Grailog:
Graph inscribed logic

Harold Boley
NRC-IIT Fredericton
Faculty of Computer Science
University of New Brunswick
Canada

12 November 2008
(Revised: 14 Dec. 2008)

Generalized Graphs for the Presentation and Mapping of Logic Languages

- We have used generalized graphs for presenting various logic languages, where basically:
  - Graph nodes (vertices) present individuals, classes, etc.
  - Graph arcs (edges) present relations
- What are the principles of this presentation and what graph generalizations are required?
- How can these graphs be mapped to logic?

Grailog Principles

- Graphs should be able to present all logic constructs, incl. description logic TBoxes
- Graphs should make it easier for humans to read and write logic constructs by exploiting the 2-dimensional presentation
- Graphs should be natural extensions of Directed Labeled Graphs (DLGs) for the presentation of atomic ground formulas in function-free binary predicate logic

Grailog Generalizations

- **Recursive (hierarchical) graphs:** For nested terms and formulas, modal logics, and modularization, ‘flat’ graphs should be generalized to allow other graphs as *complex nodes* to any level of ‘depth’

- **Directed hypergraphs:** For n-ary relations, directed (binary) arcs should be generalized to directed (n-ary) *hyperarcs*

- **LabelNode graphs:** For allowing hybrid logics describing both instances and predicates, arc *labels* should also become usable as *nodes*

Based on: Directed Recursive Labelnode Hypergraphs (DRLHs)

Graphical Elements: Basic Shapes (1)

- Boxes for atomic and complex nodes
  - Rectangle: Atomic for instances. Complex for instance-denoting function applications
  - Oval: Classes
  - Octagon: Embedded propositions
- Labeled arrows (directed links) for arcs and hyperarcs (where hyperarcs ‘cut through’ nodes intermediate between first and last)

Graphical Elements: Basic Shapes (2)

- Unlabeled wavy/heavy links for special arcs
  - without arrow heads: instanceOf/subClassOf in bottom-up ‘flow’ direction
  - with arrow heads on one end: instanceOf/subClassOf in indicated arrow direction
  - with arrow heads on both ends (heavy links): equal in symmetric manner

Graphical Elements: Line Styles

- Solid Lines (boxes & links): Positive
- Dashed Lines (boxes & links): Negative
- Dotted Lines (boxes): Disjunctive
- Wavy Lines (links): instanceOf
- Heavy Lines (links): subClassOf

Graphical Elements: Hatching Patterns

- No hatching (boxes): Constant
- Hatching (boxes): Variable

Instances: Individual Constants

	Graph (node)	mapping	Logic
General:	*instance*	⟶	*instance*
Examples:	Graph (node)		Logic
	Warren Buffett		Warren Buffett
	General Electric		General Electric
	US$ 3 000 000 000		US$ 3 000 000 000

Unknowns: Individual Variables

General: Graph (hatched node) Logic

variable

Examples: Graph (node) Logic

X

Y

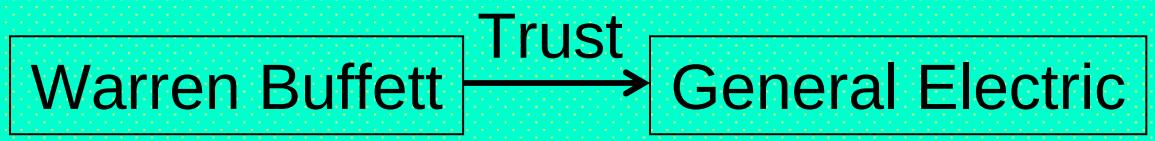
A

Predicates: Binary Relations (1)

General: Graph (labeled arc) Logic

relation($inst_1$, $inst_2$)

Example: Graph (labeled arc) Logic



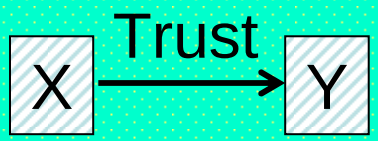
Trust(Warren Buffett, General Electric)

Predicates: Binary Relations (2)

General: Graph (labeled arc) Logic

relation(var$_1$, var$_2$)

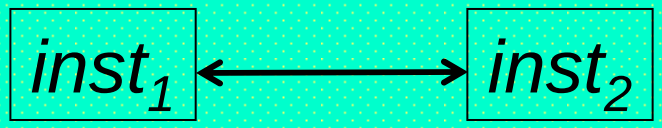
Example: Graph (labeled arc) Logic



Trust(X,Y)

Equality Predicate: Symmetric

General: Graph (unlabeled undirected arc) Logic (with equality)



$inst_1 = inst_2$

Example: Graph (unlabeled undirected arc) Logic (with equality)

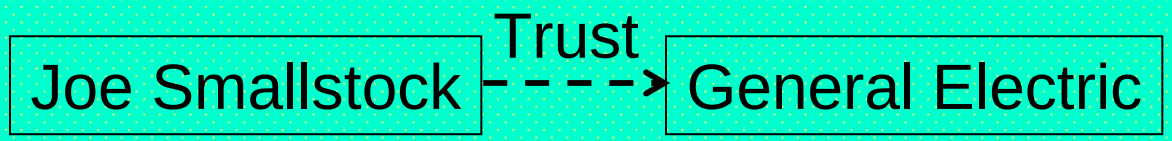
GE = General Electric

Negated Predicates: Binary Relations

General: Graph (dashed arc) Logic

$\neg\, relation(inst_1, inst_2)$

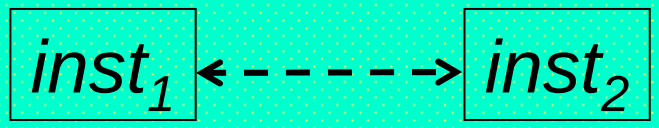
Example: Graph (dashed arc) Logic



$\neg$ Trust(
Joe Smallstock,
General Electric
)

Inequality Predicate: Symmetric

General: Graph (dashed undirected arc)



Logic (with equality)

$inst_1 \neq inst_2$

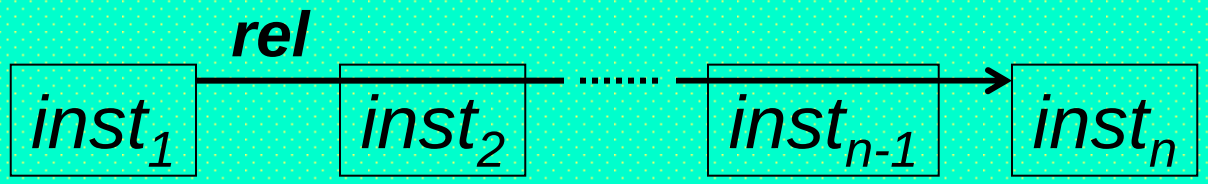
Example: Graph (dashed undirected arc)

Logic (with equality)

Joe Smallstock ≠ Warren Buffett

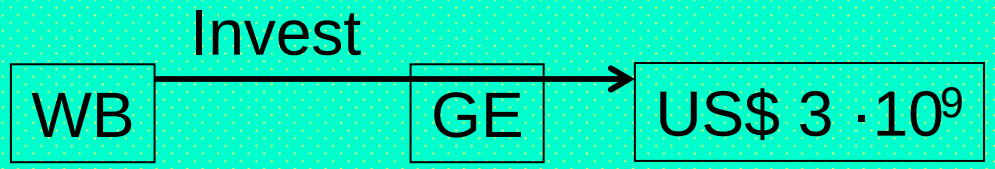
Predicates: n-ary Relations (n≥2)

General: Graph (hyperarc) Logic



$rel(inst_1, inst_2, ..., inst_{n-1}, inst_n)$

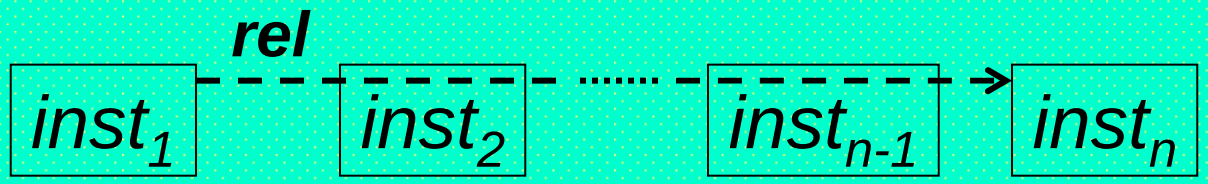
Example: Graph (hyperarc) Logic
(n=3)



Invest(WB, GE, US$ 3·10^9)

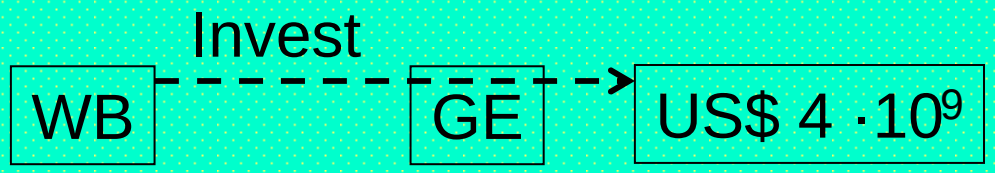
Negated Predicates: n-ary Relations

General: Graph (dashed: not) Logic



$\neg rel(inst_1, inst_2, ..., inst_{n-1}, inst_n)$

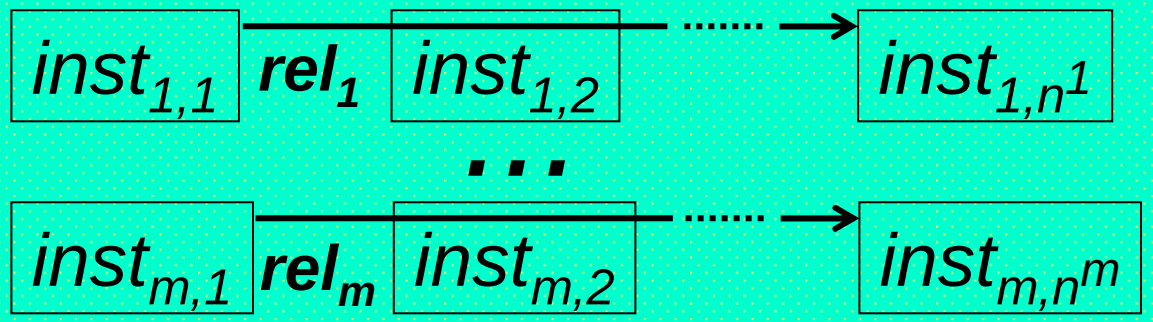
Example: Graph (dashed: not) Logic
(n=3)



$\neg$ Invest(WB, GE, US$ $4 \cdot 10^9$)

Implicit Conjunction of Formula Graphs: Co-Occurrence on Top-Level

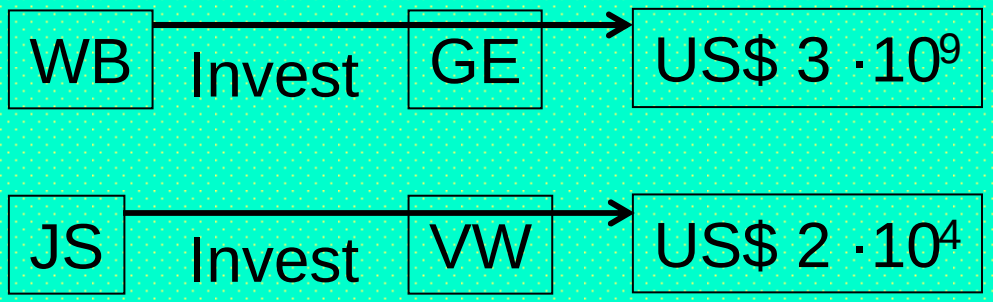
General: Graph (m hyperarcs)



Logic

$$rel_1(inst_{1,1}, inst_{1,2}, \ldots, inst_{1,n^1}) \wedge \ldots \wedge rel_m(inst_{m,1}, inst_{m,2}, \ldots, inst_{m,n^m})$$

Example: Graph (2 hyperarcs)

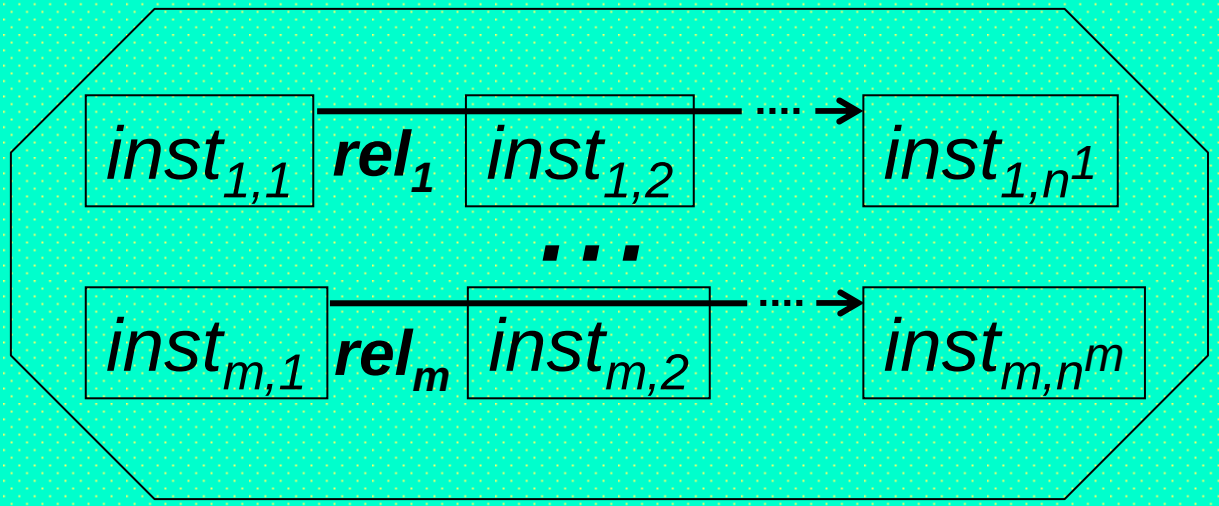


Logic

$$Invest(WB, GE, US\$\ 3 \cdot 10^9) \wedge Invest(JS, VW, US\$\ 2 \cdot 10^4)$$

Explicit Conjunction of Formula Graphs: Co-Occurrence in Complex Node

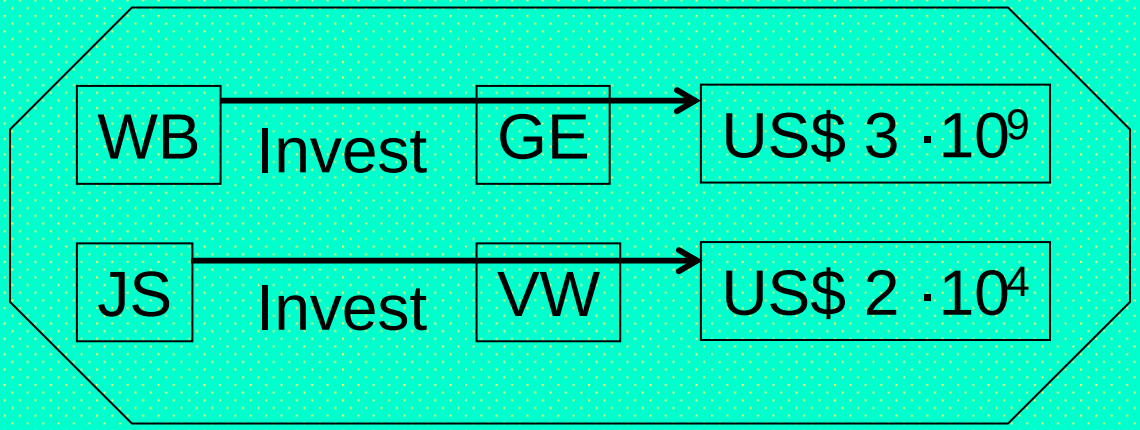
General: Graph (m hyperarcs)



Logic

$$rel_1(inst_{1,1}, inst_{1,2}, \ldots, inst_{1,n^1}) \wedge \ldots \wedge rel_m(inst_{m,1}, inst_{m,2}, \ldots, inst_{m,n^m})$$

Example: Graph (2 hyperarcs)

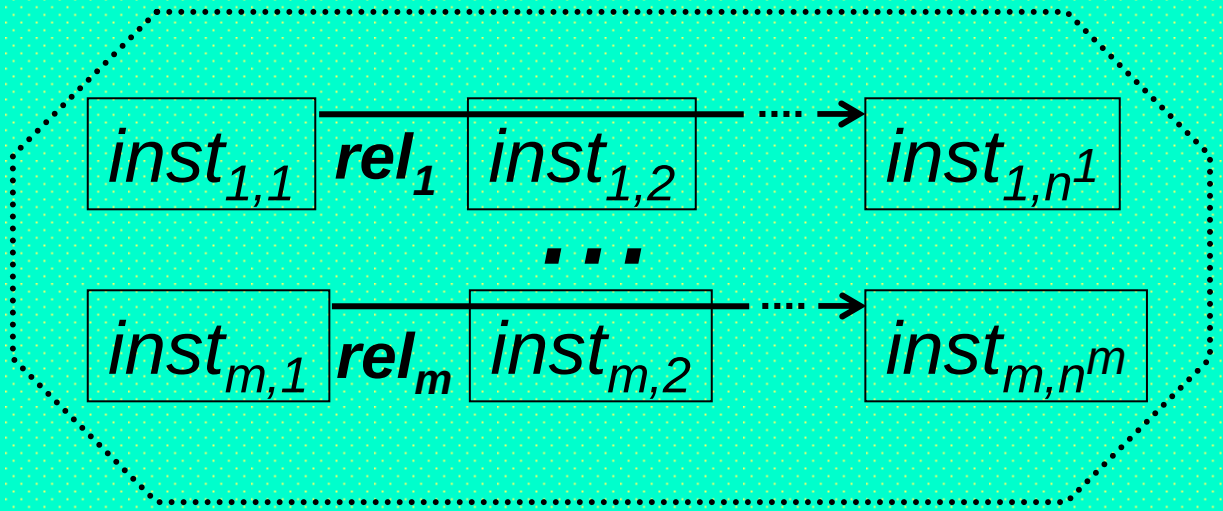


Logic

$$\text{Invest(WB, GE, US\$ } 3 \cdot 10^9) \wedge \text{Invest(JS, VW, US\$ } 2 \cdot 10^4)$$

Disjunction of Formula Graphs: Co-Occurrence in Disjunctive Node

General: Dotted Graph



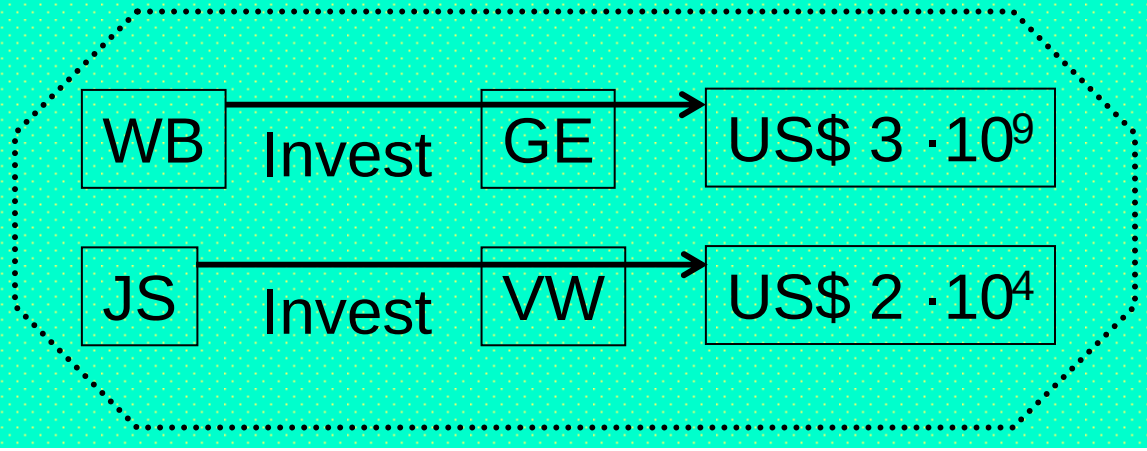
Logic

$rel_1(inst_{1,1}, inst_{1,2}, \ldots, inst_{1,n1})$ $\vee$

$\ldots$ $\vee$

$rel_m(inst_{m,1}, inst_{m,2}, \ldots, inst_{m,nm})$

Example: Dotted Graph



Logic

Invest(WB, GE, US$ $3 \cdot 10^9$) $\vee$

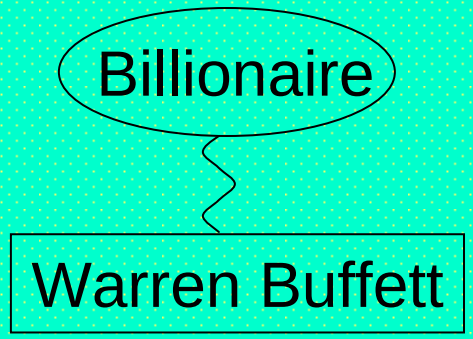
Invest(JS, VW, US$ $2 \cdot 10^4$)

Predicates: Unary Relations (Classes, Concepts, Types)

General: Graph (instance node classified with wavy arc) Logic

$class(inst_1)$

Example: Graph (classified node) Logic



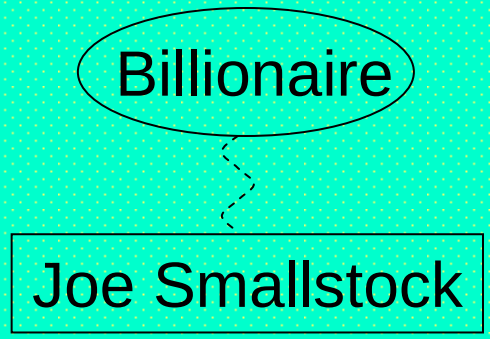
Billionaire(
Warren Buffett)

Negated Predicates: Unary Relations

General: Graph (node classified with dashed wavy arc) Logic

$\neg\ class(inst_1)$

Example: Graph (classified node) Logic



¬ Billionaire(
Joe Smallstock)

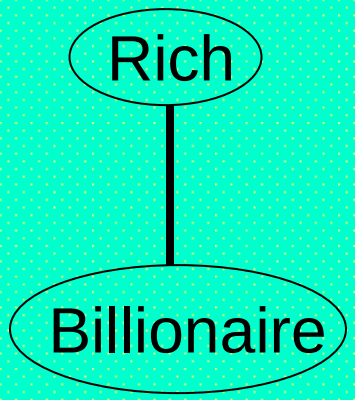
Class Hierarchies (Taxonomies): Subclass Relation

General: Graph (two nodes)

(Description) Logic

$class_1 \sqsubseteq class_2$

Example: Graph (two nodes)

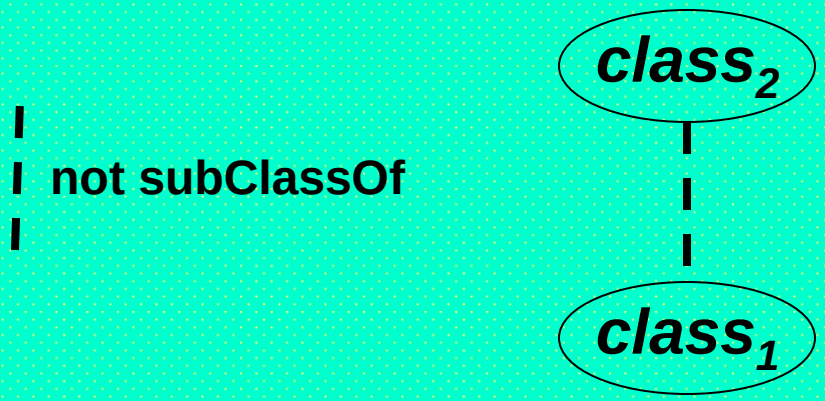


(Description) Logic

$\text{Billionaire} \sqsubseteq \text{Rich}$

Class Hierarchies (Taxonomies): Negated Subclass Relation

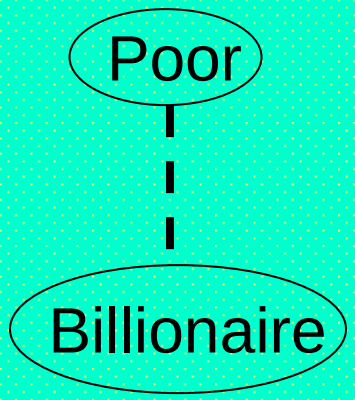
General: Graph (two nodes)



(Description) Logic

$class_1 \not\equiv class_2$

Example: Graph (two nodes)

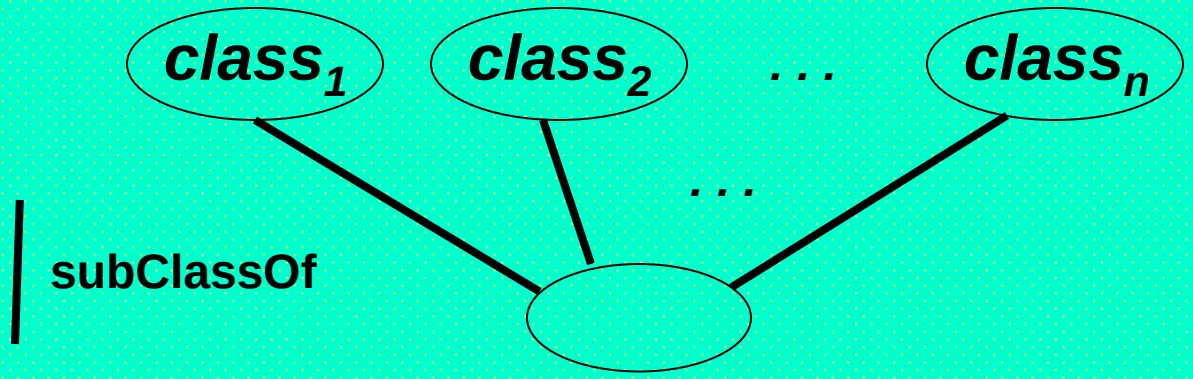


(Description) Logic

Billionaire $\not\equiv$ Poor

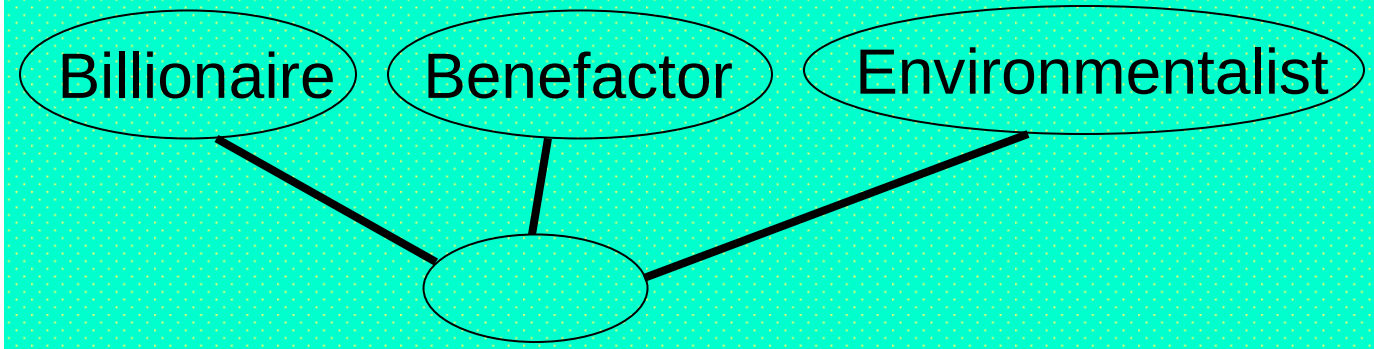
Class Hierarchies (Taxonomies): Class Intersection

General: Graph (blank node under n) (Description) Logic



$class_1 \sqcap$
$class_2 \sqcap$
$\ldots \sqcap$
$class_n$

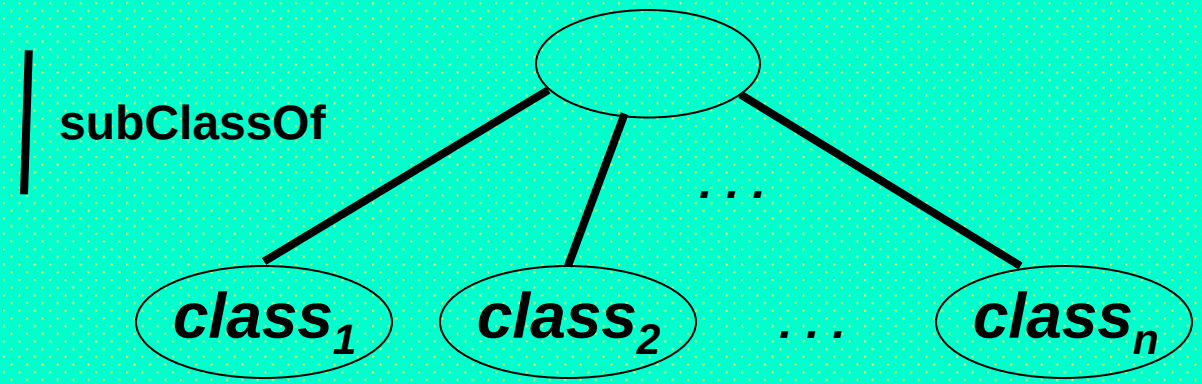
Example: Graph (blank node under 3) (Description) Logic



Billionaire ⊓
Benefactor ⊓
Environmentalist

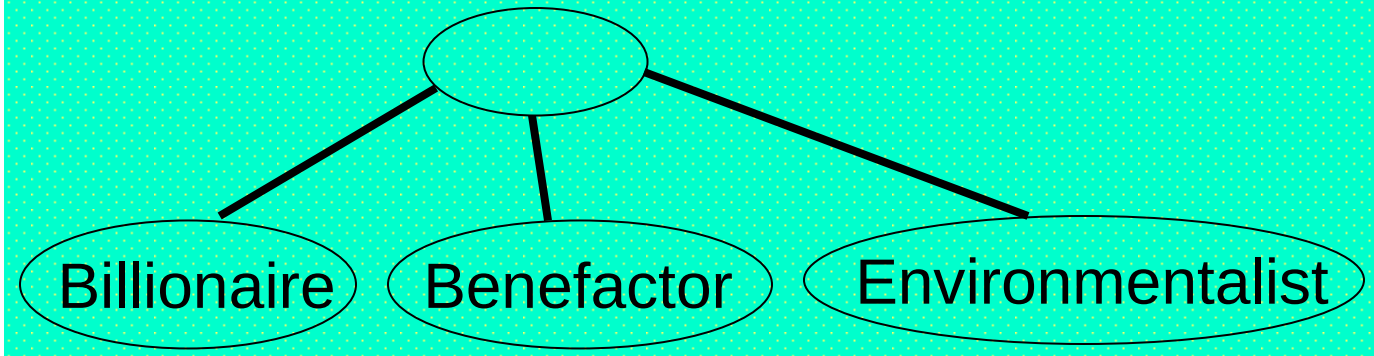
Class Hierarchies (Taxonomies): Class Union

General: Graph (blank node over n) (Description) Logic



$class_1 \sqcup class_2 \sqcup \ldots \sqcup class_n$

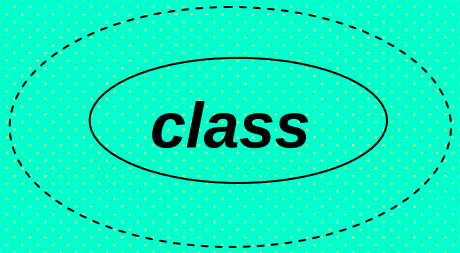
Example: Graph (blank node over 3) (Description) Logic



Billionaire $\sqcup$ Benefactor $\sqcup$ Environmentalist

Class Hierarchies (Taxonomies): Class Complement

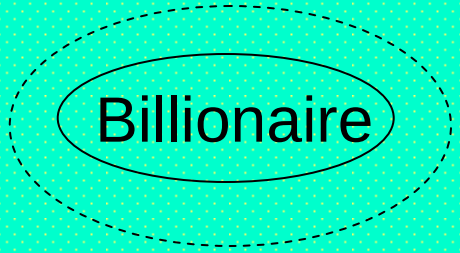
General: Graph (*dashed node* contains node to be complemented)



(Description) Logic

$\neg$ *class*

Example: Graph



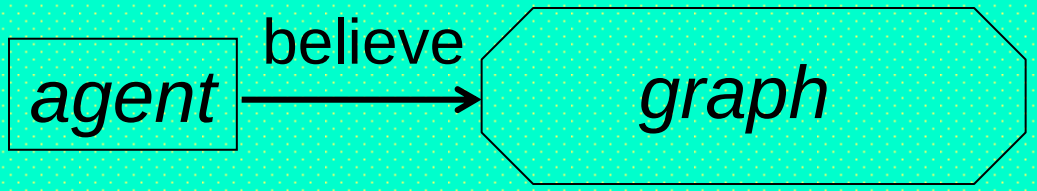
(Description) Logic

$\neg$ Billionaire

Modally Embedded Propositions

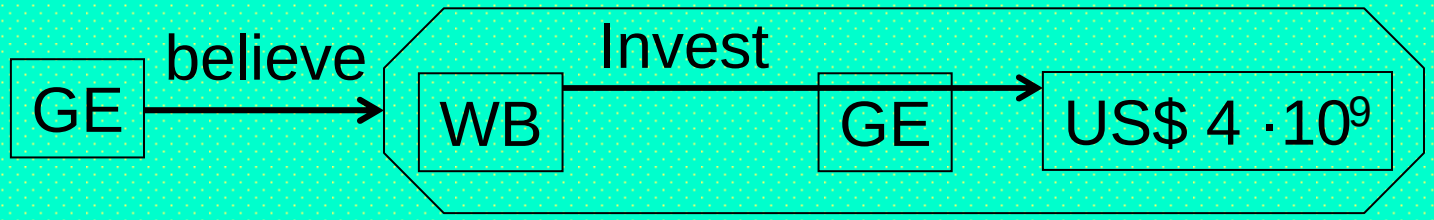
General: Graph (Modal) Logic

(complex *octagon node* used to 'quarantine' what another agent believes etc.)



$believe_{agent}(graph)$

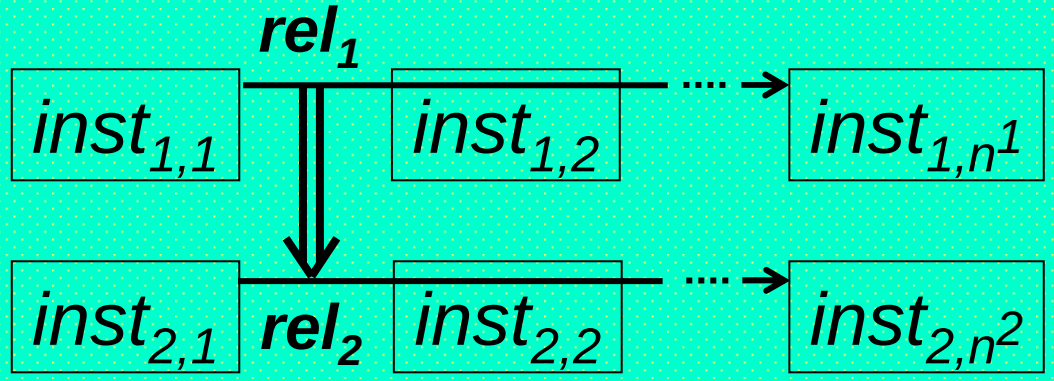
Example: Graph (Modal) Logic



$believe_{GE}(Invest(WB, GE, US\$\ 4 \cdot 10^9))$

Rules: Relations Imply Relations (1)

General: Graph (ground)

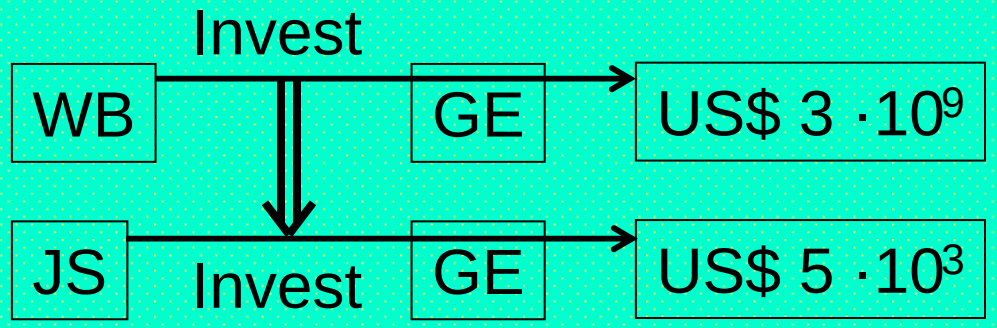


Logic

$rel_1(inst_{1,1}, inst_{1,2}, ..., inst_{1,n1}) \Rightarrow$

$rel_2(inst_{2,1}, inst_{2,2}, ..., inst_{2,n2})$

Example: Graph (ground)



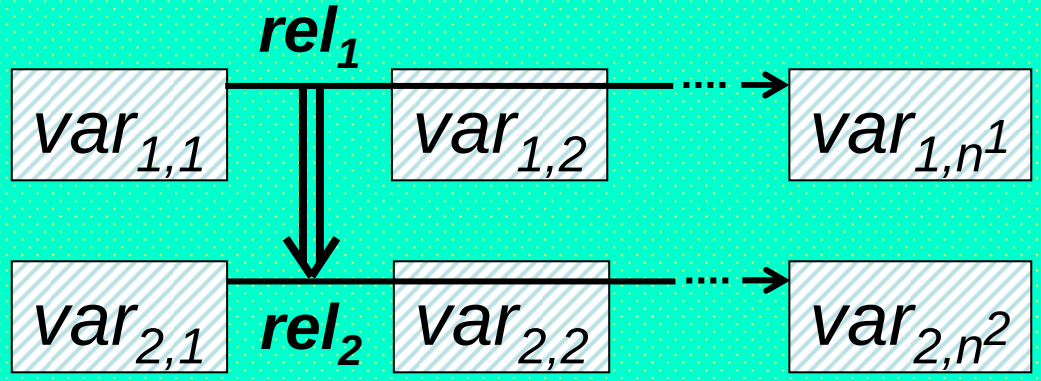
Logic

Invest(WB, GE, US$ $3 \cdot 10^9$) $\Rightarrow$

Invest(JS, GE, US$ $5 \cdot 10^3$)

Rules: Relations Imply Relations (2)

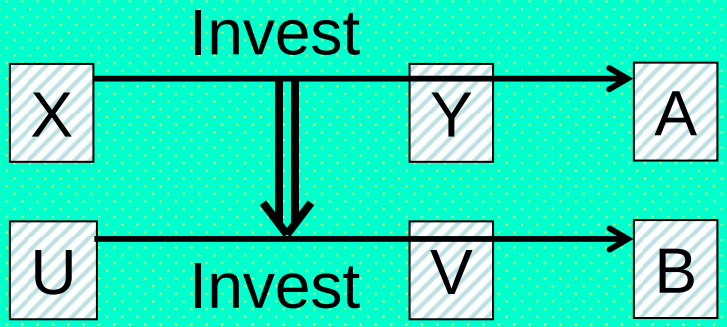
General: Graph (non-ground)



Logic

$(\forall var_{i,j})$
$rel_1(var_{1,1}, var_{1,2}, \ldots, var_{1,n1}) \Rightarrow$
$rel_2(var_{2,1}, var_{2,2}, \ldots, var_{2,n2})$

Example: Graph (non-ground)

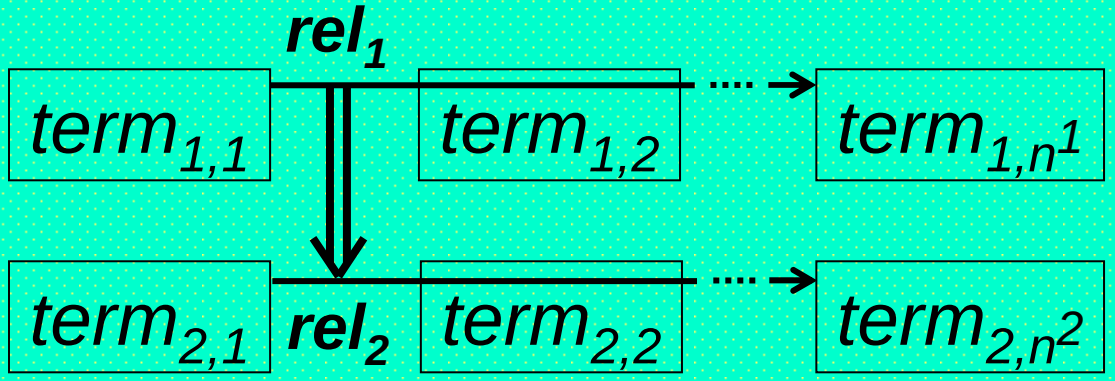


Logic

$(\forall$ X, Y, A, U, V, B)
Invest(X, Y, A) $\Rightarrow$
Invest(U, V, B)

Rules: Relations Imply Relations (3)

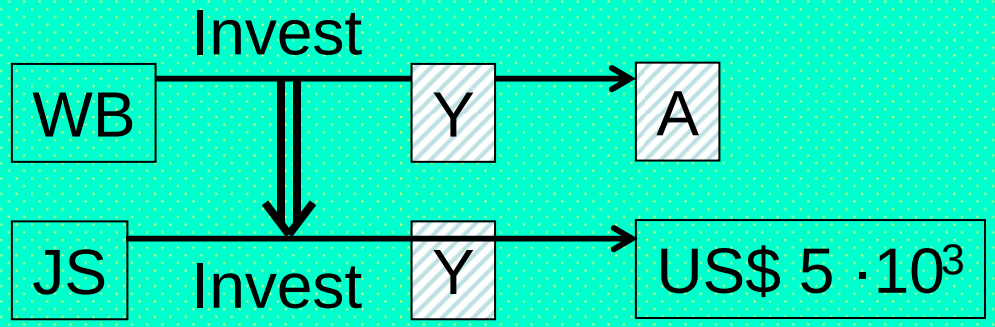
General: Graph (inst/var terms)



Logic

$(\forall var_{i,j})$
$rel_1(term_{1,1}, term_{1,2}, \ldots, term_{1,n1}) \Rightarrow$
$rel_2(term_{2,1}, term_{2,2}, \ldots, term_{2,n2})$

Example: Graph (inst/var terms)

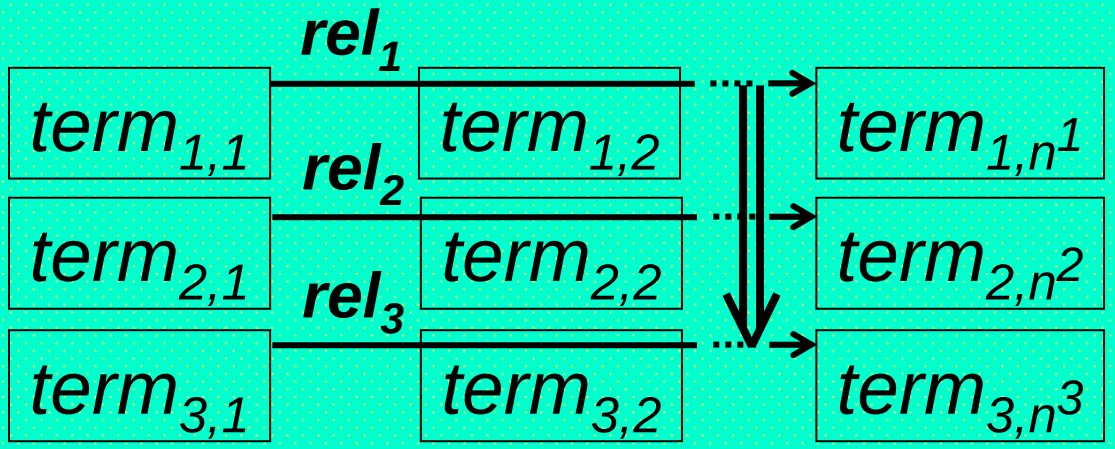


Logic

$(\forall\ Y, A)$
$Invest(WB, Y, A) \Rightarrow$
$Invest(JS, Y, US\$\ 5 \cdot 10^3)$

Rules: Conjunctions Imply Relations

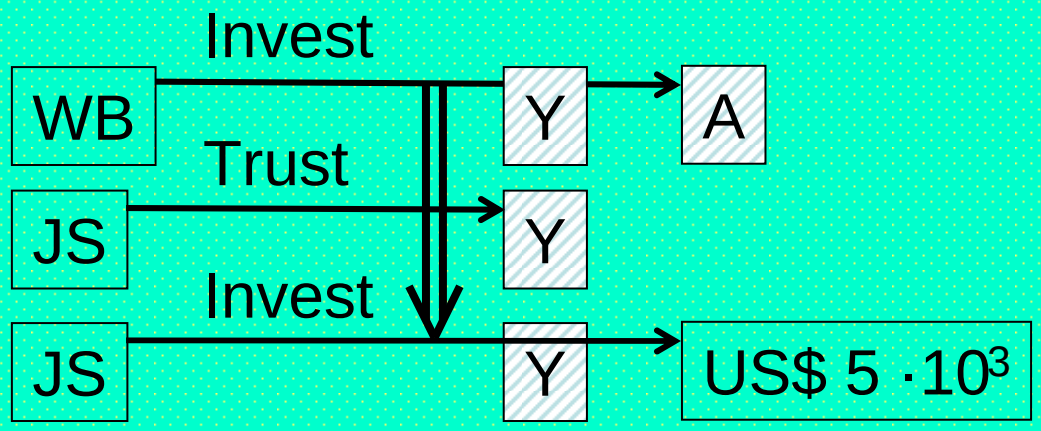
General: Graph (inst/var terms)



Logic

$$(\forall var_{i,j})\ rel_1(term_{1,1}, term_{1,2}, \ldots, term_{1,n1}) \wedge rel_2(term_{2,1}, term_{2,2}, \ldots, term_{2,n2}) \Rightarrow rel_3(term_{3,1}, term_{3,2}, \ldots, term_{3,n3})$$

Example: Graph (inst/var terms)



Logic

$$(\forall\ Y, A)\ Invest(WB, Y, A) \wedge Trust(JS, Y) \Rightarrow Invest(JS, Y, US\$\ 5 \cdot 10^3)$$

Optional Readings

Focus — Directed Recursive Labelnode Hypergraphs (DRLHs) :

Boley, Harold. Declarative Operations on Nets.
http://www.ict.tuwien.ac.at/lva/Boley_LFCS/drlhops.pdf

Extra — Formal Logic and Logic on the Web:

Enderton, Herbert. A Mathematical Introduction to Logic, Second Edition, Academic Press, 2001.
http://www.math.ucla.edu/~hbe/amil/index.html

ISO/IEC. Information technology — Common Logic (CL): a framework for a family of logic-based languages. Reference number ISO/IEC 24707:2007(E). First edition 2007-10-01.
http://cl.tamu.edu

Boley, Harold and Kifer, Michael. RIF Basic Logic Dialect.
http://www.w3.org/TR/rif-bld