidt	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0
Kohl	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0
Schroeder	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0
Thatcher	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0
Major	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0
Blair	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0
Gonzales	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0
Aznar	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0
Zapatero	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0

	Reagan	Giscard	Schmidt	Thatcher	Gonzales
Reagan	1	0	0	0	0
Clinton	1	0	0	0	0
Bush	1	0	0	0	0
Giscard	0	1	0	0	0
Mitterand	0	1	0	0	0
Chirac	0	1	0	0	0
Schmidt	0	0	1	0	0
Kohl	0	0	1	0	0
Schroeder	0	0	1	0	0
Thatcher	0	0	0	1	0
Major	0	0	0	1	0
Blair	0	0	0	1	0
Gonzales	0	0	0	0	1
Aznar	0	0	0	0	1
Zapatero	0	0	0	0	1

# Outline

1. Motivation for a relational language
2. Aims
3. Architecture
4. Substrate on which to build
- 5. Language details**
6. Case study: Measure and Order
7. Case study: De Morgan Triples
8. Case study: Relational Integration

## Reference form      vs.      T<sub>E</sub>X form

A :\*\*\*: (NegaR B :|||: (Convs C :&&&: D))

upon output translated to       $A; (\overline{B} \cup (C^\top \cap D))$

## Reference form      vs.      T<sub>E</sub>X form

A :\*\*\*: (NegaR B :|||: (Convs C :&&&: D))

upon output translated to       $A; (\overline{B} \cup (C^\top \cap D))$

NullR d c :\*\*\*: (Ident c) :|||: (UnivR d c)

long vs. short form:     $\perp_{d,c}; \mathbb{I}_c \cup \top_{d,c}$     vs.     $\perp; \mathbb{I} \cup \top$

## Reference form vs. T<sub>E</sub>X form

A :\*\*\*: (NegaR B :|||: (Convs C :&&&: D))

upon output translated to  $A; (\overline{B} \cup (C^\top \cap D))$

NullR d c :\*\*\*: (Ident c) :|||: (UnivR d c)

long vs. short form:  $\perp_{d,c}; \mathbb{I}_c \cup \top_{d,c}$  vs.  $\perp; \mathbb{I} \cup \top$

Convs (Pi d1 d2) :\*\*\*: (Rho d1 d2)

$\pi_{d1,d2}^\top; \rho_{d1,d2}$  or  $\pi^\top; \rho$  available generically!

## Reference form vs. T<sub>E</sub>X form

A :\*\*\*: (NegaR B :|||: (Convs C :&&&: D))

upon output translated to  $A; (\overline{B} \cup (C^\top \cap D))$

NullR d c :\*\*\*: (Ident c) :|||: (UnivR d c)

long vs. short form:  $\perp_{d,c}; \mathbb{I}_c \cup \top_{d,c}$  vs.  $\perp; \mathbb{I} \cup \top$

Convs (Pi d1 d2) :\*\*\*: (Rho d1 d2)

$\pi_{d1,d2}^\top; \rho_{d1,d2}$  or  $\pi^\top; \rho$  available generically!

(Iota d1 d2) :\*\*\*: (Convs (Kappa d1 d2))

$\iota_{d1,d2}; \kappa_{d1,d2}^\top$  or  $\iota; \kappa^\top$  available generically!

# Concepts of the language **TITUR<sub>E</sub>L**

constants, variables,  $\neg$ ,  $\cap$ ,  $\cup$ ,  $:$ ,  $\subseteq$  and the generic constructs  $\pi$ ,  $\rho$  (for direct product) and  $\iota$ ,  $\kappa$  (for direct sum)

# Concepts of the language **TITUR<sub>E</sub>L**

constants, variables,  $\neg$ ,  $\cap$ ,  $\cup$ ,  $:$ ,  $\subseteq$  and the generic constructs  $\pi$ ,  $\rho$  (for direct product) and  $\iota$ ,  $\kappa$  (for direct sum)

Epsi d generic membership relation  $\varepsilon : D \longrightarrow \mathcal{P}(D)$

v : ||--: w column vector  $\times$  row vector giving a relation

# Concepts of the language **TITUR<sub>E</sub>L**

constants, variables,  $\neg, \cap, \cup, :, \subseteq$  and the generic constructs  $\pi, \rho$  (for direct product) and  $\iota, \kappa$  (for direct sum)

Epsi d generic membership relation  $\varepsilon : D \leftrightarrow \mathcal{P}(D)$

v :||--: w column vector  $\times$  row vector giving a relation

SyQ r s symmetric quotient

SupRela relationSet

# Concepts of the language **TITUR<sub>E</sub>L**

constants, variables,  $\neg$ ,  $\cap$ ,  $\cup$ ,  $:$ ,  $\subseteq$  and the generic constructs  $\pi$ ,  $\rho$  (for direct product) and  $\iota$ ,  $\kappa$  (for direct sum)

**Epsi** d generic membership relation  $\varepsilon : D \leftrightarrow \mathcal{P}(D)$

**v** : ||--: w column vector  $\times$  row vector giving a relation

**SyQ** r s symmetric quotient

**SupRela** relationSet

**Project** equRel generic natural projection

**InjTerm** v generic conversion of vector to subset extrusion

# Concepts of the language **TITUR<sub>E</sub>L**

constants, variables,  $\neg, \cap, \cup, :, \subseteq$  and the generic constructs  $\pi, \rho$  (for direct product) and  $\iota, \kappa$  (for direct sum)

**Epsi** d generic membership relation  $\varepsilon : D \leftrightarrow \mathcal{P}(D)$

**v** :||--: w column vector  $\times$  row vector giving a relation

**SyQ** r s symmetric quotient

**SupRela** relationSet

**Project** equRel generic natural projection

**InjTerm** v generic conversion of vector to subset extrusion

**PointDiag** e      **FuncToRela** f      **Fct2ToRela** f2

**ProdVectToRela** v  $\mathbb{B}^{n \times m}$  vector  $\longrightarrow n \times m$  relation

**RFctAppl** rf arg applies term-given relation function

and others as demonic operators, e.g.

# Concepts of the language **TITUR**

In addition we have operations

# Concepts of the language **TITUREL**

In addition we have operations

on elements:

element variables and constants, pairs of elements,

**That**, **Some** selection constructs

**VectToPowElem** v converts vector to powerset element

# Concepts of the language **TITUREL**

In addition we have operations

on elements:

element variables and constants, pairs of elements,

**That**, **Some** selection constructs

**VectToPowElem** v converts vector to powerset element

and on vectors :

vector constants and variables,  $\neg$ ,  $\cap$ ,  $\cup$ ,  $\subseteq$ ,  $\top$ ,  $\perp$

**r :\*\*\*\*: v** Peirce product

**SupVect vectorSet**

**PointVect e**

**RelaToVect r**

```
transitiveClosureFunction rt =
let o    = domRT rt
  rv1 = VarR "rv1" o o;      r1  = RV rv1
  rv2 = VarR "rv2" o o;      r2  = RV rv2
  f   = RFCT "f" (RVar rv1)
        (r1 :|||: (r1 :***: r1))
  c   = COND "c" (RVar rv2)
        (RF $ RFctAppl f (ArgR r2) :====: r2)
in SupRela $ RI Incr rt f c
```

```

transClosAbstracted =
let o      = OV $ VarO "Domain"
    rv0 = VarR "rv0" o o;   r0 = RV rv0
    rv1 = VarR "rv1" o o;   r1 = RV rv1
    rv2 = VarR "rv2" o o;   r2 = RV rv2
    f = RFCT "f" (RVar rv1)
        (r1 :|||: (r1 :***: r1))
    c = COND "c" (RVar rv2)
        (RF $ RFctAppl f (ArgR r2) :===: r2)
in RFCT "transClos" (RVar rv0)
    (SupRela $ RI Incr r0 f c)

```

Called for the term `rt` with

`RFctAppl transClosAbstracted (ArgR rt)`

# Outline

1. Motivation for a relational language
2. Aims
3. Architecture
4. Substrate on which to build
5. Language details
- 6. Case study: Measure and Order**
7. Case study: De Morgan Triples
8. Case study: Relational Integration

**Measuring** traditionally means to attach some value  $v \in \mathbb{R}$

We are largely unaware of the problems in finding an adequate scale as this has mostly been achieved earlier in history.

Even in Physics there occur linear transformations

$$\phi(x) := \alpha x + \beta, \alpha > 0 \text{ on } \mathbb{R}$$

between Fahrenheit, Celsius, Reaumur, but also

logarithms

as for Kelvin and Dezibel

In the 1680's, when switching Julian  $\rightarrow$  Gregorian calendar  
translation along  $\mathbb{N}$

Measuring may be purely nominal by attachment of linguistic variables, even if these are small numbers

Mohs scale for hardness of minerals “ $A$  writes on  $B$ ”

Hurricane scale 1..5

Storm power scale 1..12

Earthquake Richter scale 1.. open upper end

Raw scores of intelligence tests often undergo monotone transformations  $x \leq y \iff \phi(x) \leq \phi(y)$ .

Sometimes physical values are related to a psychological quantity as for loudness with frequency and intensity. If one achieves to arrive at a scale where important operations work additive, e.g., such a scale will then get high acceptance. This is known as the problem of conjoint measurement.

**Definition.** A relation  $R : X \longrightarrow Y$  is said to be

- i)  **$\mathbb{R}$ -realizable** wrt. “ $\leq$ ” if there exist two mappings  $f : X \longrightarrow \mathbb{R}$  and  $g : Y \longrightarrow \mathbb{R}$  such that
$$(x, y) \in R \iff f(x) \leq g(y)$$
- ii) **realizable**, if there exists a linear order  $E$  together with two mappings  $f, g$  such that
$$R = f; E; g^\top.$$

# Mathematicians who worked on measuring and realizability in $\mathbb{R}$

Garrett Birkhoff, Harvard

Peter C. Fishburn, Princeton

R. Duncan Luce, Johns Hopkins

Dana Scott, varying

Patrick Suppes, Stanford

Everyone of these is a multiple book author

## Theorem of Birkhoff-Milgram.

Let  $C$  be a linear strictorder. Then there exists a mapping  $f$  into the real numbers such that

$$(x, y) \in C \iff f(x) < f(y)$$

precisely when  $C$  has a countable order-dense subset.

**Definition.** A (necessarily homogeneous) relation

$R$  is a **semiorder**

: $\iff$   $R$  is irreflexive, Ferrers, and semi-transitive

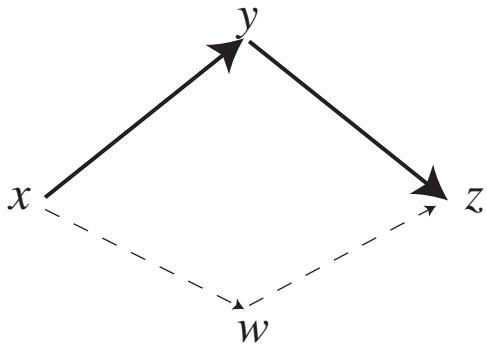
**Theorem of Scott-Suppes.**

A relation  $R \subseteq X \times X$  is a finite semiorder precisely when there exists

- mapping  $f : X \longrightarrow \mathbb{R}$  satisfying  
 $x < y \iff f(x) + 1 < f(y)$
- a linear mapping  $f$  into a strictorder  $C$  such that  
 $R = f; C^2; f^\top$ .

## Semi-transitivity shown with dotted arrow convention

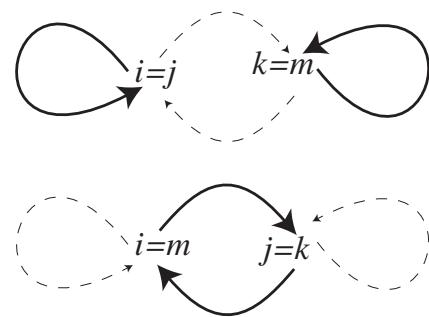
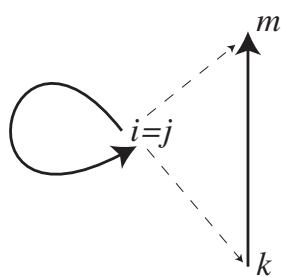
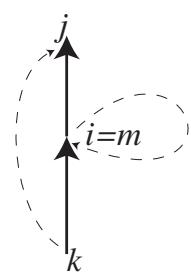
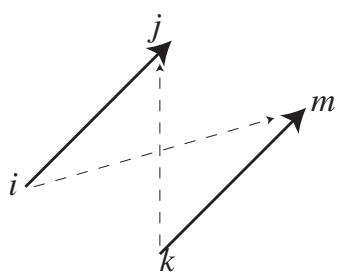
$$\begin{aligned} R \text{ semi-transitive} &\iff R; R; \overline{R}^T \subseteq R \\ &\iff \overline{R}; \overline{R} \subseteq \overline{R; R} \end{aligned}$$



$$R \text{ Ferrers} \iff R; \overline{R}^T; R \subseteq R$$

$$\text{fringe}(R) := R \cap \overline{R; \overline{R}^T; R}$$

**Ferrers property** shown with dotted arrow convention



	$j$		$m$	
	↓		↓	
1	1	2	3	4
2	1	1	1	0
3	0	0	0	0
4	0	0	0	0
5	1	1	1	0
6	1	1	1	0
7	1	1	1	0
8	1	1	1	0
9	0	0	0	0
10	1	1	1	0
11	1	1	1	0
12	0	0	0	0

**Norman Macleod Ferrers** (\*1829, †1903)

British mathematician

at Cambridge Gonville and Caius College,  
vice chancellor of Cambridge University 1884

Remembered mainly for pointing out a conjugacy  
in integer partition diagrams, which are accordingly  
called Ferrers graphs and are closely related to Young  
diagrams.

— N. M. Ferrers, An Elementary Treatise on  
Trilinear Coordinates (London, 1861)

The spelling “Ferrars” is sometimes also used.

**Proposition.** If an ordering  $E$  is Ferrers, then also its corresponding irreflexive version  $C := \overline{\mathbb{I}} \cap E$  — but not necessarily vice versa.

**Proof:**  $C; \overline{C}^\top; C$

$$\begin{aligned} &= (\mathbb{I} \cap E); \overline{\mathbb{I} \cap E}^\top; (\mathbb{I} \cap E) \\ &= (\mathbb{I} \cap E); (\mathbb{I} \cup \overline{E}^\top); (\mathbb{I} \cap E) \\ &= (\mathbb{I} \cap E); (\mathbb{I} \cap E) \cup (\mathbb{I} \cap E); \overline{E}^\top; (\mathbb{I} \cap E) \subseteq E \end{aligned}$$

The left part is contained in  $\overline{\mathbb{I}}$  using antisymmetry:

$$\begin{aligned} (\mathbb{I} \cap E)^2 &= (\overline{E} \cap \overline{E}^\top \cap E)^2 = ([\overline{E} \cup \overline{E}^\top] \cap E)^2 \\ &= (\overline{E}^\top \cap E)^2 \subseteq E; \overline{E}^\top \subseteq \overline{\mathbb{I}} \end{aligned}$$

The right part:  $(E; \overline{E}^\top); E \subseteq \overline{E}^\top; E \subseteq \overline{\mathbb{I}}$

**Counter example:**

The ordering relation on  $\mathbb{B}^2$ :

$$\begin{pmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad \begin{pmatrix} 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

$$R = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 1 & \textcolor{red}{1} & \textcolor{red}{1} & \textcolor{red}{1} & 0 & 0 & 0 \\ 2 & \textcolor{red}{1} & \textcolor{red}{1} & \textcolor{red}{1} & 0 & 0 & 0 \\ 3 & \textcolor{red}{1} & \textcolor{red}{1} & \textcolor{red}{1} & 0 & 0 & 0 \\ 4 & 0 & 0 & 0 & \textcolor{red}{1} & \textcolor{red}{1} & 0 \\ 5 & 0 & 0 & 0 & \textcolor{red}{1} & \textcolor{red}{1} & 0 \\ 6 & 0 & 0 & 0 & 0 & 0 & \textcolor{red}{1} & \textcolor{red}{1} \\ 7 & 0 & 0 & 0 & 0 & 0 & \textcolor{red}{1} & \textcolor{red}{1} \end{pmatrix}$$

	1	2	3	4	5	6	7
1	1	1	1	0	0	0	0
2	1	1	1	0	0	0	0
3	1	1	1	0	0	0	0
fringe( $R$ ) = 4	0	0	0	1	1	0	0
5	0	0	0	1	1	0	0
6	0	0	0	0	0	1	1
7	0	0	0	0	0	1	1

**block diagonal relation**

$$R = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 1 & \textcolor{red}{1} & \textcolor{red}{1} & \textcolor{red}{1} & \textcolor{red}{1} & \textcolor{red}{1} & \textcolor{red}{1} \\ 2 & \textcolor{red}{1} & \textcolor{red}{1} & \textcolor{red}{1} & \textcolor{red}{1} & \textcolor{red}{1} & \textcolor{red}{1} \\ 3 & \textcolor{red}{1} & \textcolor{red}{1} & \textcolor{red}{1} & \textcolor{red}{1} & \textcolor{red}{1} & \textcolor{red}{1} \\ 4 & 0 & 0 & 0 & \textcolor{red}{1} & \textcolor{red}{1} & \textcolor{red}{1} \\ 5 & 0 & 0 & 0 & \textcolor{red}{1} & \textcolor{red}{1} & \textcolor{red}{1} \\ 6 & 0 & 0 & 0 & 0 & 0 & \textcolor{red}{1} \\ 7 & 0 & 0 & 0 & 0 & 0 & \textcolor{red}{1} \end{pmatrix}$$

	1	2	3	4	5	6	7
1	1	1	1	0	0	0	0
2	1	1	1	0	0	0	0
3	1	1	1	0	0	0	0
fringe( $R$ ) = 4	0	0	0	1	1	0	0
5	0	0	0	1	1	0	0
6	0	0	0	0	0	1	1
7	0	0	0	0	0	1	1

## upper block triangle

$$R = \begin{pmatrix} & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 2 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 3 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 4 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 5 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 6 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 7 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

	1	2	3	4	5	6	7
1	0	0	0	1	1	0	0
2	0	0	0	1	1	0	0
3	0	0	0	1	1	0	0
fringe( $R$ ) = 4	0	0	0	0	0	1	1
5	0	0	0	0	0	1	1
6	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0

# irreflexive upper block triangle

$$R = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & \infty \\ 1 & \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix} & & & & & & \\ 2 & \begin{pmatrix} 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix} & & & & & & \\ 3 & \begin{pmatrix} 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \end{pmatrix} & & & & & & \\ 4 & \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{pmatrix} & & & & & & \\ 5 & \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix} & & & & & & \\ 6 & \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \end{pmatrix} & & & & & & \\ 7 & \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix} & & & & & & \\ 8 & \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} & & & & & & \end{pmatrix}$$

$$R = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 \\ 1 & \textcolor{red}{1} \\ 2 & \textcolor{red}{1} \\ 3 & 0 & 0 & 0 & \textcolor{red}{1} & \textcolor{red}{1} & 0 & 0 & 0 & 0 & \textcolor{red}{1} & \textcolor{red}{1} & \textcolor{red}{1} \\ 4 & 0 & 0 & 0 & \textcolor{red}{1} & \textcolor{red}{1} & 0 & 0 & 0 & 0 & \textcolor{red}{1} & \textcolor{red}{1} & \textcolor{red}{1} \\ 5 & 0 & 0 & 0 & 0 & 0 & \textcolor{red}{1} \\ 6 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \textcolor{red}{1} & \textcolor{red}{1} & \textcolor{red}{1} \\ 7 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \textcolor{red}{1} & \textcolor{red}{1} & \textcolor{red}{1} \end{pmatrix}$$

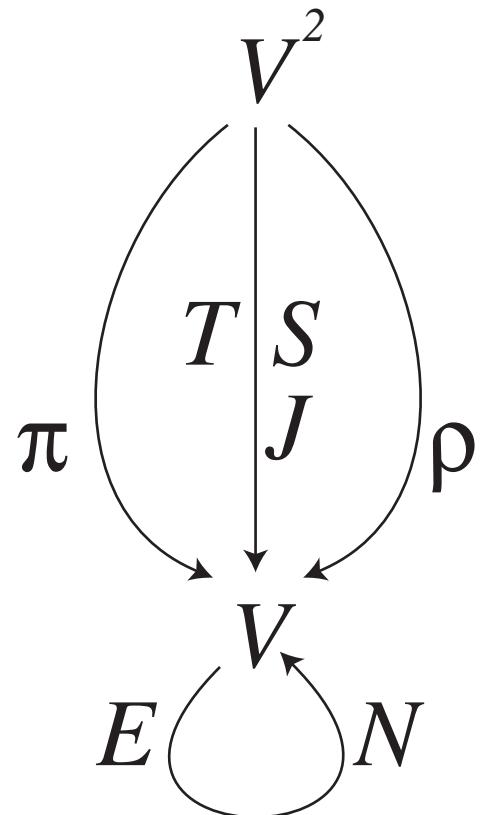
	1	2	3	4	5	6	7	8	9	10	11	12	13
1	1	1	1	0	0	0	0	0	0	0	0	0	0
2	1	1	1	0	0	0	0	0	0	0	0	0	0
3	0	0	0	1	1	0	0	0	0	0	0	0	0
fringe( $R$ ) = 4	0	0	0	1	1	0	0	0	0	0	0	0	0
5	0	0	0	0	0	1	1	1	1	0	0	0	0
6	0	0	0	0	0	0	0	0	0	1	1	1	1
7	0	0	0	0	0	0	0	0	0	1	1	1	1

# order-shaped relation

# Outline

1. Motivation for a relational language
2. Aims
3. Architecture
4. Substrate on which to build
5. Language details
6. Case study: Measure and Order
- 7. Case study: De Morgan Triples**
8. Case study: Relational Integration

## Basic situation for a De Morgan triple



## Definition of a $t$ -norm

Given a set  $V$  of **values**, lattice-ordered by  $E$ , one will ask for a  $t$ -norm to work as a conjunction operator  $T$  and demand

*normalized*       $T(\top, x) = x$

*commutative*     $T(x, y) = T(y, x)$

*monotone*        $T(x, y) \leq T(u, v)$  whenever

$$\perp \leq x \leq u \leq \top, \perp \leq y \leq v \leq \top$$

*associative*       $T(x, T(y, z)) = T(T(x, y), z)$

Usually  $V := [0, 1] \subseteq \mathbb{R}$  is taken as the set of values.

We accept any **finite lattice**.

## Definition of a $t$ -conorm

Given a set  $V$  of **values**, ordered by  $E$ , one will ask for a  $t$ -conorm to work as a **disjunction operator**  $S$  and demand

*normalized*       $S(\perp, x) = x$

*commutative*     $S(x, y) = S(y, x)$

*monotone*        $S(x, y) \leq S(u, v)$  whenever

$$\perp \leq x \leq u \leq \top, \quad \perp \leq y \leq v \leq \top$$

*associative*       $S(x, S(y, z)) = S(S(x, y), z)$

## Definition of a (strict, strong) negation

Given a set  $V$  of **values**, ordered by  $E$ , one will ask for a negation to work as an complement operator  $N$  and demand

$$N(\perp) = \top$$

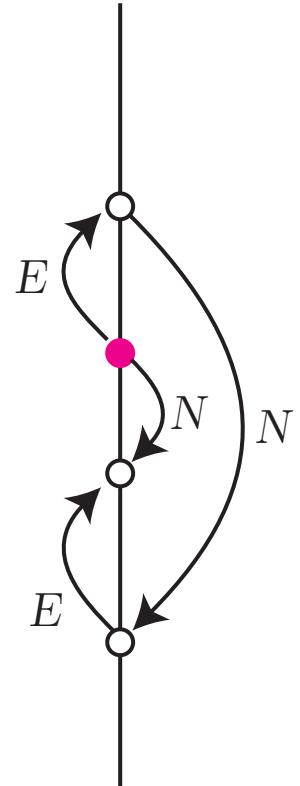
$$N(\top) = \perp$$

$$N(x) \geq N(y) \text{ whenever } x \leq y$$

$$N(x) > N(y) \text{ whenever } x < y \text{ (if strict)}$$

$$N(N(x)) = x \quad (\text{if in addition strong})$$

# Negation



$$e_0; e_1^\top \cup e_1; e_0^\top \subseteq N$$

where  $e_0 := \text{lub}_E(\mathbb{T})$ ,  $e_1 := \text{glb}_E(\mathbb{T})$

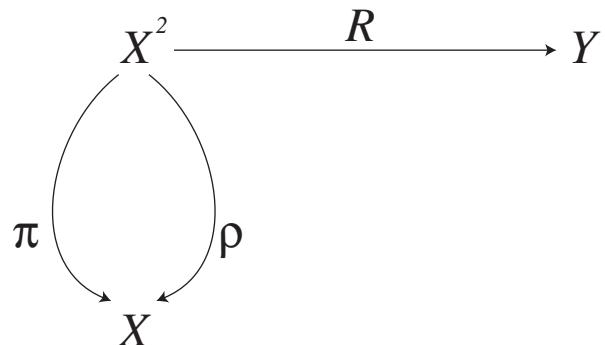
$$E; N \subseteq N; E^\top$$

$$C; N \subseteq N; C^\top$$

with  $C := \mathbb{I} \cap E$

$$N; N = \mathbb{I}$$

# Commutativity formulated component-free:

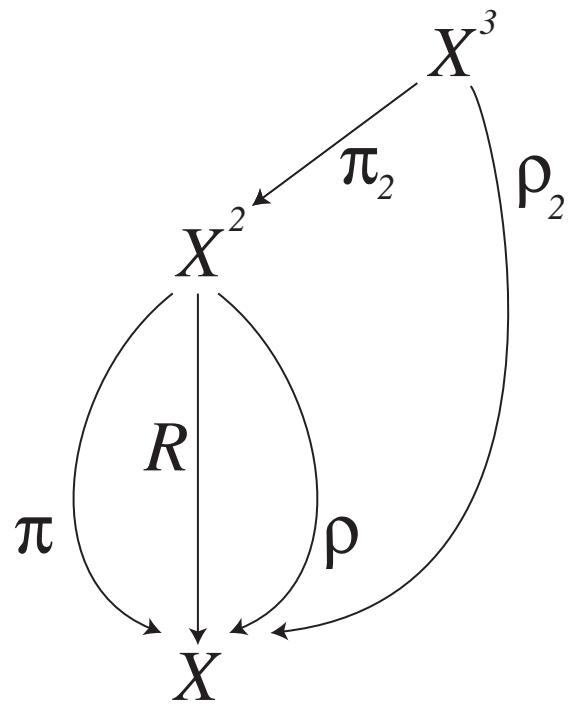


Result of  $R$  must not change  
with arguments flipped

$$\pi^\top; R = \rho^\top; R$$

```
isCommFuncFormula rt =
  case domRT rt of
    DirPro first secnd ->
      let pi    = Pi first secnd
          piT   = Convs pi
          rho   = Rho first secnd
          rhoT = Convs rho
      in  RF $ piT :***: rt :==: (rhoT :***: rt)
      _ -> error "Domain is not a product"
```

**Associativity** formulated component-free:



$$R(R(a, b), c) = R(a, R(b, c))$$

$$(\rho_2; \rho^T \cap \pi_2; R; \pi^T); R = (\pi_2; \pi; \pi^T \cap (\pi_2; \rho; \pi^T \cap \rho_2; \rho^T); R; \rho^T); R$$

## Associativity formulated component-free:

$$(\rho_2; \rho^\top \cap \pi_2; R; \pi^\top); R = (\pi_2; \pi; \pi^\top \cap (\pi_2; \rho; \pi^\top \cap \rho_2; \rho^\top); R; \rho^\top); R$$

isAssocFuncFormula r = case domRT r of

DirPro first secnd ->

```
let p      = domRT r;      o      = codRT r
    pi     = Pi   o o;    piT     = Convs pi
    rho    = Rho  o o;    rhoT    = Convs rho
    pi2   = Pi   p o;    pi2T   = Convs pi2
    rho2  = Rho  p o;    rho2T  = Convs rho2
    leS = ((rho2 :***: rhoT) :&&&:
            (pi2 :***: r     :***: piT)) :***: r
    riS = ((pi2 :***: pi   :***: piT) :&&&:
            ((pi2 :***: rho :***: piT) :&&&:
            (rho2 :***: rhoT)) :***: r :***: rhoT) :***: r
```

in case o == first && o == secnd of

True -> RF \$ leS :===: riS

False -> error "Assoc needs 3 equal cat0s"

\_ -> error "Domain is not a product"

## Definition of an implication

Given a set  $V$  of **values**, ordered by  $E$ , one will ask for an **implication operator**  $J$  and demand

$$J(\perp, x) = \top$$

$$J(x, \top) = \top$$

$$J(\top, \perp) = \perp$$

$$J(x, y) \geq J(z, y) \quad \text{for all } y \quad \text{whenever } x \leq z$$

$$J(x, y) \leq J(x, t) \quad \text{for all } x \quad \text{whenever } y \leq t$$

Often we study an  **$S$ -implication** characterized by a  $t$ -conorm  $S$  and a strong negation  $N$  as

$$J_{S,N}(x, y) := S(N(x), y)$$

Monotony in implication formulated component-free:

$$J(x, y) \geq J(z, y) \quad \text{for all } y \quad \text{whenever } x \leq z$$

$$E \subseteq \inf_y \pi^\top; (J; E^\top; J^\top \cap \rho; y; \mathbb{T} \cap (\rho; y; \mathbb{T})^\top); \pi$$

```
isImplicationRightMonotoneFormula rtE rtJ =
let o = domRT rtE
prod = DirPro o o
pi = Pi o o; rho = Rho o o
y = VarE "y" o; yTe = PointVect $ EV y
jEJ = rtJ :***: (Convs rtE) :***: (Convs rtJ)
rhoY = (rho :****: yTe) :||--: (UnivV prod)
rf = RFCT (EVar y)
          (Convs pi :***: (jEJ :&&&: rhoY :&&&:
                           (Convs rhoY)) :***: pi)
in RF $ rtE :<==: (InfRela $ RT rf (ET o))
```

An  $R$ -implication associated with a  $t$ -norm  $T$  is defined as  
 $J_T(x, y) := \sup\{z \mid T(x, z) \leq y\}$

How to come from  $(x, y)$  to  $z$ ?

$$(x, y) (\pi; \pi^T_{(x, ?)} \cap \rho_y; E^T_{T(u, z)}; T^T_{(u, z)}) (x, z); \rho_z$$

Therefore we consider

$$(\pi; \pi^T \cap \rho; E^T; T^T); \rho$$

and position it ready for applying **lub** columnwise

$$J_R = [\text{lub}_E \{\rho^T; (\pi; \pi^T \cap T; E; \rho^T)\}]^T$$

# De Morgan model no. 1

$E, S, T, N$  are given

$$E = \begin{matrix} & \text{low} & \text{medium} & \text{high} \\ \text{low} & \begin{pmatrix} 1 & 1 & 1 \end{pmatrix} \\ \text{medium} & \begin{pmatrix} 0 & 1 & 1 \end{pmatrix} \\ \text{high} & \begin{pmatrix} 0 & 0 & 1 \end{pmatrix} \end{matrix}$$

$$N = \begin{matrix} & \text{low} & \text{medium} & \text{high} \\ \text{low} & \begin{pmatrix} 0 & 0 & 1 \end{pmatrix} \\ \text{medium} & \begin{pmatrix} 0 & 1 & 0 \end{pmatrix} \\ \text{high} & \begin{pmatrix} 1 & 0 & 0 \end{pmatrix} \end{matrix}$$

$$T = \begin{matrix} & \text{low} & \text{medium} & \text{high} \\ \text{(low,low)} & \begin{pmatrix} 1 & 0 & 0 \end{pmatrix} \\ \text{(medium,low)} & \begin{pmatrix} 1 & 0 & 0 \end{pmatrix} \\ \text{(low,medium)} & \begin{pmatrix} 1 & 0 & 0 \end{pmatrix} \\ \text{(high,low)} & \begin{pmatrix} 1 & 0 & 0 \end{pmatrix} \\ \text{(medium,medium)} & \begin{pmatrix} 0 & 1 & 0 \end{pmatrix} \\ \text{(low,high)} & \begin{pmatrix} 1 & 0 & 0 \end{pmatrix} \\ \text{(high,medium)} & \begin{pmatrix} 0 & 1 & 0 \end{pmatrix} \\ \text{(medium,high)} & \begin{pmatrix} 0 & 1 & 0 \end{pmatrix} \\ \text{(high,high)} & \begin{pmatrix} 0 & 0 & 1 \end{pmatrix} \end{matrix}$$

$$S = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix}$$

$$J_{S,N} = \begin{matrix} & \text{low} & \text{medium} & \text{high} \\ \text{(low,low)} & \begin{pmatrix} 0 & 0 & 1 \end{pmatrix} \\ \text{(medium,low)} & \begin{pmatrix} 0 & 1 & 0 \end{pmatrix} \\ \text{(low,medium)} & \begin{pmatrix} 0 & 0 & 1 \end{pmatrix} \\ \text{(high,low)} & \begin{pmatrix} 1 & 0 & 0 \end{pmatrix} \\ \text{(medium,medium)} & \begin{pmatrix} 0 & 1 & 0 \end{pmatrix} \\ \text{(low,high)} & \begin{pmatrix} 0 & 0 & 1 \end{pmatrix} \\ \text{(high,medium)} & \begin{pmatrix} 0 & 1 & 0 \end{pmatrix} \\ \text{(medium,high)} & \begin{pmatrix} 0 & 0 & 1 \end{pmatrix} \\ \text{(high,high)} & \begin{pmatrix} 0 & 0 & 1 \end{pmatrix} \end{matrix}$$

$$J_R = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix}$$

# De Morgan model no. 2      $E, S, T, N$ are given

$$E = \begin{pmatrix} & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 1 & \textcolor{red}{1} \\ 2 & 0 & \textcolor{red}{1} & 0 & \textcolor{red}{1} & 0 & \textcolor{red}{1} & 0 & \textcolor{red}{1} \\ 3 & 0 & 0 & \textcolor{red}{1} & \textcolor{red}{1} & 0 & 0 & \textcolor{red}{1} & \textcolor{red}{1} \\ 4 & 0 & 0 & 0 & \textcolor{red}{1} & 0 & 0 & 0 & \textcolor{red}{1} \\ 5 & 0 & 0 & 0 & 0 & \textcolor{red}{1} & \textcolor{red}{1} & \textcolor{red}{1} & \textcolor{red}{1} \\ 6 & 0 & 0 & 0 & 0 & 0 & \textcolor{red}{1} & 0 & \textcolor{red}{1} \\ 7 & 0 & 0 & 0 & 0 & 0 & 0 & \textcolor{red}{1} & \textcolor{red}{1} \\ 8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \textcolor{red}{1} \end{pmatrix}$$

$$N = \begin{pmatrix} & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \textcolor{red}{1} \\ 2 & 0 & 0 & 0 & 0 & 0 & 0 & \textcolor{red}{1} & 0 \\ 3 & 0 & 0 & 0 & 0 & 0 & \textcolor{red}{1} & 0 & 0 \\ 4 & 0 & 0 & 0 & 0 & \textcolor{red}{1} & 0 & 0 & 0 \\ 5 & 0 & 0 & 0 & \textcolor{red}{1} & 0 & 0 & 0 & 0 \\ 6 & 0 & 0 & \textcolor{red}{1} & 0 & 0 & 0 & 0 & 0 \\ 7 & 0 & \textcolor{red}{1} & 0 & 0 & 0 & 0 & 0 & 0 \\ 8 & \textcolor{red}{1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

$$T = [[1,1,1,1,1,1,1,1], [1,2,1,2,1,2,1,2], [1,1,3,3,1,1,3,3], [1,2,3,4,1,2,3,4], [1,1,1,1,5,5,5,5], [1,2,1,2,5,6,5,6], [1,1,3,3,5,5,7,7], [1,2,3,4,5,6,7,8]]$$

$$S = [[1,2,3,4,5,6,7,8], [2,2,4,4,6,6,8,8], [3,4,3,4,7,8,7,8], [4,4,4,4,8,8,8,8], [5,6,7,8,5,6,7,8], [6,6,8,8,6,6,8,8], [7,8,7,8,7,8,7,8], [8,8,8,8,8,8,8,8]]$$

$$S_{J,N} = [[8,8,8,8,8,8,8,8], [7,8,7,8,7,8,7,8], [6,6,8,8,6,6,8,8], [5,6,7,8,5,6,7,8], [4,4,4,4,8,8,8,8], [3,4,3,4,7,8,7,8], [2,2,4,4,6,6,8,8], [1,2,3,4,5,6,7,8]]$$

$$R_T = [[8,8,8,8,8,8,8,8], [7,8,7,8,7,8,7,8], [6,6,8,8,6,6,8,8], [5,6,7,8,5,6,7,8], [4,4,4,4,8,8,8,8], [3,4,3,4,7,8,7,8], [2,2,4,4,6,6,8,8], [1,2,3,4,5,6,7,8]]$$

# Outline

1. Motivation for a relational language
2. Aims
3. Architecture
4. Substrate on which to build
5. Language details
6. Case study: Measure and Order
7. Case study: De Morgan Triples
- 8. Case study: Relational Integration**

# Example of relational integration

$$\mu = \begin{array}{ll} \{\} & \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \\ \{\text{color}\} \\ \{\text{price}\} \\ \{\text{color,price}\} \\ \{\text{speed}\} \\ \{\text{color,speed}\} \\ \{\text{price,speed}\} \\ \{\text{color,price,speed}\} \end{array}$$

$$\begin{array}{ll} \{\text{color}\} & \begin{pmatrix} 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{pmatrix} \\ \{\text{price}\} \\ \{\text{speed}\} \end{array}$$

$$4 = \text{lub} [\text{glb}(2_{v(\text{speed})}, 5_{\mu\{s,c,p\}}), \\ \text{glb}(4_{v(\text{color})}, 4_{\mu\{c,p\}}), \\ \text{glb}(4_{v(\text{price})}, 4_{\mu\{c,p\}})]$$

$$\begin{array}{ll} \{\text{color}\} & \begin{pmatrix} 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix} \\ \{\text{price}\} \\ \{\text{speed}\} \end{array}$$

$$3 = \text{lub} [\text{glb}(2_{v(\text{price})}, 5_{\mu\{p,c,s\}}), \\ \text{glb}(4_{v(\text{color})}, 3_{\mu\{c,s\}}), \\ \text{glb}(4_{v(\text{speed})}, 3_{\mu\{c,s\}})]$$

**Definition.** Assume a set of criteria  $\mathcal{C}$  and a (finite) lattice  $\mathcal{L}$ , ordered by  $E$ , in which subsets of these criteria shall be measured. Let  $\Omega$  be the ordering on  $\mathcal{P}(\mathcal{C})$ .

A mapping  $\mu : \mathcal{P}(\mathcal{C}) \longrightarrow \mathcal{L}$  is a **relational measure** if

- $\Omega; \mu \subseteq \mu; E$ , meaning that  $\mu$  is isotonic wrt. to the orderings  $\Omega$  and  $E$ .
- $\mu^\top; 0_\Omega = 0_E$ , meaning that the empty subset of  $\mathcal{P}(\mathcal{C})$  is mapped to the least element of  $\mathcal{L}$ .
- $\mu^\top; 1_\Omega = 1_E$ , meaning that the full subset of  $\mathcal{P}(\mathcal{C})$  is mapped to the greatest element of  $\mathcal{L}$ .

The **Sugeno integral** operator corresponds to

$$M_{S,\mu}(x_1, \dots, x_m) = (S) \int x \circ \mu = \bigvee_{i=1}^m [x_i \wedge \mu(A_i)]$$

The **Choquet integral** operator corresponds to

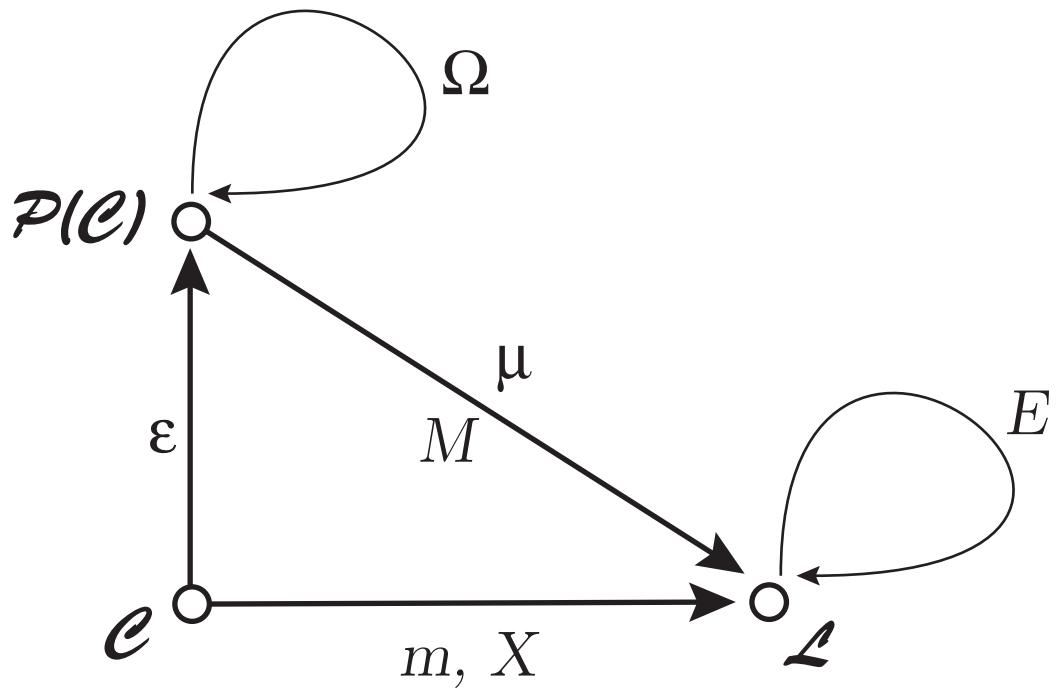
$$M_{C,\mu}(x_1, \dots, x_m) = (C) \int x \circ \mu = \sum_{i=1}^m [(x_i - x_{i-1}) \cdot \mu(A_i)]$$

In both cases the elements of vector  $\underline{x} : (x_1, \dots, x_m)$  have been reordered such that

$$0 = x_0 \leq x_1 \leq x_2 \leq \dots \leq x_m \leq x_{m+1} = 1$$

and  $\mu(A_i) = \mu(C_i, \dots, C_m)$ .

# Basic situation for relational integration



$$(R)\int X \circ \mu := \text{lubR}_E(\mathbb{T}; \text{glbR}_E[X \cup \text{syq}(X; E^\top; X^\top, \varepsilon); \mu])$$

```

isMeasure e mu =
  let p@(DirPow d) = domRT mu
      omega = powerSetOrdering d
      rf1 = RF $ omega :***: mu :<=: (mu :***: e)
      rf2 = VF $ Convs mu :****: (leaEleFct omega)
                  :=====: (leaEleFct e)
      rf3 = VF $ Convs mu :****: (greEleFct omega)
                  :=====: (greEleFct e)
  in Conjunct rf1 (Conjunct rf2 rf3)

```

```

relationIntegral e mu x =
  let xExT      = x :***: e :***: (Convs x)
      syq       = SyQ xExT (Epsi $ domRT x)
      syqMu    = syq :***: mu
      x0Rsyq   = x :|||: syqXMu
      glbfx0Rs = glbRelaFunctionR e x0Rsyq
      allGlb   = UnivR Unit0b (domRT x) :***: glbfx0Rs
  in lubRelaFunctionR e allGlb

```

**TITUREL** ontvangt de Heilige Graal en de Heilige Speer uit handen van een Engelenschaar die neder daalt uit de hemel. Hij bouwt een Tempel voor deze heilige relikwiën, de Graalburcht Montsalvat. Ridders die tot de Graal worden geroepen vormen de ridderschap van de Heilige Graal, hun Koning is Titurel. Op hoge leeftijd draagt hij zijn ambt over op zijn zoon Amfortas.



The system **TITUR****E****L** runs under one of the following acronym interpretations

- This is the ultimate relation system
- Towards improved techniques using relations
- Teaching informaticians to use relations
- Try it, to use relations
- Toolkit intended to use relations
- Testing innovative tools using relations
- Think innovative – try using relations