Glossary

Little Algebras

□	Operator symbol for "necessity" see *modal logic, necessary, possibility*
◇	Operator symbol for "it is possible that" see also *modal logic, necessary, possibility*
abstract algebra$_1$	A branch of meta-mathematics
abstract algebra$_2$	A set (the *domain of discourse*) with operations (the *signature)* on that set. Often there are constants, axioms, and functions. There are no *relations* employed in a traditional algebra. The logic is deductive logic *only*, and the statements are *declarative.* Algebraic formulae are translated (in part) to procedures within the calculus so that useful work may be performed (i.e., finding the values that satisfy). Issues, such as decidability, are not interesting in a calculus; the formula is presumed satisfiable (i.e., that it is true for some values) and focuses on finding the values. Unsatisfiable formulae are treated as exceptional cases. see *algebraic structure, calculus, Chomsky hierarchy, compile, domain of discourse, interpretation function, language, model theory, numerical methods, signature, valuation function.*
accessible	If world A can lead to world B see also *conversational background, propositional attitude, states of world*
accessibility relation	A function that is true if world A can lead to world B. See also *propositional attitude*
algebraic structure	An abstract algebra or the list of the major elements that make up an abstract algebra. see *abstract algebra*
ambiguity referential ambiguity	Uncertainty about which of a word's or expression's possible meanings is the one intended. An expression of reference can be interpreted as designating more than one thing. Can be semantic structure (e.g. "every man loves exactly one woman"). See also *diaphoric, vague*
fallacy of equivocation	Treating two distinct meanings of a word as though they were the same.
anonymous	There is not a name (or designator) for the item, relationship, etc.
arguments kind	First kind: objects Second kind: first-level functions of one argument Third kind: first-level functions of two arguments

ATN's

Bill Woods BBN, Debated with rule-based knowledge system. LUNAR system. IBM=Stan Petrick, TansWA.

ATNS: "Too powerful" – costly in terms of memory. Generate too much output.

Seems little different from shift-reduce parsers

See also *parsers*

Terry Winograd, The Blocks World
John Kimbal, Lyn Frazier, Janet Dean, Fodor.

parts

Hold Stack

Lexicon (table of word and properties)

Register

Active stack

History stack.

Everytime the stack is indeterminant:

- Record all relevant information and
- Push it onto the stack

standard conditions: arc labels

Push X: Invoke the X network

Cat X: Check the lexicon t see if word is of category X

Jump

standard actions

Hold, retrieve

Attachment of a new daughter subtree to the top of the stack

Name for jump

Non-standard condition: standard condition and user defined

Nonstandard action.

attribute

'Characteristics, qualities, or performance parameters of alternatives' whereby an item can be distinguished. An attribute is essential: all members of a class must have the attribute. This allows classes to be distinguished.

See also *property, quality*

axiomatic development

Starts with:

- Undefined terms
- Undefined relationships
- Axioms relating the undefined terms and the undefined relationships
- Development of theorems based upon axioms and the definitions

basis

The underlying core 'axiomatic' functions and operations in a language. Other functions, including conventional ones, are defined in terms of these.

belief context

Belief contexts are akin to a possible world and are an important case in intensional semantics.. They seldom introduce a new relation, or a new linking relation. Belief contexts primarily provide a relation's intensional definition.

Table 1: *Belief Context*

Context #	*Relation Name*	*Definitions*
1	loves	see loves[1]() below
1	likes	see likes[1]()
1	knows	see knows[1]()
…		

2	loves	see loves^2()
2	likes	see likes^2()
2	knows	see knows^2()
…		

param$_1$	…	*param$_n$*

Table 2: *loves^1()*

Similar for likes^1(), knows^1(), likes^2(), etc.

believes Believes is a context for interpreting the concept. Set of different belief contexts. Few algebra's of meta logic discuss or are interested in belief.

Trust network adds signatures by others to the <property id, value> pair, and/or a level of certainty. The trust function (metrics) are built on this.

See also *propositional attitude*

bill of materials problem

- Bill of materials
- Product mix
- Quantity on hand (stock)
- Quantity to buy from suppliers. Suppliers offer different price points. Suppliers differ in min/max quantities, identifying suppliers, services agreements.

Query:

```
Select ItemId, Qty
  from B BOM, P Projects
where B . projectID == P . projected

Update from P0, P1
Qty = (select Qty
        from BOM B
       where B . projectId = P1 . projected && B . itemId = P1 . itemId)
```

binding The sense of a symbol having an assigned value.

bound Symbol has a precise value

free Symbol is without value.

Büchi automaton A state machine, similar to a finite state machine, which can take infinite inputs; it includes a set of initial states and a set of good states. (They are also like Kripke structures, but interpreted slightly differently). They are used in model (or protocol) checking to say that some action will eventually be taken after event y (to find cases where this doesn't happen), or that action x will never occur after the event.

See also *bounded model checking, Kripke structure, model checking, temporal logic (linear)*

can see *capability*

capability

Table 3: Different forms of capability/possibility

term		
can	capability	ability
could	capability	ability

did, do, does		emphatic
may	permission	
	possibility	
must	obligation	
shall	obligation	
should	obligation	
will	obligation	
would	possibility	intension

causality Logic has difficulty fully explaining A causes B in a satisfactory way. However, it recognizes the following counterfactual and subjunctive conditionals:

- if B did not happen, than A must not have happened
- if A should happen then B must happen

characteristic function A stylistic means of treating a set as a function. Usually this is used where everything a function but a specific task or pedagogical example is best represented as a set.

choice machine Uses interactive choices as a form of computation.

see also *oracle machines, unorganized machines, universal computation machine.*

Chaitin constant Ω The probability of halting. This value, ironically, is not computable.

see also *Halting problem*

Chomsky hierarchy Each type of syntax can be processed by the type above it. Syntaxes, other than regular ones, allow a production to refer to itself (this recursion is different than the recursive syntax type). The automaton (FSM) is equivalent to the syntax: each can be converted into the other. Each type of automaton has proved to be a distinct specialized area of study.

In mathematics literature syntax *is referred to as* language.

see also *parsing, syntax, regular expression*

Table 4: *Chomsky hierarchy*

Type	*Language*	*Minimal automaton*	*Production restrictions*
0	recursively enumerated	Universal computing machine	None
	recursive	Decidable universal computing machine	None
1	context-sensitive	Linear bounded, NFA	Each production rule has atleast as many symbols on the right side as the left. Example: $a^nb^nc^n$
2	context-free	Pushdown NFA	Only non-terminal symbols on left side of each production. Example: a^nb^n
3	regular	Finite	Every rule has a single non-terminal on left and at most a single non-terminal and/or a single terminal on the right.

closed formula aka *sentence*

see *formula (closed)*

closed world assumption Assumes that everything, and every world, in the world are those that were listed.

see *well-formed problem*

coherence theory See *truth*

complete — complete in the sense that if a set of statements is unsatisfiable, it is possible to decide they are false.

see also *satisfiable, soundess*

completeness — Can all of the possible statements in the language be evaluated (as either true or false)? This is very sensitive to the given axioms, and what statements are admitted. A related question is the rule true in all models of a theory?

see *incompleteness theorem*

compositionality — The meaning of a structure, phrase or formula is a function of the meaning of the elements, their mode of combination, and the context. Decomposition is often concerned with finding such a structuring.

'consistent with' — The hypothesis does not contradict the axioms rules; it may or may not derive from the axioms.

constraints — Controls the range of potential values to which a symbol (variable) may be assigned.

over-constrained — There are too many constraints, making it possible to satisfy them all. There can't be a perfect solution for such a problem, but there can solutions that are close.

conversational backgrounds — The accessibility relations can be determined from this. Includes:

Angelika Kratzer

- A set of propositions
- Accessibility relation
- Set of worlds accessible from a particular world

Time of speech, time of event, reference time.

modal base — The conversational background that determines the set of accessible worlds

ordering source — A conversational background used to determine a partial ordering or worlds.

correspondence theory — See *Truth*

counterfactual — Complex what-if statements ('if X had been the case, so would Y'); intensional logic can be modified to handle counterfactuals by employing similar worlds.

if A were true, B would be the case. Both A and B are true in all worlds that Y(w,A) picks. Y(w,A) picks worlds, based on this one, with a high probability of A being true.

see also *possible worlds, subjunctive conditional, tense*

counterfactual conditionals — Modal auxiliaries, modal adverbs, propositional attitudes, habituals, generics

See also *modality*

Modal auxiliary	It may be raining outside.
Modal adverb	It is possible that it may rain tomorrow.
Proposition	Alice believes it is raining outside.
Habitual	Bob drinks.
Generic	Coffee is bitter.

Table 5: *Distinction between types of counterfactual conditions*

counterfactual historians	"Robert Fogel in his 1962 classic *Railroads and American Economic Growth* – who won the 1993 Noble in economic science – revolutionized economic history by posing counterfactuals and assembling data to statistically test the narratives [frequently put forth]." Counterfactual historians use simulations and game theory to evaluate the necessity and possibilities posed by past events – esp. among WWII re-creationists.[1]	*Postrel, Viriginia, "Rational Exuberance"* New York Times Book Review*, July 22, 2007. Quote edited (ellipsis omitted) for continuity.*
counter part	Opposite of a rigid designator, people are slightly different at each time, place and possible world.	
covering	A logical proposition or a set of propositions *covers* an event or logical hypothesis if the later are the logical outcome of the propositions or can otherwise be deduced from them.	
crossworld identity problem	A problem in possible worlds, where one has to decide who in world B is the same as a given person in world A. This problem is avoided if you use a rigid designator, or kept tractable if you take care with counter parts. see also *counterparts, rigid designator*	
decidability	Do we have a systematic method of evaluating a statement – determining if it is true or false? see also *decision procedure, Entscheidungsproblem*, *Halting problem*	
decision procedure	A stepwise method to decide if a given statement is true or false. see also *evaluation procedure, valuation function*	
de dicto	See *diaphoric*	
de re	See *diaphoric*	
de se	See *diaphoric*	
definitions hierarchical	Hierarchical definitions (e.g. Genus & Species) organize the senses of meaning in such a way that shared properties can be represented concisely.	
dense	In dense domains, given any two non-identical items, there is always an infinite number of items between them. Methods of deduction differ for dense domains (such as real numbers) from sparse domains (such as discrete sets).	
denotational semantics	Semantics of computer programs, discussed in a ponderous manner.	
deontic logic	Deontic logic is a specific type of moral logic, limited to topics of rights and responsibilities. The logic is composed of subjective and counterfactual conditions: obligation and permission; ought and may. Deny, allow, permit. Information security is interested in *users, objects, actions,* and *media/channel* – and relations based upon those. see also *modal logic*	
designator	There are different things meant by designator intension, or extension (set of items	
rigid designator	rigid designators refer to the same entity in every possible world.	
diagnosis	An application of declarative systems, depending on decision trees, examining symptoms, forming and ranking hypothesis. This works well for small systems,	

[1] http://www.wired.com/gaming/virtualworlds/commentary/games/2007/05/gamefrontiers_0521

but not so much with large #'s of components and large numbers of combinations with a large set of observations – a problem of combinatorial 'explosion.'

model based — Starts with a declarative description of normal behaviour, looks for a distinction (or difference) between current behaviour. Consistency-based diagnosis is a specific method.

diaphoric — Distinctions in the intensional reading of a statement

de dicto — Translates "Miss America has always been blonde" into:

$\forall_x MissAmerica(x) \wedge Blonde(x)$

de re — Translates "Miss America has always been blonde" into:

$MissAmerica(x, now) \wedge \forall_t Blonde(x, t)$

de se — Peter see a picture of man and thinks the man is handsome; the pictured man is Peter.

de signo — Value range

disambiguation — A method of handling several words with the same meaning, or a word with several meanings (possible based on sense)

dual — Symmetric logic relations

entailment — B entails C when both encode the same information, possibly very differently.

Entscheidungs-problem — Literally *decision problem* – a question of whether or not one can decide if a statement is true. The Halting problem, where a machine changes what it does to invalidate your prediction (decision), is the classic proof. Most problems are not so perverse, but those that are can be prevented by not allowing such decisions to be known to the system.

see also *Halting problem*

equation — See also *expression, function, interpretation, model, propositions, sentences, situations*

Table 6: *distinction between expression*

equation	A relation between two expression
expression	Doesn't have a relational comparison
function	Assigns unique value to each input
interpretation	Assigns intention
model	Assigns extension
proposition	Set of situations
sentence	Denotes proposition(s)
situation	Part of world

evaluation procedure — A stepwise, often mechanical, process of inferring the values of variables, or it a given statement is true.

see also *intermediary language reduction procedure, valuation function*

explanation of facts principled — "explanations that emerge from a tightly interconnected system of general statements and which lead to further predictions about as yet undiscovered facts."

expression See *equation*

extensible language "A base language which provides a complete but minimal set of primitive. Facilities, such as elementary data types, and simple operations and control constants.

"Extension mechanisms which allow the definition of new language features in terms of the base language primitives.

Semantic extensions: introduce new kinds of objects, data types

Syntactic extensions. New notations for existing or user defined mechanisms.

extension The set of items referred to by a variable or phrase, or that satisfy a sentence or phrase's logical specifications.

see also *intension*

extensional value True/false, quantity or set when the statement (formulae) is evaluated against a specific world, and other context dummy variables.

See also *conversational background*

falsifiability Karl Popper's idea that theories should produce verifiable statements – experiments could decide if the statement is true of false – as a form of negative feedback in the scientific process. Popper argued that knowledge increases as people over-predict and fall back upon falsification,

see also *observability, pragmatics, strength, verificationist*

form
canonical Written in the most standard, conventional, logical way. The rules & process is called *canonicalization*

normal form A simplified form (where otherwise many are possible) that allows some method to be applied. The process is called *normalization*

prenex normal form Strings of quantifiers followed by a quantifier-free portion

formal semantics Referent, extensions, intension,

formal system
characterizing precisely

- Syntax streamlining
- An arithmetization method (e.g. Gödel numbering)
- A definite method of going back and forth between the arithmetic number coding and conventional notation
- To make tractable assign id # to the objects of your attention: each symbol, string, well-formed formula, finite chain of those, proof, etc. get such a number

formula
closed All of the variables are bound – if not as parameters or constants, then by 'quantification'

generalization An accurate statement in precise language of what was found with respect to the tendencies, relationships, regularities or patterning among variables under study.

Gödel number A method for assigning statements a unique number:

1. Setup axioms for the predicate calculus along with a rule of inference by which one can get not formulae from old ones.
2. Set up axioms for standard arithmetic in the language of predicate calculus.
3. Define a numbering for each formula or sequence of formulae in the resulting formal system.

The code number for each symbol (and operator) is then computed:

4. A statement is treated as a string of elements (i.e. symbols and operators).

The index of each element within the string is assigned a prime number.

5. Each prime corresponding to each element is raised to the power of the element's code number.
6. These are multiplied with each other to yield the Gödel number.

symbol	=	+	*	(	)	,	s	x	Y	b	m
code	1	2	3	4	5	6	7	8	9	10	11

Table 7: *Gödel codex*

formula	y	=	B	+	m	*	x
code	2^9	3^1	5^{10}	7^2	11^{11}	13^3	17^8

Table 8: *Gödel code*

formula	y	=	S	(	x	,	b	)
code	2^9	3^1	5^7	7^4	11^8	13^6	17^{10}	19^5

Table 9: *Gödel code*

Gödel theorem In any sound, consistent, formal system containing arithmetic, there are true statements that cannot be proved.

guessing role of

When NFA's have multiple items to choose from at some stages of computation, they perform faster (than other methods) if they guess well; backtracking allows them to handle when they guess wrong. Oracle machines can compute, with less complexity (e.g. faster), some problems that other machines would do poorly. Most applications of these methods are guessing the intent in parsing, or solving difficult numerical problems (an application in satisfaction). One form of guessing is to try all acceptable paths at once, (but only track all states the machine may be in, rather than the path).

see also *back-tracking, oracle machines, regular expression, satisfaction, witness function*

halting problem A classic 'decidability' problem: Given a Turing Machine and a tape (program), decide whether the machine will halt while running that tape. If a Turing Machine or program is allowed to know the answer to that question, and it can do that opposite – if you say it will halt, it makes sure that it will not; if you say that it will run forever, it chooses to halt.

see also *Entscheidungsproblem*

Heckel's algorithm Finds the longest recurring substring.

See also *Bentley-McIlroy matching*

ILOG Tools: solver, scheduler, dispatcher, configuration

Terms: powerful, advanced, versatile, easy, clear

Application	Long-term	Published schedule	Operational Schedule
Scope	Strategic	Tactical	Operational
Timesteps	Month	<-Week->Days->	Hour
Drivers	Money	…	Feasibility
Technique	LP	MIP/Hybrid	CP

Table 10: ***Planning horizon***

incompleteness Gödel showed some systems – including sufficiently strong and consistent – will have statements that can't be evaluated. Some such or similar systems are also undecidable.

see also *Entscheidungsproblem, Halting problem*

'independent of' The hypothesis is not a logical outcome of the axioms – they can't be combined in a finite number of steps to generate the hypothesis. This hypothesis may or may not valid.

indexical terms Terms defined by context

See also *conversational background*

intension The formula or statement as declarative logical specification. The formulae is marked against which world/time it is to be evaluated against (to extensional form).

see also *extension*

intermediary language There is some debate whether a model's interpretation should be described in terms of a machine (i.e., an evaluation procedure), or translated into a set of declarative statements that must be used by another model to infer the values. With a single model, or small number of models, it is simpler to use a direct evaluation procedure. With a large number of models, it may be easier to translate each into a more sophisticated intermediary language. This also reduces the combinatorial complexity of translating from one language to another: either you need two translators for every language (two and from the Intermediary language) or $2n^2$ translators.

see also *compiler, evaluation procedure, valuation function.*

interpretation

interpretation function Interprets sentences in the language. The language can be very simple or complex. I'm not familiar with any past a Chomsky level 2. The language can be a non-trivial language of any form that can be systematically interpreted.

see also *evaluation procedure, intermediary language, valuation function*

Kripke frame k set of states

i<= j relation, compatbility of these states

values/ models

Kripke structure A non-deterministic automata:

- The nodes are the reachable (possible) states,
- The edges are the operations that changed the state
- Something that maps the state (node) to what it represents

Checks that temporal logic formulae are valid. A counter example is a trace of the system that violates the property. State transition structure; each state is a value at time. All behaviours of the Kripke structure satisfy or violate formula.

language elements Indexicals (items that are identified in context), determiners, quantifiers, propositional connectives

formal Includes axioms, entities are grouped by classes. Much of the discussion of a formal language involves details of the syntax – especially what constitutes a *well-formed formula* – precluding the much larger issue of meaning and methods of evaluation.

fragment A language fragment is employed for any of the following reasoning: for tutorial purposes (pedagogy), it is as far we got in the analysis of larger (possibly hypothetical) language, or if the language elements used to describe a problem conform to the language fragment, there may be many advantages to using the special-case.

metalanguage Describes an object language. This is needed to define the truth notions about an object language. An object language that is allowed to 'describe' which of its statements are true or false will often be 'incomplete' – see the Halting problem.

object language — In logic the language of the axioms; the language or logic under study.

license — Fields: permits, requires, prohibits, jurisdiction, deprecation date, legal code URL.

Toulmin style argument

logic problems — Logic problems tend to become very narrow and regimented into purpose, form, style and method. However, they are not very interesting.

logicians — Logicians study structures – a set with relations and functions. "Space" is a structure, so is first order logic with equality.

many valued
Jan Lukasiewicz — True, false, possible

see *modal logic*

map — A model that connects two models

Markov — What can a markov table represent?

Regex: the old* man ate.

What can't they represent?

A::= '(' A ') | Foo

See also *parsing*

metadata — Metadata answers important questions about information:

- What do I have?
- What does it mean?
- Where is it?
- How did it get there?
- How do I get it?

Andrew Tanenbaum, Metadata Solutions: Using Metamodels, Repositories, XML, and Enterprise Portals to Generate Information on Demand, *Addison-Wesley, Boston, 2002*

methodology$_1$ — Pretentious way of saying 'method'

methodology$_2$ — Study of methods employed

modal logic — Concerned with constructing a logic calculus that includes the following operators:

- Believe
- Before/after
- Necessary that
- Will happen
- Possibility

There is disagreement whether a modal logic may take on more. These operators can apply to sequences as well as episodic events. They have slightly different definitions (and implications) in tense, moral, epistemological, derivation, and other contexts.

see also *deontic logic, reference point, temporal logic, tense logic*

***Table 11:** Different forms of possibility*

	'truth' analogue	*Definition*
Logical	Coherence	*X* is possible, being allowed by a definition
nomological	verificationist	*X* is possible, being allowed by the laws of nature

nomohistorical	correspondence	*X* is possible because it doesn't violate the laws of nature, compatible with actual history

***Table 12:** Modal operators in more detail*

Operator	*Intension*	*Notation*
after	A before B	A after B
before	A < B	A before B
necessary	$P(A)=1$ $P(A \to B)=1$ if A is not unrelated to B: In all possible worlds that A is true, so is B; if unrelated, it is not likely.	A must happen when A happens so must B
possible	$P(A)>\varepsilon$ $P(A \to B)>\varepsilon$ if A is not unrelated to B: There exists a possible worlds where A & B; if unrelated, this merely Possible(B)	A can happen when A happens so may B

modal system A modal system uses a language, a set of worlds, and a means of relating the worlds. The worlds might correspond to alternative outcomes for choices (they would be related by a common choice point), beliefs, tableau of facts, etc.

see also *model world*

***Table 13:** Distinction between types of modality*

Type	*Distinction*
ability	
alephic modality	Necessary, possible; if it is necessary must be possible
circumstantial / dynamic	
deontic	Permissible, obligatory
epistemic modality	knows, believes
temporal modality	always, sometimes

model$_1$ A binding of variables to values. See also *satisfaction*

model finder A satisfaction procedure, finds the bindings of variables (the model) that make the specification true; often a constraint solver (compile and hand to a SAT solver)

modeling language Expresses structure constraints and behaviour

model$_2$ A formal framework for using a few central relationships to represent the basic features of a complex system; models discard important elements and philosophical considerations: they are *not* truth. Models are often described by their role, elements, and test of specification error.

Models are "clipped and pruned till they resemble the conventional birds and animals of decorative art." Alfred Marshall.

Models should be open about the underlying theoretical principles. These principles must have a concrete form in definite algebraic terms. The model should be transparent about its connections, mechanisms, and flow, coupling effects to outputs. It should be easy to tinker with, yet the user should not have to understand exactly how it works. What are the (hereto fore) unseen expectations?

see *endogenous variables, functional explanation, Markov model, Poisson model*

***Table 14:** Distinction between related terms*

Term	*Distinction*

emulation	Imitates the behaviour of a system, without concern for internal processes
evaluation	To assign a value to an expression
execution	A sequence of instruction passed to an external interpreter
interpretation	Assigns interpretation.
model	Assigns extensions – the values and sets
paravirtualization	Similar to virtualization, except it presents the illusion of a device slightly different from the underlying hardware.
simulation	Mimics the behaviour of a system, with a high degree of fidelity to internal processes, state, etc.
virtualization	Effects the illusion of each user of a device being the only user; the multiplexing software typically saves and restores the state context for each user.

Table 15: *Distinction between types of models*

Type	*Distinction*
analogical models	
behavioral	Imitates the behaviour of a system, with-out concern for internal processes
declarative models	can represent important aspects of static systems, but dynamic systems are largely beyond their ability. Most tense analysis in modal systems treat histories as points in time with different sets of facts, ignoring change.
idealized models	
Measurement models	Maps measurements to their theoretical constructs
Parametric models	Predicts values, especially when observables and/or actions are primarily numerical.
Phenomological models	
Statistical models	A type of behavioural model based on probabilities
Structural models	Maps causal and correlative links between theoretical variables. Specifies components and interconnection, often a structural model is a specific implementation.

Table 16: *Modeling techniques per scale of system*

Scale		*Modeling technique*
sub-atomic		Quantum theory
atomic	group	Thermodynamics
	individual	Mechanics
Person	individual	Automaton
		- Cellular automaton
		- L-system
		Diffusion equations
	group	System dynamic theory
	group	Markov / Stochastic walk

	group	Boids
	groups	Network theory
cosmos		Relativity

***Table 17:** Technique for a given circumstance*

Circumstance	*Technique*
Decisions in competitive circumstance	Game strategy
Lines	queuing
Many others	queuing
Minimizing costs	stock control
Future events directly affected by preceding events	markov chains
Attention to risk	project management
Explore Ideas	simulation – without loss or humiliation
Time-based events	queuing

analytical model — Set of formulae that map program characteristics, and architecture characteristics to performance characteristics. Employed to identify relative usage level of subsystems, power, etc.

behavioural model — Describes the system primarily using

- Its actions and actions of its components,
- Its interaction with the outside world,
- Interactions of its components,
- Causality relation

Describes the function and timing, independent of a specific implementation.

see also *functional explanation*

economic models — Modeling economics poses a challenge since economic relations are very vague. Relationships only have a topology, but no definitive structure. (Does a rise in output, mean a small linear change, exponential, or a probability?) This means the integration of changes will be way off. The relationships may be wrong, or purely ideological; they may be correlative for a while, but the correlations may disappear once the state or other factor tries to manipulate them. Can't predict results based on the results under an old regime.

Many of the elements are linked in a complex system of symbolic equations. They are not sufficiently independent or isolated to examine a subsystem; to solve one part, you need to solve all of the equations simultaneously. Easy to have results that cannot be predicted with naïve models. The messy transitions of the real world are not predicted.

There are genres of economic models. Macro-economic models to demonstrate the circular flow of the economy. Computable General Equilibrium models: these focus on the underlying structure of the economy, ignoring business cycles variations. They can capture one-off difference policy but not the recurring, continuing effects.

equivalent models — It is common for many kinds of systems to find an equivalent electrical circuit, mechanical, or acoustical system. The help promote understanding of the behaviour. There are many specialized techniques that can be applied to a particular form, as well techniques to convert between the representations.

Olson, H F, Dynamical Analogies, *D Van Nostrand Co, 1946. This book provides great detail on electrical circuits equivalent to mechanical and acoustical systems.*

identification — Constructing a model by parts and specification

limits of models — Models are not independent checks of their creators: models largely exist to codify a view. Some limits include: experts have their own incentives, there is a high demand for models, no matter their quality. Model selection and designed is to confirm the researcher's ideology, based on (in part) topological and structural changes.

models in logic — A model represents a particular context in which a little algebra is evaluated – a system of axioms, operators, rules for combining variables and operators into formulae, a set of entities, their properties and relationships, and a specification of the language relates to those entities and relationships, constraints on what properties there are. These models allow only deductive logic.

see also *valuation function*

non-standard — Alternative interpretations. Try to rule out those interpretations with ambiguity, although this can be hard to spot. Things other than intended may be well described by the model.

numerical model — Numerical models provide numerical answers to policy questions.

partial model — Only can evaluation some statements.

physics — Series of equations of state, relationships between material bodies, and describe their movement, action, behaviour, etc. This is usually divided into parameters, expressions, functions, geometry, coordinate system, materials, analysis.

satisfaction models — In order of increasing difficult: parameters are independent; pairwise; all pairs.

statistical model — Combines analytical models and simulation to create a typifying trace.

Mathsoft S-Plus 2000, Guide to Statistics Vol 1*, 1999 p15*

1. "Determine the variables to observe." These variables link to "the hypothesis being tested" or "the phenomena being modeled."

2. "Collect and record the data observations."

3. "Study graphics and summaries of the collected to data to discover and remove mistakes and to reveal low-dimensional relationships between variables.

4. "Choose a model describing the important relationships seen or hypothesized in the data

5. "Fit the model using the appropriate modeling technique

6. "Examine the fit through model summaries and diagnostic plots

7. "Repeat steps 4-6 until you are satisfied with the model"

structured models — Means of evaluating a model's quality and characteristics.

model checking — A method for verifying whether an implementation satisfies a design specification. The implementation is translated into a model from which a system state machine can be derived. The specification describes properties, and the checking verifies that the state-machine satisfies them.

model structure — Moments, individuals, agents, concepts, attributes, values, predictions, beliefs

model theory$_1$ — study of formal languages and their interpretation

model theory$_2$ — Concerned with making models of a theory. A theory has a model if and only if the theory is consistent. Such a model is a language with an abstract algebra to implement the semantics. An interpretation function that maps language elements to constants, functions, and predicates. The description of the language is often a table with the syntax and how to evaluate predicate phrases of that syntax. The syntax: the kinds of variable (if the language is typed) and how they combine with operators and other variables. The set of entities allowed may be more than a variable – it may include more complex noun phrases, e.g. GlobalCheckFor $var.

Discussions of such models focus largely on the syntax (esp. *well-formed formula*) although the issues with interpreting meaning and finding satisfactory solutions is of greater importance in the long term (a language is learned 'once' but used for a long time), and more difficult.

see *archetypical language understanding, evaluation*

model world Composed of

- A set of possible elements
- A set of possible attribute names
- A set of possible attribute values
- A set of possible world states

see also *universe of discourse*

monotonic logic More predicates don't change outcome

non-monotonic logic Where other data affects outcome, even if it is not really relevant.

Montague It is composed of:

- functions are the central organizing tool for phrases and words
- events (and manifestations within time and space)
- processes
- states
- properties
- actions

Table 18: *Accessibility relations for propositions*

what	*authority*	*description*
adverb	Montague	f:proposition → t/f
'believes'	Montague	f:individual×proposition → t/f
		relation between individual and proposition
determiner		quantifier on a set
indeterminate phrase	Montague	predicate
indexical		
individual	Montague	e st e∈E
	Cresswell	f:world → subworld (of the given world)
individual concepts	Montague	f:world×time → {i|i is an individual}
	Cresswell	f:world →individual
name		{property set | name ∈ property set}
		the of sets which name is part of
noun (singular)		
noun (plural)		{s | s is an n}
noun phrase		general quantifier
property	Montague	f:world×time →sets
property of a noun		the sets mentioned earlier for noun
situation type		f:relation×individual → t/f

verb	Montague	relation

90-10 Relational databases need to pull in (IO) and consider data that is about 9 times the size of the resultant set.

necessary It must be true; it cannot be otherwise. Defined operationally as X is true in all worlds at all times.

nomological Criteria for determining if a deductive nomological explanation is worthy of acceptance:

1. The explanation contains at least one law established by evidence and accepted as true.
2. The law(s) are employed in explaining the event
3. The sentences describing the initial conditions are true
4. The description of the event is true
5. The event's description is a deductive consequence of the laws and initial conditions.

nondeterministic The next state is not completely determined by the current state and symbols in memory. A set of next possible states is so determined. Backtracking is often be employed. Non-deterministic finite automata recognize the same class of languages as deterministic finite automata, but typically have fewer states than a DFA, and are faster to construct. The backtracking often slows down execution time for a NFA, so one might employ a DFA when NFA features are not needed, or to use a DFA to find likely interesting matches, then switch to a NFA.

see also *Chomsky hierarchy, deterministic*

normal form standardizes local formula into a specific format. Types of analysis (i.e. family of algorithms and measures) prefer one specific form.

clausal normal form

```
cnf::= disjunct
cnf ::= disjunc ^ cnf
disjunct ::= literal
disjunct ::= (literal ˅ disjunct)
literal ::= term
literal ::= !term
```

disjunctive normal form

```
dnf::= conjunct
dnf ::= conjunct ˅ dnf
conjunct ::= literal
conjunct ::= (literal ^ dconjunct)
literal ::= term
literal ::= !term
```

notation Often a skillful choice of reference system simplifies the work.

selecting The choice of notation depends on:

- The kinds of problems you're trying to solve
- What environment you're trying to solve it in
- With whom you're trying to solve it
- How does the problem or task decompose into a given notation
- How easy is the problem to solve in the framework?
- How elegantly?
- Will it perform well?

numerical Solving questions of valuation is better with (computer) analytic rather than symbolic method. Most realistic problems can't be solved analytically. There is no

methods single method (or a small number of methods) that both suffices and is tractable. Each potential definition substituted for a given relation name requires a different method to solve – each is a different problem. Worse, descriptions involving differential equation are even more difficult than the rest: solutions of differential equation is a large of subfield of math.

observability You might think that is X is not observable, is not worth talking about, or not unless it has some further level of interest. Similar observable, and seen as false, no one would talk about it.

see also *falsifiable, pragmatics, verificationist*

oracle machine An oracle computes a f() in finite time that the Universal Computing Machine can't do. This allows computation that a UCM can't do – or tractably do. In satisfaction problems this often takes the form stochastic and probabilistic methods.

see also *witness function*

Copeland, BJ. Proudfoot, D. "Alan Turing's Forgotten ideas in Computer Science." Scientific American*, April 1999, V28N4. p 98-*

order Usually the number of parameters.

See also *rank*

parsers

A parser converts a sequence into another sequence:
$\text{Output}_j = \text{Parser}_{i,j}\ \text{Sequence}_i$

this involves:

- lexical: turning it into words and symbols
- parsing based on the syntax
- resolving the named variables, functions, types, and other elements
- semantic actions based on matching the patterns

Special cases of Parsers:
Top-down: LL(k)
Bottom-up: LR(k)
k = the amount we need to look ahead

Objectives:

1. Minimize the amount we need to look ahead
2. Minimize backtracking
    a. # of times we ned to back track
    b. Max depth we would back track
    c. Average depth we would back track
3. Minimize the amount of state need to keep
4. Minimize work parser does. Backtracking, tests.

See also *ATN, Chomsky hierarchy, Markov, regex, shift-reduce,*

LALR(1) An approximation to LR(1) parsing.

Frank DeRemer, MIT PhD thesis, 1969

LR(k) Bottom-up parser that became the definitive parsing solution (surpassing precedence methods).

Donald Knuth "On the Translation of Languages from Left to Right" Information and Control*,* **8** *p607-639, 1965*

precedence 1963 Floyd: operator precedence
1966 Wirth: simple precedence

static parsing Take piece of text, determine its structure without executing it.

places
kind

first kind: suitable for proper names

second-kind: names of first-level functions of one argument

third-kind: names of first-level functions of two arguments

possible worlds Interested in counterfactual and subjunctive conditionals as well as notions of causality. Possible worlds, being imaginary, are difficult to reconcile. The concern is how much else can be true in such a world. No approach is entirely satisfactory in what else might be true.

Structure and relationship of worlds (but not of a world). David Lewis Counterfactuals

Method to decide if a formula is true/false

GE Hughes, MJ Creswall, An Introduction to Modal Logic. *Summary: Method to decide if formula is true or false.*

Form of empirical data. Fundamental tenet that data is stored as row & columns in tables; we treat it as accessible in terms of rows and columns

See also *belief context, propositional attitude*

possibility It might be the case that. Operationally defined as a world and time exists that it is true in.

predicate It is a phrase posited to be either true or false. It includes atleast one variable, attribute or function; it may include an operator. There is often atleast one free (unbound) variable. Not all predicates are genuine properties.

see also *sentence*

problem solution search

1. Start with users knowledge of problem
2. Clear separation of constraints and combinatorial search
    a. Discrete variables represent the primary decisions in the problem
    b. High-level constraints represent the relationship between variables
    c. Constraints can be combined to match the real-word's complex constraints
3. Generate multiple solutions quickly
4. Refine solutions

procedural semantics The operations that one is supposed to carry out (rather than merely discussions of possible facts). Meaning that a statement takes action or changes the world. Backtracking can be very expensive (by throwing 'exception'), unreliable (errors reversible only by best effort) or not possible at all (as with destructive operations).

property_1 An attribute (i.e. shared by all members of a class), often one that can be measured;

See also *attribute, quality*

physical property That which can be measured and observed with changing the composition or identity of a substance. Some physical properties are defined as a relation on two vectors.

chemical property In order to observe this property we must carry out a chemical change.

extensive property Depends on how much matter is being considered.

intensive property This measured value does not depend on how much matter is being considered.

macroscopic property Measurement determined directly.

microscopic property Measurement determined by an indirect method.

property_2 A function that returns, for a given situation, the set of entities that are in that state or express that features. For example, the property ***is-asleep*** returns the set of people asleep in a given situation. This definition is reverse of the conventional one.

proposition

propositional attitude A relation between individuals and propositions. Applies to *believes, know, doubt, regret, hope,* etc.

Jaakko Hintikka

See also *belief context*

Table 19: Accessibility relations for propositions

Proposition	*Individual*	*World1*	*World2*
believes	Bob	1	2
believes	Bob	1	27
...			
knows	Bob	1	3
knows	Sally	7	31
...			

Table 20: Proposition attitude

Proposition	*Individual*	*Function*	*Context #*
believes	Bob	believes()	2
believes	Sally	believes()	3
...			
knows	Bob	knows()	3
knows	Sally	knows()	31
...			

propositional connectives Boolean operators (not, and, or, etc.) or set operators.

puzzle SEND+MORE=MONEY

Dudeney, Strand Magazine, 1924

- 'VIOLIN * 2 + VIOLA == TRIO + SONATA',
- 'SEND + A + TAD + MORE == MONEY',
- 'ZEROES + ONES == BINARY',
- 'DCLIZ + DLXVI == MCCXXV',
- 'COUPLE + COUPLE == QUARTET',
- 'FISH + N + CHIPS == SUPPER',
- 'SATURN + URANUS + NEPTUNE + PLUTO == PLANETS',
- 'EARTH + AIR + FIRE + WATER == NATURE',
- ('AN + ACCELERATING + INFERENTIAL + ENGINEERING + TALE + ' +
- 'ELITE + GRANT + FEE + ET + CETERA == ARTIFICIAL + INTELLIGENCE'),
- 'TWO * TWO == SQUARE',
- 'HIP * HIP == HURRAY',
- 'PI * R ** 2 == AREA',
- 'NORTH / SOUTH == EAST / WEST',
- 'NAUGHT ** 2 == ZERO ** 3',
- 'I + THINK + IT + BE + THINE == INDEED',
- 'DO + YOU + FEEL == LUCKY',
- 'NOW + WE + KNOW + THE == TRUTH',
- 'SORRY + TO + BE + A + PARTY == POOPER',
- 'SORRY + TO + BUST + YOUR == BUBBLE',
- 'STEEL + BELTED == RADIALS',
- 'ABRA + CADABRA + ABRA + CADABRA == HOUDINI',
- 'I + GUESS + THE + TRUTH == HURTS',
- 'LETS + CUT + TO + THE == CHASE',
- 'THATS + THE + THEORY == ANYWAY',

quality Distinguishing essential attribute or characteristic property.

See also *attribute, property*.

quantifier few, many, more .. than, each, almost all, etc.

elimination Quantifiers can be eliminated, in some circumstances, allowing easier analysis. The approach is to try to show it is equivalent to another statement, one without quantifiers. The later can be evaluated in a fixed number of steps.

generalized Set theoretic notation, primarily using set disjunction (and count) to verify. The quantifier is the comparison, number, and set expression (whose cardinality is examined):

All A are B	$\lvert A \cap B \rvert = \lvert A \rvert, A \subseteq B$
Some A are B	$\lvert A \cap B \rvert > 0$
# A are B	$\lvert A \cap B \rvert = \#$
No A are B	$\lvert A \cap B \rvert = 0$
Most A are B	$\lvert A \cap B \rvert \geq 0.5$

rank The rank of a formula is greater than or equal to the rank of each of its elements, operators, and parameters.

See also *order*

reducibility The reverse of composability, concerned with decomposing statements into observable terms.

reduction procedure Converts a declarative language into a procedural one.

see also *compiler*

reference A symbol may refer to something (usually this must be done thru a distinct meaning).

reference point Used in modal logics, a formula has two clauses, both with their own modal operators. With tense logic there is often a reference time.

see also *modal logic*

regimented Orderly separation of premises, facts, and conclusions so that conclusions are true in a stricter sense – by preventing invalid ones.

Truth separated into analysis outside of the language (see Halting problem)

Deductive vs inductive methods.

regular expression

Two regular expressions are equivalent if they recognize the same set of strings. Regular expressions can be differentiated using a set of rules analogous to Leibniz rules of differentiation. Given a regular expression R_1, the derivative (with respect to symbol 'a') is a regular expression R_2. R_1 recognizes the strings matched by R_2 when they are prefixed by 'a'.

See also *Chomsky hierarchy, the method affine transforms for generating strings.*

Summary: A description of how neurons behave, a pre-cursor of regular expressions
Warren McCulloch and Walter Pitts, "A logical calculus of the ideas imminent in nervous activity," Bulletin of Math. Biophysics *5 (1943) (reprinted in* Embodiments of Mind, *MIT Press, 1965)*

Summary: A regular expression compiler (targeting the GE-TSS machine), using an NFA.
Ken Thompson, "Regular expression search algorithm," Communications of the ACM *11(6), June 1968, p 419-422. (http://doi.acm/org/10.1145/363347.363387)*

Janusz Brzozowksi, "Derivatives of Regular Expressions" Journal of the Association of Computing Machinery, *V11N4 (October 164), p481-494*

Table 21: Regular equivalences

	Equivalent to
a^*	aa^*
ØX	Ø
{empty string}X	X
(Ø|X)	
({empty string}|X)	

Table 22: Symbolic differentiation of regular expressions

	Equivalent to
$\frac{d}{da}b$	Ø (b ≠ a)
$\frac{d}{da}a$	{empty string}
$\frac{d}{da}a^*$	a^*
$\frac{d}{da}a^+$	a^*
$\frac{d}{da}XY$	$\left(\frac{d}{da}X\right)Y$
$\frac{d}{da}(X\mid Y)$	$\left(\frac{d}{da}X\mid\frac{d}{da}Y\right)$

relation algebra

Variables – properties of an entity – are compared. In CS this is used to specify sets of entities. In bulk, files of fixed-length records of multiple fields, which were selected and merged.

Summary: Relational DBs are a relabelling of existing practices promoting a pretense.
Henry Baker, letter to ACM, *Oct 15 1991, http://home.pipeline.com/~hbaker1/letters/CACM-RelationalDatabases.html*

Table 23: Regular to Relation translator

	Relational
fields	Column
files	Relations
merges	Joins
pointer	Key
records	Rows

resolution

Rule that yields inferred clause

resolution method

A technique to solves truth-conditional problems in clausal form; typically this is further restricted to conjunctive normal form. Works by testing almost every combination of variable assignment against the rules, keeping only those that do not contradict. Set of support are the primary and supporting axioms; no two primary axioms are resolved against each other.

John A Robinson "A machine oriented logic based on the resolution principle." Journal of the ACM *12(1):23-41 January 1965, Syracuse University*

First, prepare the formulae:

1. First negate the theorem to be proved, i.e. make F into ~F
2. Adjoin ~F to the axioms
3. Rewrite the system as: ~F, A1, …,An

The method involves five steps:

1. Resolve pairs of clauses until a contradiction is reached; this is done by unifying the variables and treating each clause is the theorem to be proved in its own resolution process. The resolution of each clause also provides further unification information

2. F has been proved if a contraction was found. Otherwise cannot be proved by the axioms.

See also *satisfaction, unification*

clausal form

Each term is either a variable, or f(x1,..,xn) (where f is a function of n arguments, and $x_1..x_n$ are terms). Formula's are of 3 kinds:

1. Atomic – any predicate the arguments of which are terms. All atomic formulas are formulas

2. If F∨G, F∧G, ~F are all formulas if F,G are formulas,

3. All F(v), Exists F(v) are formula, if F is a formula and v is a variable.

unit preference strategy

Choose clauses that are as short as possible to unify and resolve.

satisfaction
Carnap

The values a formula is true for; if true for the value or range of values. Or, rather, checking that a symbols value is consistent with the constraints.

See also *resolution method, unification*

Tarski

Every possible value for every variable in the universe, so long as the formula is true.

boolean

Givens:

A set of variables: $v_0,\ldots v_n$
A formula using those variables

Platzner, Marco "Boolean Satisfiability" IEEE Computer, *IEEE Computer, April 2000, p60*

Summary: based on binary Hyper-Resolution & Equality Reduction can solve many SAT problems without search. Bacchus "Exploring the Computation Trade of more Reasoning and Less Searching" 2002

Assign each variable a value (0,1) such that the formula evaluates to 1 – or find all such valid assignments. This is an NP complete task.

Steps:

1. "Decision step selects a variable for the next assignment, either statically with a fixed variable order, or dynamically, depending on information gathered during search.

2. "Deduction step infers information from the current partial assignment. Boolean constraint propagation… exploits the fact that a partial assignment can imply values for other variables.

3. "Diagnosis step analysis [a] contradictions' cause and uses the inferred knowledge to search more efficiently."

see also *BDD (binary decision diagram), bounded model checking*

parameter search problem

Givens:

Initial & boundary conditions
A set of constraints
Technique to solve the problem

Algorithm:
Starts by making an initial guess for the parameters
Calls the objective function & continues to adjust parameters to minimize the objective function. If the results are not satisfactory, repeats, finds the best parameters with fewest evaluations.
Evaluating the objective function. Calls differential equation and compares them with real data.
Differential Equation solver. Returns solution of ODE's for current guesses.

semantic presupposition

Relates two prepositions, Regrets(who, preposition): *Bush regrets that he made Noriega Attorney General.*

see *counterfactual, subjunctive conditional*

semantic resolution

Starts out by assuming the variables to be true.

Nucleus(+) is a clause that evaluates to be true

Electron(-) is a clause that evaluates to be false

Rules:

1. Never resolve a nucleus with a nucleus
2. Resolve an electron with a nucleus only if the variable to be eliminated has the highest priority among the variables that appear in the electron

Anthony J Dos Reis, "Theorem Proving using Semantic Resolution" DDJ *Apr 88 #137* ***V13I4****, p50-52*

C Chang, RC Lee, Symbolic Logic and Mechanical Theorem Proving, *1973 Academic Press, New York NY, ,*

PH Winston, Artificial Intelligence, *2nd Ed, 1984, Addison-Wesley, Engelwood Cliffs, NJ*

semantic clash

A single nucleus together with a set of electrons that eliminate all its true literals under semantic resolution. Contains exactly on electron for each true literal in its nucleus.

proof by refutation

what we want to show

P_1 & .. & P_n <-> C

So try

P_1 & .. & P_n & ~C <-> []

resolution, forms of

(E_1 v E_2) & (~E_1 v E_3) <-> E_2 v E_3

(E_1 v E_2) & ~E_1 <-> E_2

(E_1 & (~E_1 v E_3) <-> E_3

E_1 & ~E_1 <-> []

sentence

see *closed formula*

sentential function

see *formula*

shift-reduce parsing

BNF grammar is converted into a series of nodes like:

- A link to the symbol table
- Whether or not the item can be a null match
- List of next states

The list of next states is made when checking the network

The symbol table is three parts:

- The symbol (character) which is matched
- The operation: Shift (which state to go to), reduce (number of items and which action to take), accept, error.
- Hint: the extra bit of information for the operation

Then it builds a TRIE, assigning a number to each node first. The Trie is like the symbol table, except that shifts have state numbers, and there is a column for number. Then all of the symbols for next state are added, given a reduce step. Finally, all of the remaining symbols are added, and given an error state.

See also *parsing*

signature A list of the operations available in the abstract algebra.

see also *algebraic structure*

situation

Skolemization A method of eliminating quantifiers, at least in certain circumstances

sound whatever the process announces is correct.

standard model Intended interpretation

See also *non-standard models*

states of the world A set of index numbers. Each is a code for a possible state of the world.

A distinguished index (why?)

A relation R on the indices; R(x,y) is either boolean (true or false) or relative probabilities

V(x) assigns a valuation to each state-of-the-world index number. Statement A is true or false based on V(x)(a)

strength Usually the strong form is "All X is Y", while the weak form is "there Exists an X that is Y". While the weak sentence is more likely to be true, logicians aggressively reflect the larger culture disposition towards strength – and what one can do with it. Deductive systems tend to use the strong form; it is biased to assume that the more deductively true conclusions are correct.

see also *falsifiability.*

strictness see *strength*

subjunctive conditional Concerned with the future: 'if X should be the case, then so will Y'

see also *counterfactual, tense*

substitutability Objects of subtypes should behave like parent type. Stacks and queues are not subtypes of the other. push push pop sequence yields different result (one is FIFO while the other is LIFO). They are in the same family, have similar signatures (sets of key operations), store items for later, different disciplines.

tableau Tables of facts and derivation rules.

See *resolution method, unification*

temporal logic Temporal logic is, largely, the same as modal logic, except that it focuses more on the analytical needs of computer science. Primarily declarative statements used to validate the behaviour of various systems.

see also *Büchi automaton, clock, modal logic, tense logic*

Allen's interval algebra
X takes place before Y
X meets Y (one starts when the other ends)
X overlaps with Y
X starts Y (start of X == start of Y, duration of X < = duration of Y)
X during Y
X finishes Y

X is equal to Y

linear time

Linear time is represented a sequence of events (there is no concept of duration), augmenting prepositional logic with 8 operators describing the past and future:

- Always after (in the future),
- Sometime after (in the future),
- Until
- Next cycle
- Always in the past
- Sometime in the past
- Since
- Previous cycle

This can be used to analyze contracts and behaviour of procedures or algorithms. This logic can be extended with counts or back-references.

See also *Büchi automaton*

metric

Extends linear temporal logic with the concept of duration – each operator allows an upper and lower bound on duration.

tense
and reference time

Tense reflects a combination of philosophers and logician competing senses of what time should be. The description of tenses in language invariably describes how *Latin* should be used and ignores the distinctions of the languages under study.

Of special concern are sentences with phrases that must unify their time sequence:

When Jack opens the door, Helen sees the books

The idea is separate the analysis into three kinds of time:[2] time of speech/statement (S), a reference time (R), and a time of event/state in each phrase (E_1, E_2,…)

The tense indicates when the event happened relative to the speaker, when the statement was uttered relative to a reference time, if the event (or state) has finished, or is continuing.

The perfect tense is that the event happened and is finished; imperfect is that the event happened and is continuous or still going on.

There is some debate; some sentences employ conventional counterfactuals, others value the future but it is not clear whether it is an event that is *possible* to occur or *necessary* to occur. Tense also has issues with handling a narrow region or window of time (*Jimmy Carter has been elected President)*

See also *Clock, Temporal logic*

Table 24: *Tense of when the speech act took place*

	Time of speech
past	S>R
present	S=R
future	S<R

Table 25: *Tense of when the state or event took place*

	Time of event / state
perfect	E<R
simple	E=R
posterior	E>R
imperfect	E_{begin}<R, E_{end}>R

[2] Riechenbach, Hans 1947 *Elements of Symbolic Logic*

Tense	Example	Time of speech	Time of event
simple present	Jack sings	S=R	E=R
simple past	Jack sang	S>R	E=R
simple future	Jack will sing	S<R	E=R
present perfect	Jack has sung	S=R	E<R
past perfect	Jack had sung	S>R	E<R
future perfect	Jack will have sung	S<R	E<R
posterior present	Jack is going to sing	S=R	E>R
posterior past	Jack was going to sing	S>R	E>R
posterior future	Jack will be going to sing	S<R	E>R

Table 26: *Tenses in more detail*

Tense	Example	Time of speech	Time of event
counterfactual	If Nixon had won in 1960, we would have …	S=R	E<R
subjunctive	If Jack should sing, I'll like it	S=R	E>R

Table 27: *Counterfactuals in more detail*

operation	Definition
after	The instant A is later than the instant B
before	The instant A is earlier than the instant B
future (strong)	It will always be the case that A
future (weak)	It shall be the case that A
happen	It is the case at the instant A that B
past (strong)	It has always been the case that A
past (weak)	It has been the case that A
present (strong)	It is always the case now that A
present (weak)	It is the case now that A

Table 28: *Modal operations*

Tense	Intension	Notation
after	A before B	A after B
before	A < B	A before B
future (strong)	$\forall_\tau[\tau \textit{ After } R \rightarrow (A,\tau)]$	A always shall happen
future (weak)	$\exists_\tau[\tau \textit{ After } R \rightarrow (A,\tau)]$	A always shall happen
happen	(A,τ)	
past (strong)	$\forall_\tau[\tau \textit{ Before } R \rightarrow (A,\tau)]$	A always has happened
past (weak)	$\exists_\tau[\tau \textit{ Before } R \rightarrow (A,\tau)]$	A has happened
present (strong)	$\forall_\tau[\tau = R \rightarrow (A,\tau)]$	
present (weak)	$\exists_\tau[\tau = R \rightarrow (A,\tau)]$	A is happening

Table 29: *Modal operators in more detail (R is a reference point)*

terms Negotiation of operational definition of terms

binding Conversion of expressions and terms into immediately operational or evaluatable forms. Evaluation produces singular output in a specified range.

theory A system of axioms – atleast in the sense here

formally complete All well-formed sentences – or their negation – can be proved

formally consistent A well-formed sentence with a proof does not also have a proof for its opposite.

those things you are referring to The term 'reference' or 'refers to' has a much narrower and stricter definition in logic. 'The things that you are referring to' is found through a combination of reference as well as extension.

see also *extension, reference*

time Time as an index ordering is naïve, but often serviceable for many analysis

See also temporal logic

transformational - generative grammar Turing-complete but there is very little linguistic or cognitive significance to machine operations or structure. Much of the work becomes equivalent to 'coding' and 'debugging' issues.

truth Roughly five theories: correspondence theory, coherence theory, pragmatics, radical-interpretation and verificationist. The verificiationist serves as decidability, and correspondence is often used to determine true or false.

see also *observability*

degrees of — Please pretend this doesn't exist! Logic completely fails!

coherence theory — A statement is true iff it coheres with other statements we hold to be true. For example, "All X is Y" might found to be true if X is defined (as special type of Y) using a Genus & Species method of definition. Coherence is subordinate to empirical (verificationist) findings. If "All X is Y" is found to be empirically false, then the Coherence theory (the statements we've accepted, specifically the aforementioned Genus & Species definition).

see also *correspondence, hierarchic definitions, verificationist*

correspondence theory — The true statements are those that correspond to facts. Most logical analysis is limited to very narrow and regimented languages; its utility has been challenged on these grounds.

see also *World Model*

pragmatics — The statements work out well in the long run

radical interpretation — A statement is interpretable only if the listener has a great deal in common with the speaker, and the listener's language has a great deal in common with the speaker's.

Donald Davidson 'Radical Interpretation', Dialectica, 27 (1973), p314-28. Reprinted in Inquiries into Truth and Interpretation (1984) p125-39

verificationist — A formula is decidable – and of possible significance – only if one knows how to verify it, such as how to observe it. Very few propositions and topics of interest can be observed or otherwise verified in such a manner. Once verified it is known as true or false.

see also *observability*

truth theoretic Focused on the construction if statements

undecidable There are no contradictions if the assertion is treated as true, and there are no contradictions if the assertion is treated as false. That is, *proof by contradiction* does not work.

See also *decidability*

unification Unification is a key step in the resolution method, operating like regular expression matching. Unification operates on a *substitution* table (see the example below) adding further entries as it binds variables. Unification takes this table, a goal clause, and a clause in the table. It tries every combination of variable assignments to make the two clauses equivalent. It steps thru the both clauses in the same way:

1. If this element is a *free* variable, *bind* it to the corresponding element in the other clause. This is done by adding an entry into the substitution table.
2. If this element is a bound variable, look up its value; if it is a *literal*, use that. Perform the same on the other side. If the two values are defined, but do not match, abort; unification cannot be performed.
3. If the element has *parameter* or sub-ordinate elements, a unification step is performed on those parameter clauses of both main clauses.

This process repeats until no more items are added to the table.

This process effects the inference of variables values (or sets of acceptable values). It can link variables together, showing those that alias each other. It can be modified to remove possibilities from a potential set.

Term Rewriting systems perform a string substitution, replacing each occurrence of a variable with its bound value.

It is easy to understand the substitution table in cases where a variable can be bound to a simple value (e.g. a scalar or a string), a structure whose elements are found. What makes unification powerful is the ability it for a variable to be bound to another variable – v_4 (in this context) will inherit whatever v_1 is bound to. A variable can also be bound to a structure, whose elements might not be bound, or might be bound to another variable.

One drawback is that the table can have cycles. An *occurs check* operation can be attempted to catch this occurrence, but the check is very expensive.

see also *resolution principle, tableau*

Table 30: *Example substitution table*

Variable	*Binding*
$\mathbf{v_1}$	1
$\mathbf{v_2}$	"bob"
$\mathbf{v_3}$	house(red)
$\mathbf{v_4}$	v_1
$\mathbf{v_5}$	house(v_2)

universal language Can express logical statements, extra-linguistic statements, and statements about meaning and truth in the language.

universe of discourse Everything we talk about. Often this is rigidly (and artificially) limited with a closed-world assumption.

unrelated

1. There is no valid combination (or chain) of relations that allow two (or more) items of that kind or type to be related.
2. If we are talking about two specific items, there are some relationships between the kinds but all deny that the two specific items are related.

valuation function In theories constructed as a *model*, one needs to know how names and terms refer to entities and their properties, and how to evaluate sentences. For example Sally's height & mass, or an electrons charge. This is called a 'valuation function' although it is seldom a simple function, and often better understood as a procedure. This valuation assigns value for formula based on those references and how they combine (composition), table of forms and their values (idiomatic).

see also e*valuation procedure*

method1 One method is to use the problems declarative specification to specify a grammar and a family of automatons. The first automaton is special in that the sentences it recognizes (accepts) are also solutions to the problem. The other, optional, automatons generate fragments of the language that may be present in the acceptable sentence(s). Despite the unusual pretense of the solution as a sentence in an imaginary language, this technique can be very efficient.

see also *Chomsky hierarchy, language fragment*

vague Quantitative ambiguity (e.g. insufficiently precise term), task-related ambiguity (needs a plausible principle to resolve the question)

variable bound Value of the variable is controlled by a quantifier, is a parameter or is a constant

free A variable that is not a constant, not a parameter, and is not controlled by a quantifier

verificationist See *Truth*

WalkSAT

```
for(I=1; I < Max Tries; I++)
{
  solution = random truth assignment
  for (J=1; J < MaxFlips; J++)
  {
    if all clauses satisfied clause then return solution
    c ← random unsatisfied clause
    with probability p
      flip a random variable in c
     else
      flip variable in c that maximizes the number of satisfied claims
  }
}
return failure
```

max WalkSAT

```
for(I=1; I < Max Tries; I++)
{
  solution = random truth assignment
  for (J=1; J < MaxFlips; J++)
  {
    m = sum of weights(sat clauses)
    if m > threshold then return solution
    c ← random unsatisfied clause
    with probability p
      flip a random variable in c
     else
      flip variable in c that maximizes m
  }
}
return failure with best solution found
```

a version without memory explosion is at http://alchemy.cs.washington.edu

well-formed formula A formula that has all variables bound. A part of the syntax of the algebra's language. Much of discussion mixes between describing the syntax and much more complex issue of meaning (satisfaction)

well-posed problem The information is clearly specified. We can determine when the problem has been solved. The problem does not change during its attempted solution.

witness function A function that 'testifies' a proposition is highly likely to be true.

see also *probability estimator*

world model A fact base about the world, and operations used in reasoning about the world. Most often needs to include a modal logic to support the history. Need to incorporate different times in a possible world and branching at times.

- The set of all possible words of category s;
- A function that generates all the words of category S
- The set of word types
- A map of the words of category S onto the set of word types.