

Logic as The Calculus of Computer Science

noneng. S. Baselice (*I have no stamp*)¹

Ottobre, 2007

¹Università di Napoli Federico II

What is a Logic?

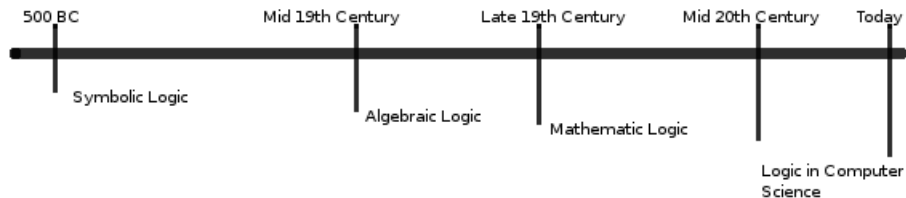
A *Logic* is a formalism with

- a syntax
- a semantics
- an inference mechanism for reasoning

$$((a \vee b) \wedge \neg a) \rightarrow b$$

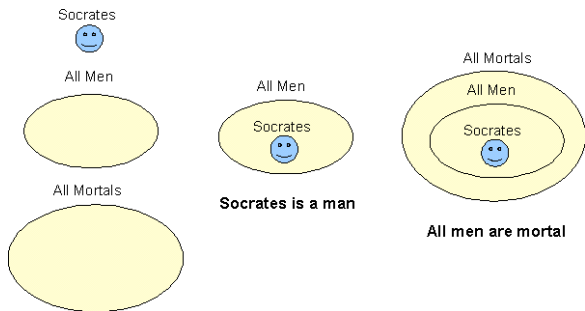
a	b	$a \rightarrow b$
0	0	1
0	1	1
1	0	0
1	1	1

Historical Diagram



The First Age of Logic: Symbolic Logic (500 B.C. - 19th Century)

- Originally, Logic dealt with arguments in the natural languages used by humans.



The First Age of Logic: Symbolic Logic (500 B.C. - 19th Century)

The Surprise Paradox:



- Natural language is too ambiguous!
- It is needed a *Symbolic Language*.

The 2nd Age of Logic: Algebraic Logic (Mid to Late 19th Century)



In 1847, George Boole attempted to formulate logic in terms of mathematical language.

Boolean algebra

AND
 $f = x \cdot y = xy$

x	y	F
0	0	0
0	1	0
1	0	0
1	1	1

OR
 $f = x + y$

x	y	F
0	0	0
0	1	1
1	0	1
1	1	1

"inclusive OR"

NOT
 $f = x'$

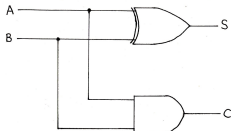
x	y	F
0	0	1
0	1	1
1	0	0
1	1	0

x'
 y is irrelevant

A and B are the inputs
 S represents the output sum and
 C represents the output carry.

The relevant truth table for this circuit is:

A	B	C	S
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0



The 3rd Age of Logic: Mathematical Logic (Late 19th Century to Mid 20th Century)

- As mathematical proofs became more sophisticated, *paradoxes* began to show up in them just as they did in natural language.
- Hilbert proposed logic for computing each mathematical problem.



- Gödel's Incompleteness Theorems proved that logic cannot prove every statement and it cannot prove its own consistency.



The 4th Age of Logic: Logic in Computer Science

- "Logic occupies a central place in computer science so that logic has been called *the calculus of computer science*. ... Computer science started as Logic." (*M. Y. Vardi*)

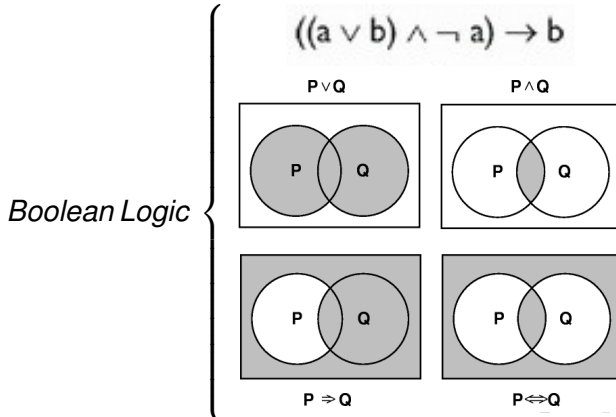


- "I expected that digital computing machines will eventually stimulate a considerable interest in symbolic logic. ... The language in which one communicates with these machines ... forms a sort of symbolic logic." (*A. Turing*)



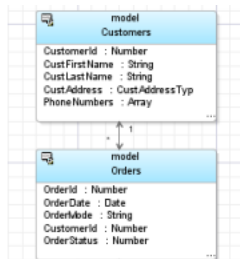
Propositional Logic

- A formalism for modeling statements that are either *true* or *false*.



First Order Logic

- A formalism for specifying properties of mathematical structures, such as *graphs*, *Database System*,...



customer(*Id*, *Name*)

order(*Id*, *Date*)

cust - ord(*C*, *O*)

Foreign Key :

(*axiom*) $\forall \vec{X}. \text{order}(\vec{X}) \exists ! \vec{Y}. (\text{customer}(\vec{Y}) \wedge \text{cust} - \text{ord}(\vec{Y}, \vec{X}))$

query : **SELECT * FROM order**

order(\vec{X})

Modal Logic

- A formalism for modeling *necessity* and *possibility*.

Model Checking: $\psi \equiv$ "there is a deadlock"

$M\psi$, is it possible there will be a deadlock?



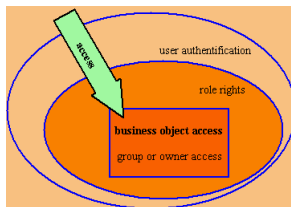
$L\neg\psi$, it is necessary there is no deadlock

Nonmonotonic Logics

A *Nonmonotonic Logic* \mathcal{L} is s.t. adding a new axiom to a theory \mathcal{T} on \mathcal{L} may cause the lost of a theorem proved in \mathcal{T} .

- Access Control Policies:

$$P = \{ \dots, \text{aut}(S_{\text{ogg}}, \text{Act}, O_{\text{gg}}, -) \leftarrow \text{not } \text{aut}(S_{\text{ogg}}, \text{Act}, O_{\text{gg}}, +), \dots \}$$



$$P \models \text{aut}(\text{john}, \text{read}, \text{file}, -)$$



$$P \cup \{ \text{aut}(\text{john}, \text{read}, \text{file}, +) \} \not\models \text{aut}(\text{john}, \text{read}, \text{file}, -)$$

Nonmonotonic Logics

A *Nonmonotonic Logic* \mathcal{L} is s.t. adding a new axiom to a theory \mathcal{T} on \mathcal{L} may cause the lost of a theorem proved in \mathcal{T} .

- Commonsense Knowledge and Reasoning:

"My car is in garage."



..oops..!

"My car is not in garage"

Impact of Logic in Computer Science

- Computer-Aided Verification (model checking)
- Database System (SQL, relational algebra)
- Computational Complexity (complexity and expressivity)
- Programming Languages (formal semantics, type theory, rewriting systems,...)
- Computer Security (access control policies, negotiation policies,...)
- Artificial Intelligence (commonsense knowledge and reasoning)
- Distributed Systems (domain description)
- Logic Programming (inference)
- Software Engineering (modeling)
-

References I



C. Baral.

Knowledge Representation, Reasoning and Declarative Problem Solving.
Cambridge University Press, Cambridge, 2003.



L. C. Aiello and F. Massacci.

Verifying security protocols as planning in logic programming.
ACM Trans. Comput. Logic, 2(4):542–580, 2001.



K. L. Clark.

Negation as failure.

In *Logic and Data Bases*, pages 293–322, 1977.



T. Eiter, W. Faber, and M. Truszczynski, editors.

Logic Programming and Nonmonotonic Reasoning, 6th International Conference, LPNMR 2001, Vienna, Austria, September 17-19, 2001, Proceedings, volume 2173 of *Lecture Notes in Computer Science*. Springer, 2001.

References II



M.H. Van Emden and R.A. Kowalski.

The semantics of predicate logic as a programming language.
J. ACM, 23(4):733–742, 1976.



M. Gabbrielli and G. Gupta, editors.

Logic Programming, 21st International Conference, ICLP 2005, Sitges, Spain, October 2-5, 2005, Proceedings, volume 3668 of *Lecture Notes in Computer Science*. Springer, 2005.



M. Gelfond and V. Lifschitz.

The stable model semantics for logic programming.
In *Proc. of the 5th ICLP*, pages 1070–1080. MIT Press, 1988.



M. Gelfond and V. Lifschitz.

Classical negation in logic programs and disjunctive databases.
New Generation Computing, 9(3-4):365–386, 1991.

References III



M. L. Ginsberg and D. E. Smith.
Possible worlds and the qualification problem.
In *AAAI*, pages 212–217, 1987.



J. W. Lloyd.
Foundations of Logic Programming, 1st Edition.
Springer, 1984.



W. Marek and M. Truszczyński.
Autoepistemic logic.
J. ACM, 38(3):587–618, 1991.



J. McCarthy.
Programs with common sense.
In *Proceedings of the Teddington Conference on the Mechanization of Thought Processes*, pages 75–91, London, 1959. Her Majesty's Stationary Office.

References IV



R. Reiter.

A logic for default reasoning.

Artificial Intelligence, 13:81–132, 1980.



J. R. Shoenfield.

Mathematical Logic.

Series in Logic. Addison-Wesley, Reading, MA, 1967.



P. Simons, I. Niemelä, and T. Soininen.

Extending and implementing the stable model semantics.

Artif. Intell., 138(1-2):181–234, 2002.



P. J. Stuckey, editor.

Logic Programming, 18th International Conference, ICLP 2002, Copenhagen, Denmark, July 29 - August 1, 2002, Proceedings, volume 2401 of *Lecture Notes in Computer Science*. Springer, 2002.

References V



M.Y. Vardi.

Logic as the calculus of computer science.

Technical report, Rice University.



M.Y. Vardi.

A brief history of logic.

Technical report, Rice University, 2003.



M. De Vos and A. Provetti, editors.

Answer Set Programming, Advances in Theory and Implementation, Proceedings of the 3rd Intl. ASP'05 Workshop, Bath, UK, September 27-29, 2005, volume 142 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2005.