Adapted from AIMA slides

From Propositional Logic to First-Order Logic

Peter Antal
antal@mit.bme.hu

Outline

- Why FOL?
- Syntax and semantics of FOL
- Knowledge engineering in FOL
- Inference in FOL
  - Reducing first-order inference to propositional inference
  - Unification
  - Generalized Modus Ponens
    - Forward chaining
    - Backward chaining
  - Resolution

Pros and cons for propositional logic

- ☺ Propositional logic is declarative.
- ☺ Propositional logic is compositional:

  meaning of $B_{1,1} \wedge P_{1,2}$ is derived from meaning of $B_{1,1}$ and of $P_{1,2}$
- ☺ Propositional logic allows partial/disjunctive/negated information

  (unlike most data structures and databases)
- ☺ Meaning in propositional logic is context-independent.

  (unlike natural language, where meaning depends on context)
- ☹ Propositional logic has very limited expressive power
  - E.g., cannot say "pits cause breezes in adjacent squares“
    - except by writing one sentence for each square

A classification of animals

- Those that belong to the emperor
- Embalmed ones
- Those that are trained
- Suckling pigs
- Mermaids (or Sirens)
- Fabulous ones
- Stray dogs
- Those that are included in this classification
- Those that tremble as if they were mad
- Innumerable ones
- Those drawn with a very fine camel hair brush
- *Et cetera*
- Those that have just broken the flower vase
- Those that, at a distance, resemble flies

***„Celestial Emporium of Benevolent Knowledge"* from** Borges, J.L., 1981. The analytical language of John Wilkins. *Borges: A reader*.

What is knowledge representation?

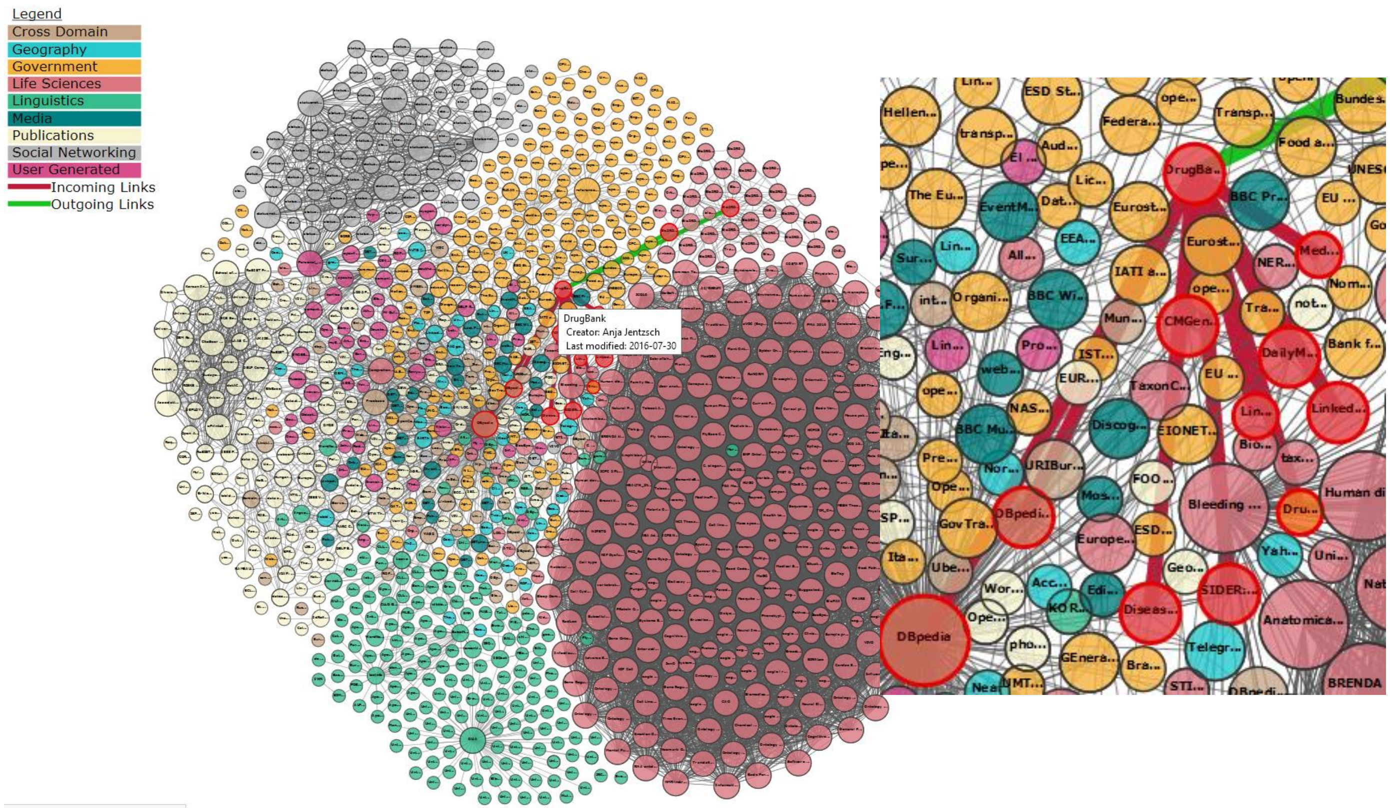
1. Surrogate
2. Set of ontological commitments
3. Theory of intelligent reasoning
4. Medium for efficient computation
5. Medium of human expression

Davis, R., Shrobe, H. and Szolovits, P., 1993. What is a knowledge representation?. *AI magazine*, *14*(1), p.17.

Network of databases: 2000

- 10k< relevant biological databases and knowledge-bases
- Petabytes of sequence and high-throughput gene/protein data
- ~10.000.000 concepts and relations explicitly in knowledge bases

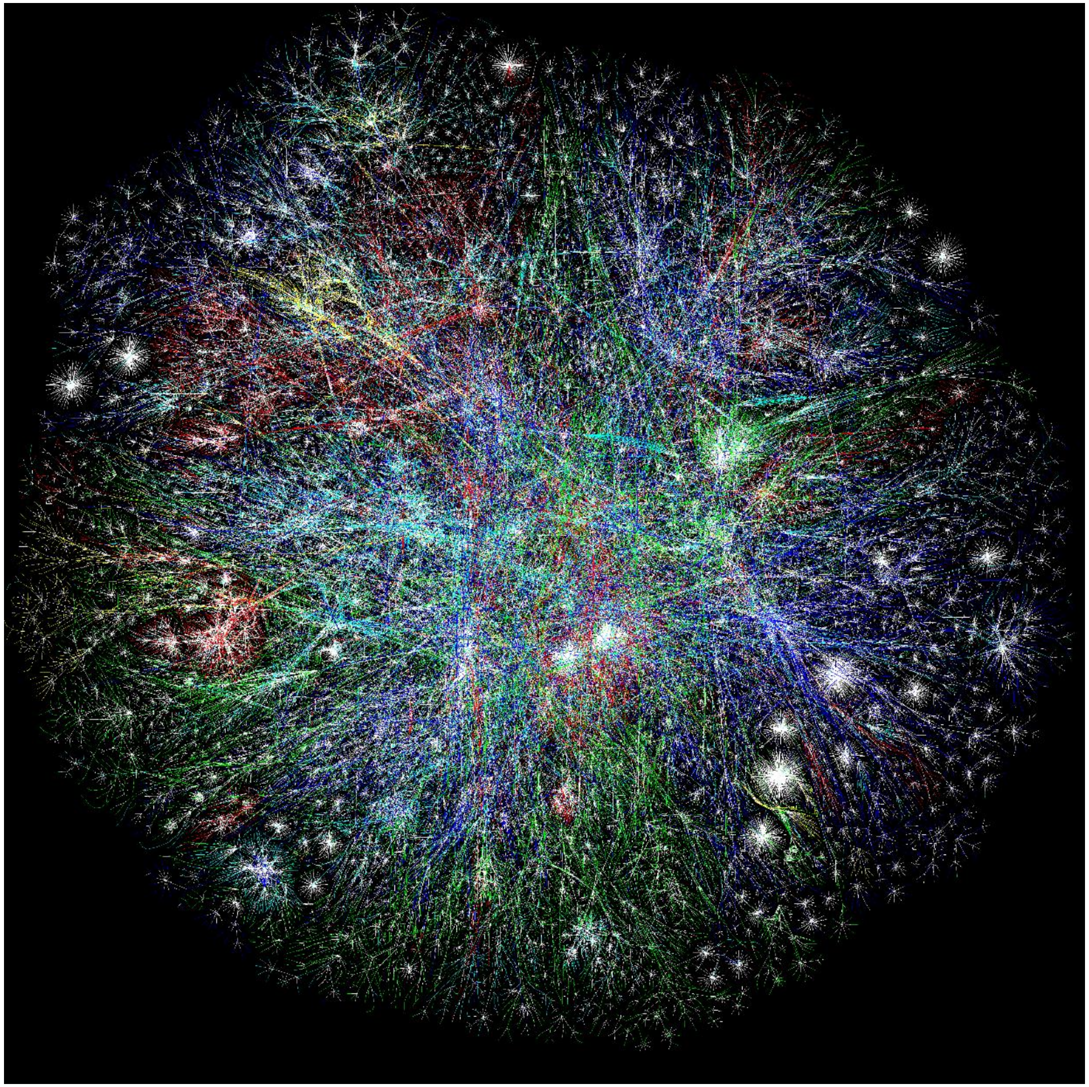
Linked Open Data: ~2020



Linking Open Data cloud diagram 2017, by Andrejs Abele, John P. McCrae, Paul Buitelaar, Anja Jentzsch and Richard Cyganiak. http://lod-cloud.net/

Semantic Web

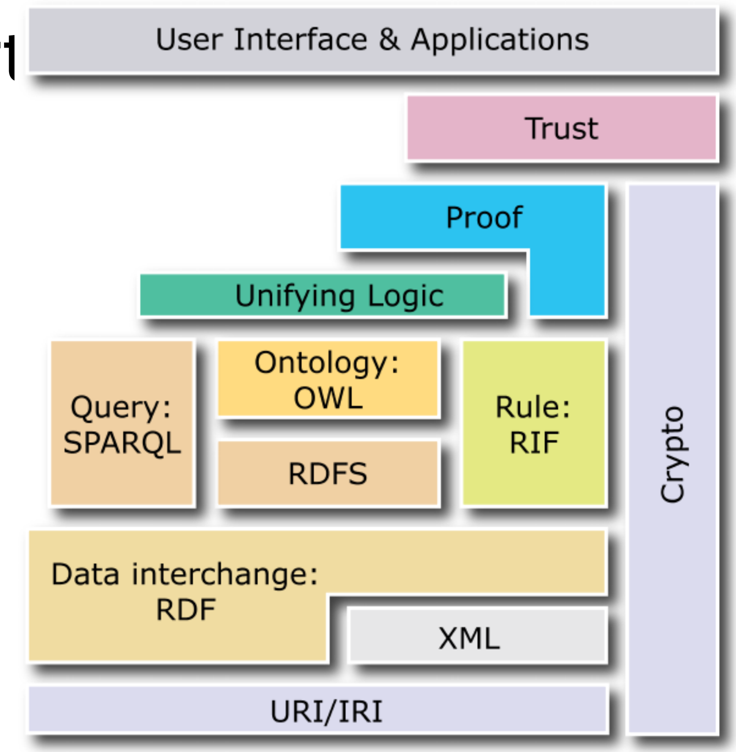
- Tim Berners-Lee, 1999, „I have a dream...”, W3C
- Web of data, Web 3.0
- Share, reuse, querying, integration of data, automatic processing, reasoning
- Publishing data in human readable HTML documents to machine readable documents
- Linked Data



The Internet network: nodes are computers or post-pc devices and links are wired or wireless connections between them.
https://users.dimi.uniud.it/~massimo.franceschet/netart/talk/netart.html

Semantic Web Architecture

- URI/IRI
- RDF
- Formats eg. RDF/XML, Turt
- RDF Schema, OWL
- SPARQL
- RIF
- ...



A Resource Description Framework (RDF) háttér

- The data model of the Semantic Web
- RDF statement
  - subject: resource identified by an IRI
  - predicate (property): resource identified by an IRI
  - object: resource or literal (constant value)
- Graph databases of RDF triples

Relational databases vs. Triplestores (graph databases)

Relational databases

- Relations are **separated** from data (cases)
- Tables&keys define the formal model (syntax) for the data (cases)
- ➔**Model-based** (~predefined)
- Meaning (semantics) is informal (out of scope of the DB)
- **Singular databases** (~they are separated)

Triplestores

- **Unified** representation of relations and data
- Triples („graph database”) stores the dynamic model for the data, together with the factual data
- ➔**Model-free** (~relations as data)
- Meaning is defined by the (explicit) relations (~**ontology**)
- **Linked open data space** (using universal identifiers & ontologies)

Model	Example Format	Data	Metadata	Identifier	Query Syntax	Semantics (Meaning)
Relational	MS SQL, Oracle, MySQL	Table Cell Values	Table Column Definitions	Primary Key (Data Column) Value	SQL	N/A
Hierarchical	XML	Tag/Attribute Values	XSD/DTD	e.g. Unique Attribute Key Value	XPath	N/A
Graph	RDF/XML, Turtle	RDF	RDFS/OWL	URI	SPARQL	Yes, using RDFS and OWL

Cf. Neumann’s principle: instructions is data

SPARQL

- a query language specification for querying over RDF triples

```
# prefix declarations
PREFIX foo: <http://example.com/resources/>
...
# dataset definition
FROM ...
# result clause
SELECT ...
# query pattern
WHERE {
    ...
}
# query modifiers
ORDER BY ...
```

Semantic technologies for drug discovery

- Whitaker, B.J. and Rzepa, H.S., 1995. Chemical publishing via the Internet. In *International chemical information conference* (pp. 62-71).
- Murray-Rust, P., Rzepa, H.S., Wright, M. and Zara, S., 2000. A universal approach to web-based chemistry using XML and CML. *Chemical Communications*, (16), pp.1471-1472.
- Murray-Rust, P. and Rzepa, H.S., 2002. Scientific publications in XML-towards a global knowledge base. *Data Science Journal*, *1*, pp.84-98.
- Murray-Rust, P., 2008. Chemistry for everyone. *Nature*, *451*(7179), pp.648-651.

BIO2RDF

Linked Data for the Life Sciences

http://bio2rdf.org/

1. Belleau, F., Nolin, M.A., Tourigny, N., Rigault, P. and Morissette, J., 2008. **Bio2RDF: towards a mashup to build bioinformatics knowledge systems**. *Journal of biomedical informatics*, *41*(5), pp.706-716.
2. Dumontier, M., Callahan, A., Cruz-Toledo, J., Ansell, P., Emonet, V., Belleau, F. and Droit, A., 2014, October. **Bio2RDF release 3: a larger connected network of linked data for the life sciences**. In *Proceedings of the 2014 International Conference on Posters & Demonstrations Track-Volume 1272* (pp. 401-404). CEUR-WS. org.

Databases:..

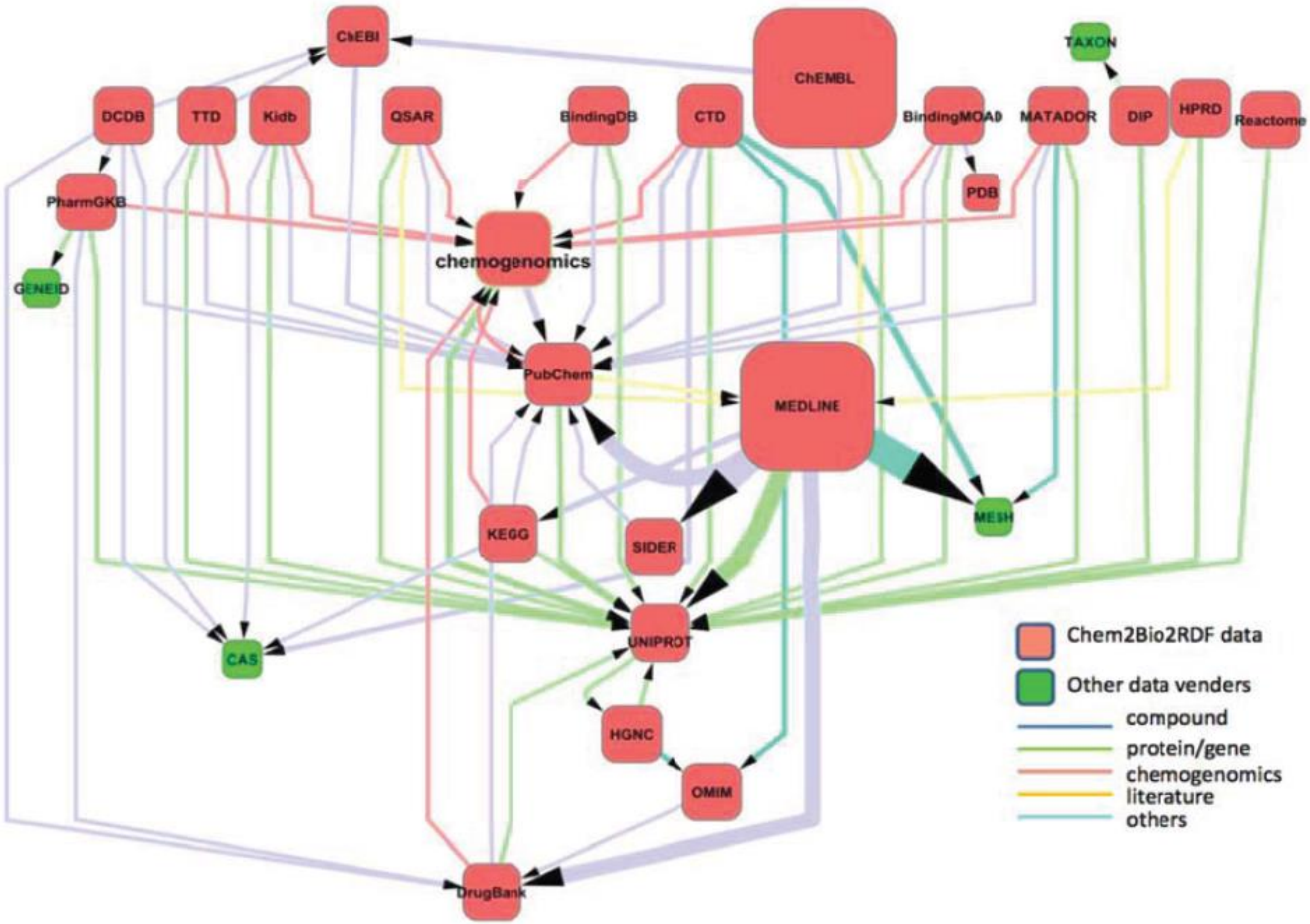
Search:

		# of triples	# of unique entities
		11895348562	1107871027
chEMBL [chembl] ChEMBL is a database of bioactive compounds, their quantitative properties and bioactivities (binding constants, pharmacology and ADMET, etc). The data is abstracted and curated from the primary scientific literature. https://www.ebi.ac.uk/chembldb/ **Links:** search query example download		409942525	50061452
ClinicalTrials.gov [clinicaltrials] ClinicalTrials.gov is a registry and results database of publicly and privately supported clinical studies of human participants conducted around the world. http://clinicaltrials.gov/ **Links:** search query example download	2014-09-25	98835804	7337123
Comparative Toxicogenomics Database [ctd] CTD includes manually curated data describing cross-species chemical-gene/protein interactions and chemical- and gene-disease relationships to illuminate molecular mechanisms underlying variable susceptibility and environmentally influenced diseases. http://ctd.mdibl.org **Links:** search query example download	2014-06-09	326720894	19768641
Database of single nucleotide polymorphism [dbsnp] The dbSNP database is a repository for both single base nucleotide subsitutions and short deletion and insertion polymorphisms. http://www.ncbi.nlm.nih.gov/SNP/ **Links:** search query example download	2014-07-15	8801487	530538
DrugBank [drugbank] The DrugBank database is a bioinformatics and chemoinformatics resource that combines detailed drug (i.e. chemical, pharmacological and pharmaceutical) data with comprehensive drug target (i.e. sequence, structure, and pathway) information. http://www.drugbank.ca/ **Links:** search query example download	2014-07-25	3672531	316950
GenAge: The Ageing Gene Database [genage] GenAge is a database of human and model organism genes related to longevity and aging, maintained by the Human Ageing Genomics Resources (HAGR) group. http://genomics.senescence.info/genes/ **Links:** search query example download	2014-06-03	73048	6995
GenDR: The Dietary Restriction Gene Database [gendr] GenDR is a database of genes associated with dietary restriction (DR). GenDR includes two datasets: 1) genes inferred from experiments in model organisms in which genetic manipulations cancel out or disrupt the life-extending effects of DR; 2) genes robustly altered due to DR, derived from a meta-analysis of microarray DR studies in mammals http://genomics.senescence.info/diet/ **Links:** search query example download	2014-06-03	11663	1129
Gene Ontology Annotation [goa] The GOA (Gene Ontology Annotation) project provides high-quality Gene Ontology (GO) annotations to proteins in the UniProt Knowledgebase (UniProtKB) and International Protein Index (IPI). This involves electronic annotation and the integration of high-quality manual GO annotation from all GO Consortium model organism groups and specialist groups. http://www.ebi.ac.uk/GOA **Links:** search query example download	2014-06-05	97520151	5950074
HUGO Gene Nomenclature Committee [hgnc] The HGNC gives unique and meaningful names to every human gene. For each known human gene we approve a gene name and symbol (short-form abbreviation). All approved symbols are stored in the HGNC database. http://www.genenames.org/ **Links:** search query example download	2014-07-04	3628205	372136

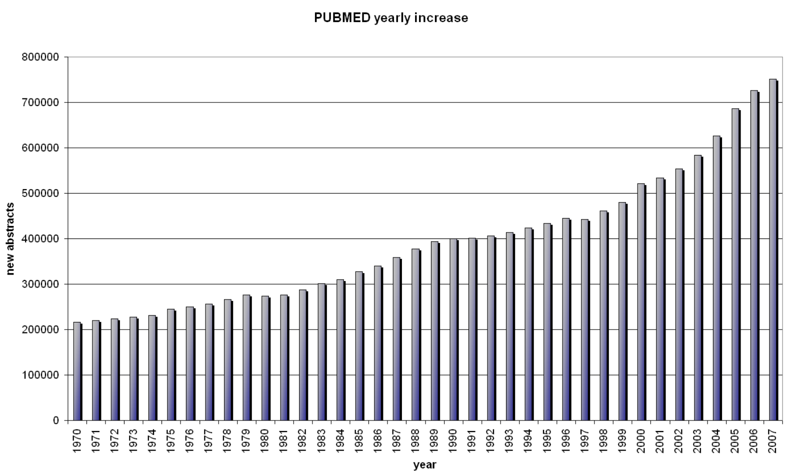
Chem2Bio2RDF I.

Data Source	RDF Resource Name	# of RDF Triples
PubChem Compound	compound	233,852
PubChem BioAssay	pubchem bioassay	1,715,247
ChEBI	chebi	2,237,330
KEGG	kegg_ligand	96,000
KEGG	kegg_interaction	70,029
KEGG	kegg_pathway_protein	84,760
CTD	ctd_interaction	2,443,826
CTD	ctd_chem_disease	2,025,513
BindingDB	bindingdb_ligand	223,818
BindingDB	bindingdb_interaction	800,016
PharmGKB	pharmgkb_drugs	14,760
PharmGKB	pharmgkb_genes	340,808
PharmGKB	pharmgkb_rela ions	73,276
DrugBank	drugbank_drug	47,640
DrugBank	drugbank _interaction	111,001
UniProt	uniprot	34,951
HPRD	hprd	408,177
Reactome	reactome	21,985
DIP	dip	1,113,840
OMIM	omim	23,432
SIDER	sider	305,510
PubMed	pubmed2compound	269,178

Chem2Bio2RDF II.



Semantic publishing: papers vs DBs/KBs



M. Gerstein, "E-publishing on the Web: Promises, pitfalls, and payoffs for bioinformatics," *Bioinformatics,* 1999

M. Gerstein: **Blurring the boundaries between scientific 'papers' and biological databases**, Nature, 2001

P. Bourne, "Will a biological database be different from a biological journal?," *Plos Computational Biology,* 2005

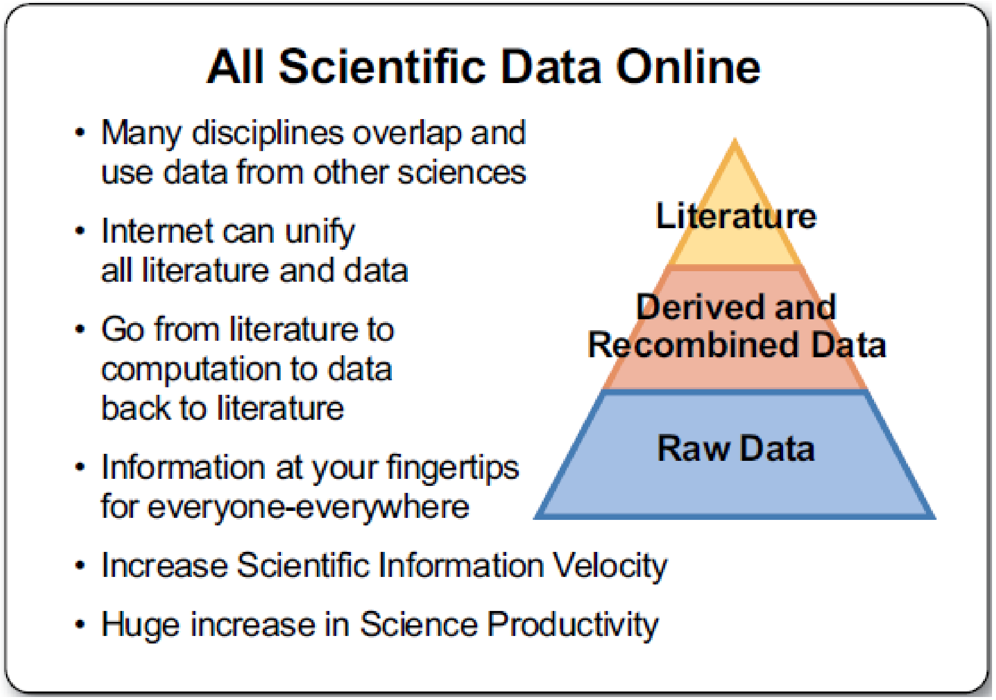
M. Gerstein et al: "Structured digital abstract makes text mining easy," *Nature,* 2007.

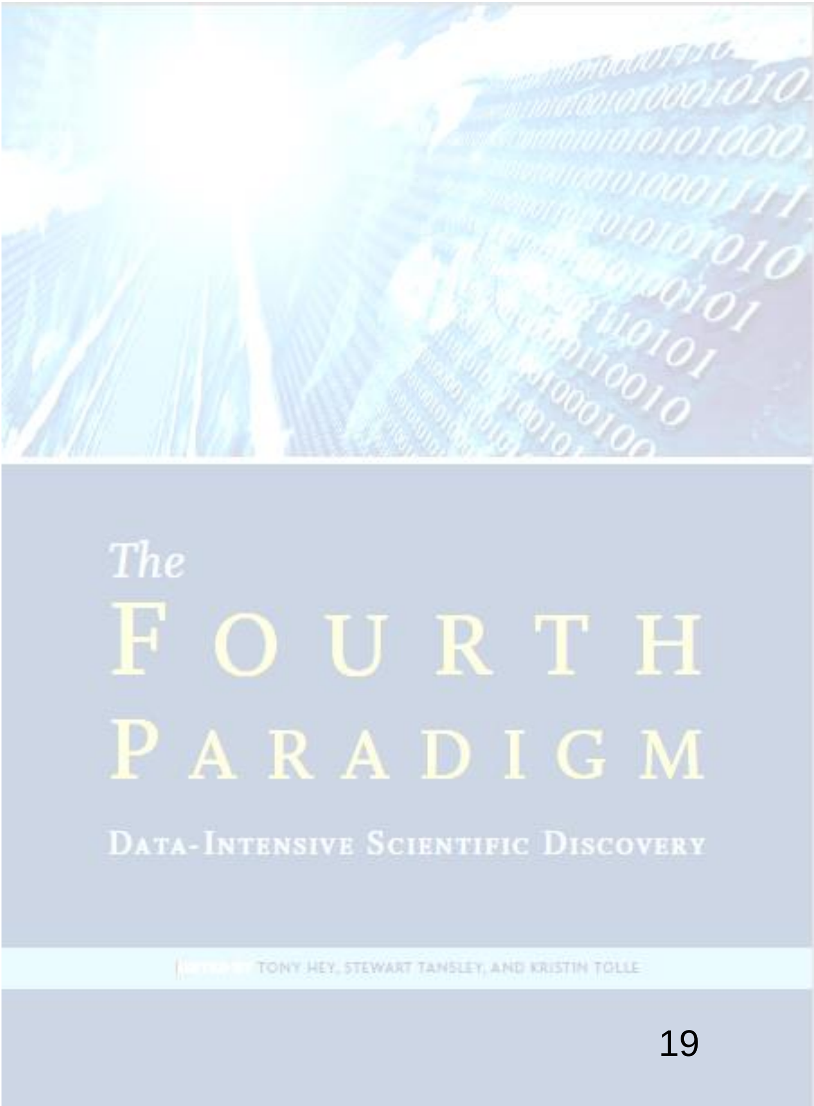
M. Seringhaus et al: "Publishing perishing? Towards tomorrow's information architecture," *Bmc Bioinformatics,* 2007.

M. Seringhaus: "Manually structured digital abstracts: A scaffold for automatic text mining," *Febs Letters,* 2008.

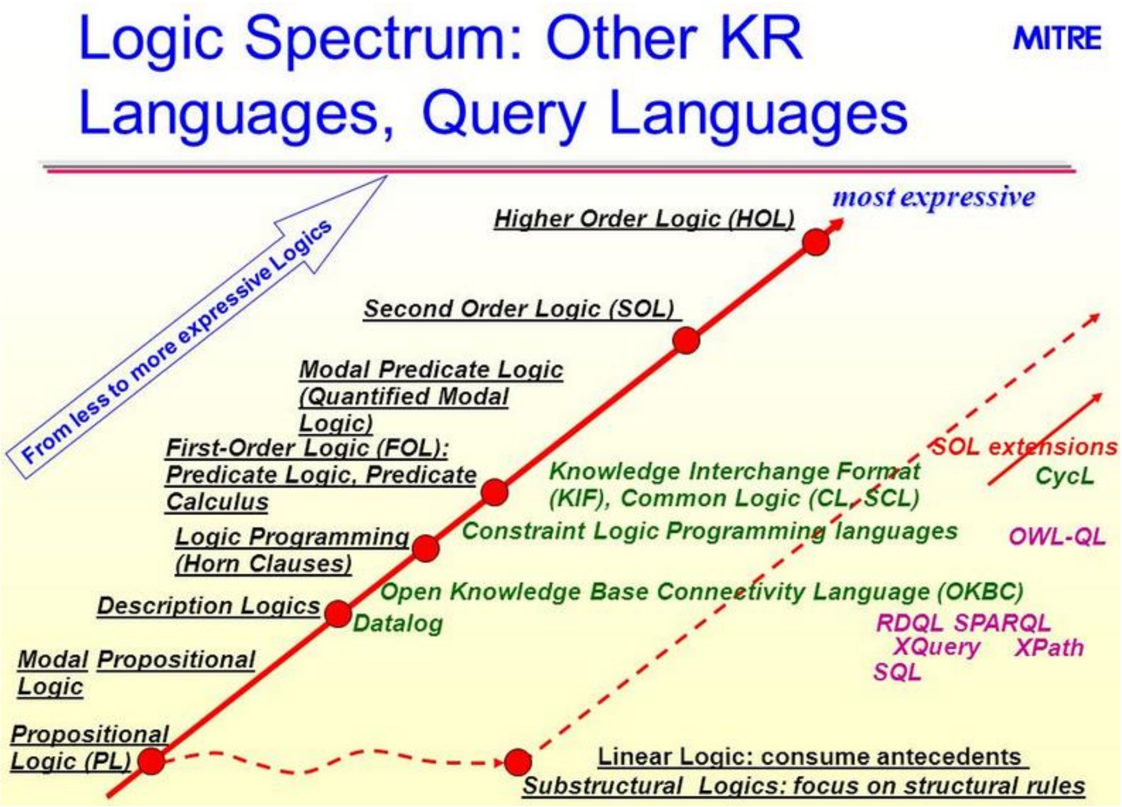
D. Shotton: "Semantic publishing: the coming revolution in scientific journal publishing," *Learned Publishing,* 2009

E-science, data-intensive science





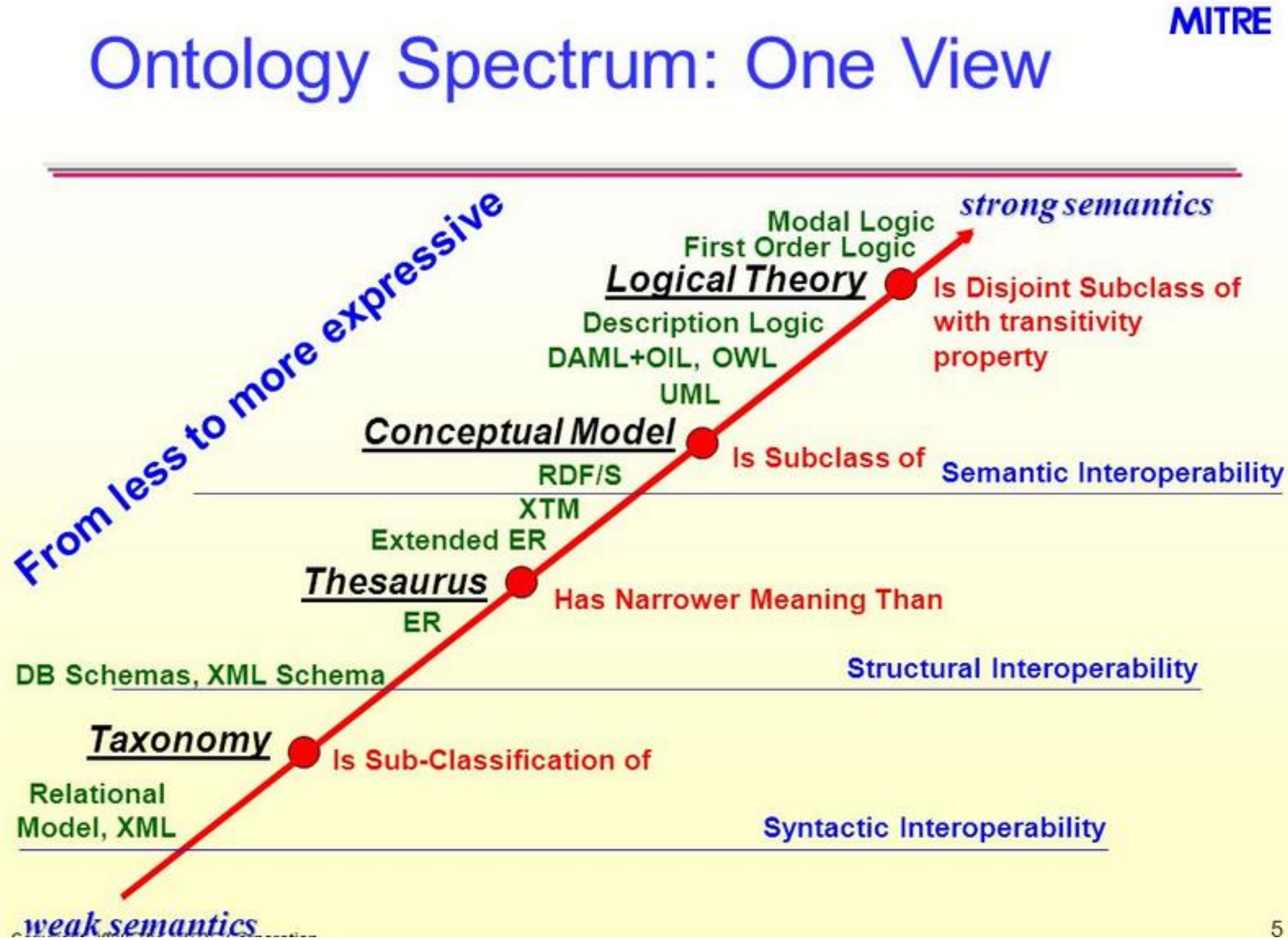
The spectrum of logics



http://slideplayer.com/slide/697642/

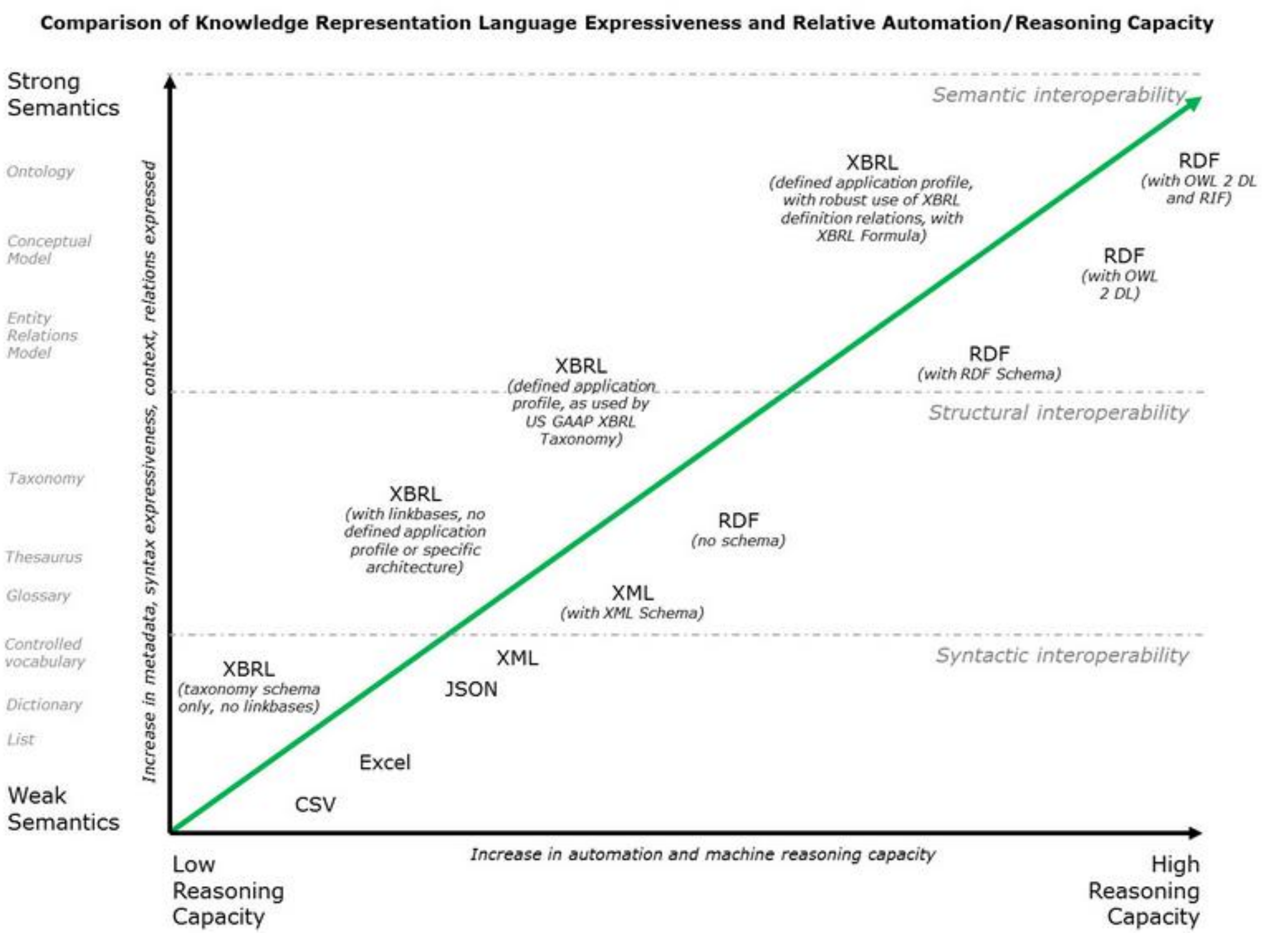
For Complexity of reasoning in Description Logics, see e.g.:
http://www.cs.man.ac.uk/~ezolin/dl/
For Reasoners in DL:
http://owl.cs.manchester.ac.uk/tools/list-of-reasoners/

The spectrum of logics II.



http://slideplayer.com/slide/697642/

Knowledge representations for financial reporting



http://xbrl.squarespace.com/

First-order logic

- Whereas propositional logic assumes the world contains facts,
- first-order logic (like natural language) assumes the world contains
  - Objects: people, houses, numbers, colors, baseball games, wars, …
  - Relations: red, round, prime, brother of, bigger than, part of, comes between, …
  - Functions: father of, best friend, one more than, plus, …

Syntax of FOL: Basic elements

Constants	KingJohn, 2,...
Predicates	Brother, $>$,...
Functions	Sqrt, LeftLegOf,...
Variables	x, y, a, b,...
Connectives	$\neg$, $\Rightarrow$, $\wedge$, $\vee$, $\Leftrightarrow$
Equality	$=$
Quantifiers	$\forall$, $\exists$

Atomic sentences

Atomic sentence = *predicate* $(term_1,\ldots,term_n)$
or $term_1 = term_2$

Term = *function* $(term_1,\ldots,term_n)$
or *constant* or *variable*

- E.g., *Brother(KingJohn,RichardTheLionheart)* > *(Length(LeftLegOf(Richard)), Length(LeftLegOf(KingJohn)))*

Complex sentences

- Complex sentences are made from atomic sentences using connectives
-

$$\neg S,\ S_1 \wedge S_2,\ S_1 \vee S_2,\ S_1 \Rightarrow S_2,\ S_1 \Leftrightarrow S_2,$$

E.g. *Sibling(KingJohn,Richard)* $\Rightarrow$ *Sibling(Richard,KingJohn)*

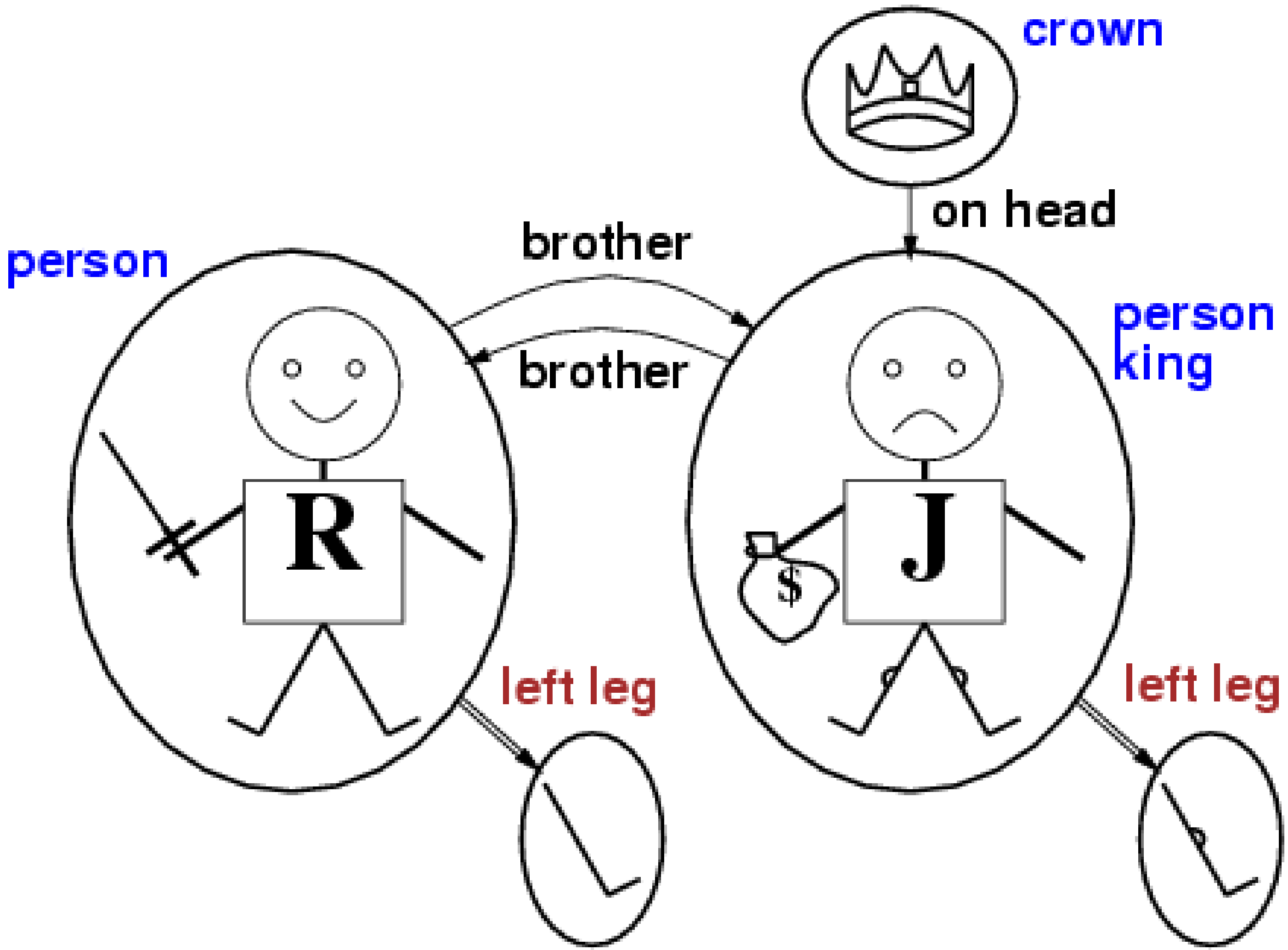
$>(1,2) \vee \leq (1,2)$

$>(1,2) \wedge \neg >(1,2)$

Truth in first-order logic

- Sentences are true with respect to a model and an interpretation
- Model contains objects (domain elements) and relations among them
-
- Interpretation specifies referents for
  constant symbols → objects
  predicate symbols → relations
  function symbols → functional relations
- An atomic sentence *predicate*$(term_1,\ldots,term_n)$ is true iff the objects referred to by $term_1,\ldots,term_n$ are in the relation referred to by *predicate*

Models for FOL: Example



Universal quantification

- ∀<*variables*> <*sentence*>
-

Everyone at Y is smart:
$\forall x \; At(x,Y) \Rightarrow Smart(x)$

- $\forall x \; P$ is true in a model *m* iff *P* is true with *x* being each possible object in the model
-
- Roughly speaking, equivalent to the conjunction of instantiations of *P*
- $At(KingJohn,NUS) \Rightarrow Smart(KingJohn)$
  $\wedge \; At(Richard,NUS) \Rightarrow Smart(Richard)$
  $\wedge \; At(NUS,NUS) \Rightarrow Smart(NUS)$
  $\wedge \; \ldots$

A common mistake to avoid

- Typically, $\Rightarrow$ is the main connective with $\forall$
-
- Common mistake: using $\wedge$ as the main connective with $\forall$:

  $\forall x\ At(x,Y) \wedge Smart(x)$

  means "Everyone is at Y and everyone is smart"

Existential quantification

- $\exists$<*variables*> <*sentence*>

- Someone at Y is smart:
- $\exists x$ At(x,Y) $\wedge$ Smart(x)$
-
- $\exists x$ *P* is true in a model *m* iff *P* is true with *x* being some possible object in the model
-
- Roughly speaking, equivalent to the disjunction of instantiations of *P*
- At(KingJohn,NUS) $\wedge$ Smart(KingJohn)
  $\vee$ At(Richard,NUS) $\wedge$ Smart(Richard)
  $\vee$ At(NUS,NUS) $\wedge$ Smart(NUS)
  $\vee$ ...

Another common mistake to avoid

- Typically, $\wedge$ is the main connective with $\exists$
- Common mistake: using $\Rightarrow$ as the main connective with $\exists$:
-

$$\exists x\ At(x,Y) \Rightarrow Smart(x)$$

is true if there is anyone who is not at Y!

Properties of quantifiers

- $\forall x\ \forall y$ is the same as $\forall y\ \forall x$
- $\exists x\ \exists y$ is the same as $\exists y\ \exists x$
- $\exists x\ \forall y$ is not the same as $\forall y\ \exists x$
- $\exists x\ \forall y\ Loves(x,y)$
  - "There is a person who loves everyone in the world"
- $\forall y\ \exists x\ Loves(x,y)$
  - "Everyone in the world is loved by at least one person"
- Quantifier duality: each can be expressed using the other
- $\forall x\ Likes(x,IceCream)\ \neg\exists x\ \neg Likes(x,IceCream)$
- $\exists x\ Likes(x,Broccoli)\qquad \neg\forall x\ \neg Likes(x,Broccoli)$

Equality

- $term_1 = term_2$ is true under a given interpretation if and only if $term_1$ and $term_2$ refer to the same object
-
- E.g., definition of *Sibling* in terms of *Parent*:
-

  $\forall x,y\ Sibling(x,y) \Leftrightarrow [\neg(x = y) \wedge \exists m,f\ \neg (m = f) \wedge Parent(m,x) \wedge Parent(f,x) \wedge Parent(m,y) \wedge Parent(f,y)]$

Using FOL

The kinship domain:

- Brothers are siblings
- $\forall x,y\ Brother(x,y) \Leftrightarrow Sibling(x,y)$
- One's mother is one's female parent
- $\forall m,c\ Mother(c) = m \Leftrightarrow (Female(m) \wedge Parent(m,c))$
- "Sibling" is symmetric
- $\forall x,y\ Sibling(x,y) \Leftrightarrow Sibling(y,x)$

Knowledge engineering in FOL

1. Identify the task.
2. Assemble the relevant knowledge.
3. Decide on a vocabulary of predicates, functions, and constants.
4. Encode general knowledge about the domain.
5. Encode a description of the specific problem instance.
6. Pose queries to the inference procedure and get answers.
7. Debug the knowledge base.

Inference in FOL

- Syllogisms
- Reducing first-order inference to propositional inference
- Unification
- Generalized Modus Ponens
- Forward chaining
- Backward chaining
- Resolution

Syllogisms of the First Figure			
	BARBARA		**CELARENT**
A A A	Every B is A. Every C is B. Therefore, every C is A.	E A E	No B is A. Every C is B. Therefore, no C is A.
	DARII		**FERIO**
A I I	Every B is A. Some C is B. Therefore, some C is A.	E I O	No B is A. Some C is B. Therefore, some C is not A.

A universal Affirmative $\forall x.\ B(x) \rightarrow A(x)$
E universal nEgative $\forall x.\ B(x) \rightarrow \neg A(x)$
I partIcular affirmative $\exists x.\ C(x) \wedge B(x)$
O particular nOt affirmative (negative) $\exists x.\ C(x) \wedge \neg B(x)$

BARBARA:

$$\frac{\forall x.\ B(x) \rightarrow A(x) \qquad \forall x.\ C(x) \rightarrow B(x)}{\forall x.\ C(x) \rightarrow A(x)}$$

Syllogisms

BARBARA:

$\forall x.\ B(x) \rightarrow A(x)$
$\underline{\forall x.\ C(x) \rightarrow B(x)}$
$\forall x.\ C(x) \rightarrow A(x)$

DARII:

$\forall x.\ B(x) \rightarrow A(x)$
$\underline{\exists x.\ C(x) \wedge B(x)}$
$\exists x.\ C(x) \wedge A(x)$

CELARENT:

$\forall x.\ B(x) \rightarrow \neg A(x)$
$\underline{\forall x.\ C(x) \rightarrow B(x)}$
$\forall x.\ C(x) \rightarrow \neg A(x)$

FERIO:

$\forall x.\ B(x) \rightarrow \neg A(x)$
$\underline{\exists x.\ C(x) \wedge B(x)}$
$\exists x.\ C(x) \wedge \neg A(x)$

Fig. I.

CESARE:

$\forall x.\ B(x) \rightarrow \neg A(x)$
$\underline{\forall x.\ C(x) \rightarrow A(x)}$
$\forall x.\ C(x) \rightarrow \neg B(x)$

CAMESTRES:

$\forall x.\ B(x) \rightarrow A(x)$
$\underline{\forall x.\ C(x) \rightarrow \neg A(x)}$
$\forall x.\ C(x) \rightarrow \neg B(x)$

FESTIMO:

$\forall x.\ B(x) \rightarrow \neg A(x)$
$\underline{\exists x.\ C(x) \wedge A(x)}$
$\exists x.\ C(x) \wedge \neg B(x)$

BAROCO:

$\forall x.\ B(x) \rightarrow A(x)$
$\underline{\exists x.\ C(x) \wedge \neg A(x)}$
$\exists x.\ C(x) \wedge \neg B(x)$

Fig. II.

érvelési hiba

DARAPTI:

$\forall x.\ C(x) \rightarrow A(x)$
$\underline{\forall x.\ C(x) \rightarrow B(x)}$
$\exists x.\ B(x) \wedge A(x)$

FELAPTON:

$\forall x.\ C(x) \rightarrow \neg A(x)$
$\underline{\forall x.\ C(x) \rightarrow B(x)}$
$\exists x.\ B(x) \wedge \neg A(x)$

DISAMIS:

$\exists x.\ C(x) \wedge A(x)$
$\underline{\forall x.\ C(x) \rightarrow B(x)}$
$\exists x.\ B(x) \wedge A(x)$

DATISI:

$\forall x.\ C(x) \rightarrow A(x)$
$\underline{\exists x.\ C(x) \wedge B(x)}$
$\exists x.\ B(x) \wedge A(x)$

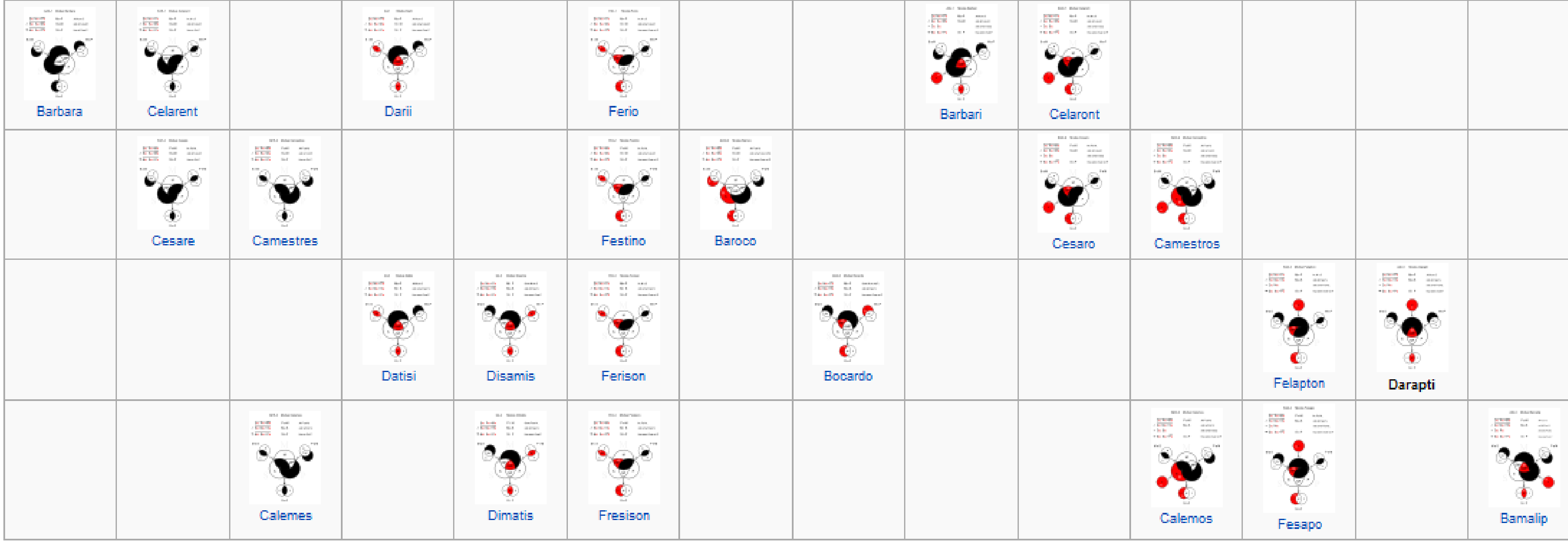
BOCARDO:

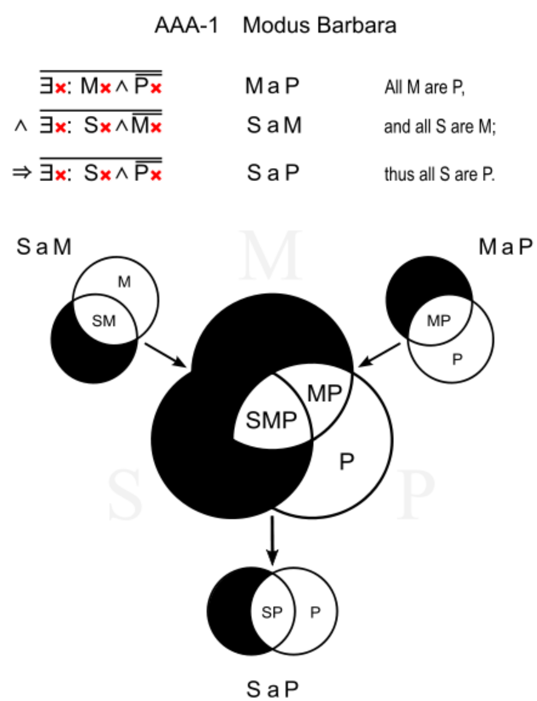
$\exists x.\ C(x) \wedge \neg A(x)$
$\underline{\forall x.\ C(x) \rightarrow B(x)}$
$\exists x.\ B(x) \wedge \neg A(x)$

Fig. III.

FERISON: $\forall x.\ C(x) \rightarrow \neg A(x)$
$\underline{\exists x.\ C(x) \wedge B(x)}$
$\exists x.\ B(x) \wedge \neg A(x)$

Fig. IV.


Barbara
Celarent
Darii
Ferio
Barbari
Celaront
Cesare
Camestres
Festino
Baroco
Cesaro
Camestros
Datisi
Disamis
Ferison
Bocardo
Felapton
Darapti
Calemes
Dimatis
Fresison
Calemos
Fesapo
Bamalip


AAA-1 Modus Barbara
∃x: Mx ∧ Px
M a P
All M are P,
∧ ∃x: Sx ∧ Mx
S a M
and all S are M;
⇒ ∃x: Sx ∧ Px
S a P
thus all S are P.
S a M
M a P
M
SM
M
MP
P
SMP
MP
P
S
P
SP
P
S a P

Reduction to propositional inference

Suppose the KB contains just the following:

$\forall$x King(x) $\wedge$ Greedy(x) $\Rightarrow$ Evil(x)
King(John)
Greedy(John)
Brother(Richard,John)

- Instantiating the universal sentence in all possible ways, we have:
  King(John) $\wedge$ Greedy(John) $\Rightarrow$ Evil(John)
  King(Richard) $\wedge$ Greedy(Richard) $\Rightarrow$ Evil(Richard)
  King(John)
  Greedy(John)
  Brother(Richard,John)
- The new KB is propositionalized: proposition symbols are
-

King(John), Greedy(John), Evil(John), King(Richard), etc.

Reduction contd.

- Every FOL KB can be propositionalized so as to preserve entailment
-
- (A ground sentence is entailed by new KB iff entailed by original KB)
-
- Idea: propositionalize KB and query, apply resolution, return result
-
- Problem: with function symbols, there are infinitely many ground terms,
  - e.g., *Father*(*Father*(*Father*(*John*)))
  -

Reduction contd.

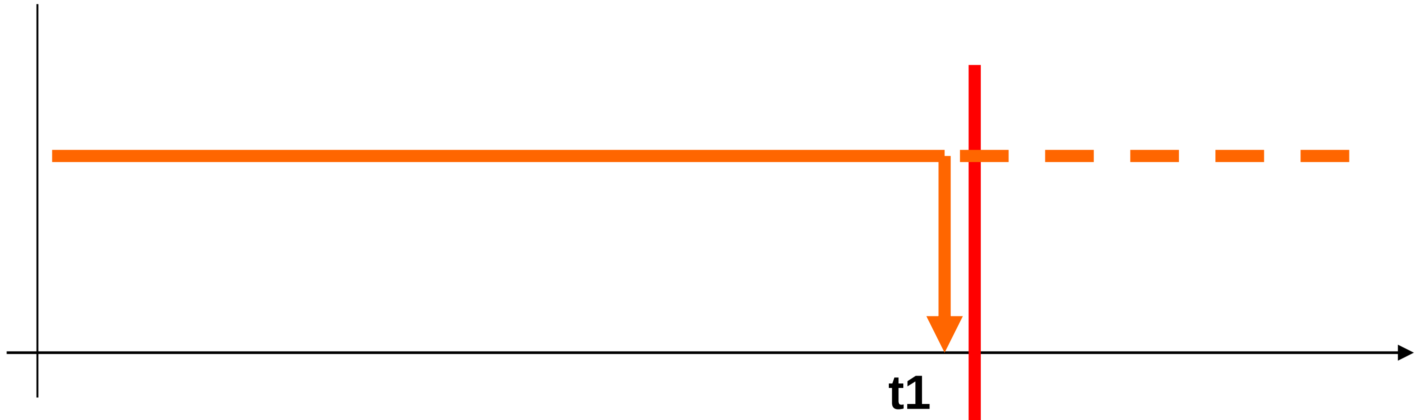
Theorem: Herbrand (1930). If a sentence α is entailed by an FOL KB, it is entailed by a finite subset of the propositionalized KB

Idea: For $n = 0$ to ∞ do
create a propositional KB by instantiating with depth-n terms
see if α is entailed by this KB

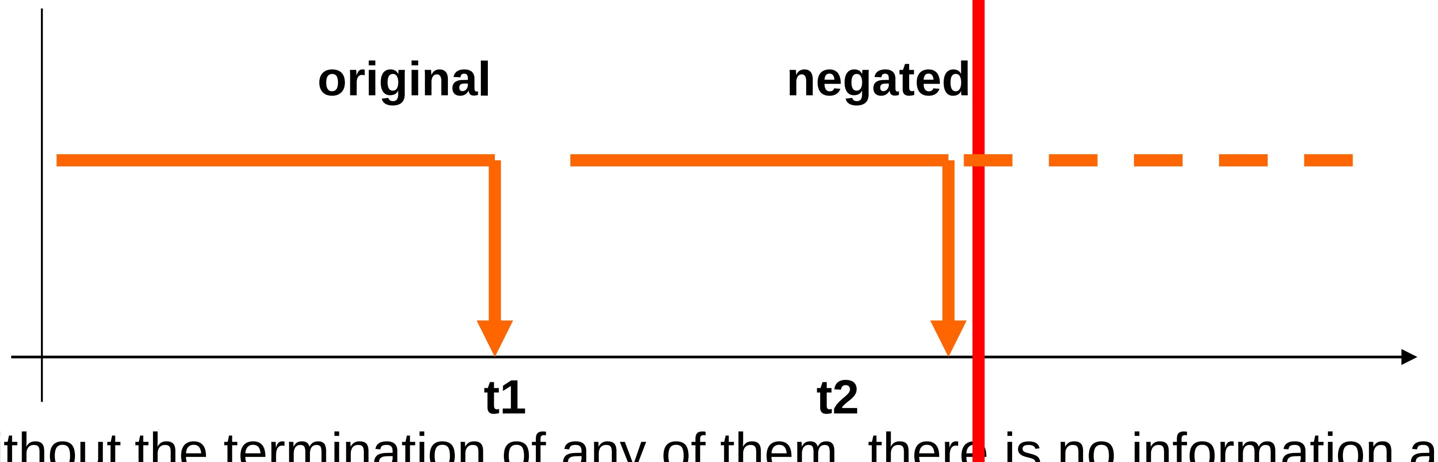
Problem: works if α is entailed, loops if α is not entailed

Theorem: Turing (1936), Church (1936) Entailment for FOL is semidecidable (algorithms exist that say yes to every entailed sentence, but no algorithm exists that also says no to every nonentailed sentence.)

Semidecidability in FOL: effect of finite time on proof



conclusion: if not proved, then false!?



Without the termination of any of them, there is no information about provability/truth.

Problems with propositionalization

- Propositionalization seems to generate lots of irrelevant sentences.
- E.g., from:
-
-

$\forall$x King(x) $\wedge$ Greedy(x) $\Rightarrow$ Evil(x)
King(John)
$\forall$y Greedy(y)
Brother(Richard,John)

- it seems obvious that *Evil*(*John*), but propositionalization produces lots of facts such as *Greedy*(*Richard*) that are irrelevant
-
- With *p k*-ary predicates and *n* constants, there are $p \cdot n^k$ instantiations.

Universal instantiation (UI)

- Every instantiation of a universally quantified sentence is entailed by it:

$$\frac{\forall v\ \alpha}{\text{Subst}(\{v/g\}, \alpha)}$$

for any variable v and ground term g

- E.g., $\forall x\ King(x) \wedge Greedy(x) \Rightarrow Evil(x)$ yields:

$King(John) \wedge Greedy(John) \Rightarrow Evil(John)$
$King(Richard) \wedge Greedy(Richard) \Rightarrow Evil(Richard)$
$King(Father(John)) \wedge Greedy(Father(John)) \Rightarrow Evil(Father(John))$
.
.
.

Existential instantiation (EI)

- For any sentence α, variable *v*, and constant symbol *k* that does not appear elsewhere in the knowledge base:
-

$$\frac{\exists v\, \alpha}{\text{Subst}(\{v/k\}, \alpha)}$$

- E.g., $\exists x\ Crown(x) \wedge OnHead(x, John)$ yields:

$$Crown(C_1) \wedge OnHead(C_1, John)$$

provided C_1 is a new constant symbol, called a Skolem constant

Unification

- We can get the inference immediately if we can find a substitution θ such that *King(x)* and *Greedy(x)* match *King(John)* and *Greedy(y)*
-

$\theta = \{x/John, y/John\}$ works

- $Unify(\alpha,\beta) = \theta$ if $\alpha\theta = \beta\theta$
-

p	q	θ
Knows(John,x)	Knows(John,Jane)	
Knows(John,x)	Knows(y,OJ)	
Knows(John,x)	Knows(y,Mother(y))	
Knows(John,x)	Knows(x,OJ)	

- Standardizing apart eliminates overlap of variables, e.g., Knows(z_{17},OJ)

Unification

- We can get the inference immediately if we can find a substitution θ such that *King(x)* and *Greedy(x)* match *King(John)* and *Greedy(y)*
-

θ = {x/John,y/John} works

- Unify(α,β) = θ if αθ = βθ
-

p	q	θ
Knows(John,x)	Knows(John,Jane)	{x/Jane}}
Knows(John,x)	Knows(y,OJ)	
Knows(John,x)	Knows(y,Mother(y))	
Knows(John,x)	Knows(x,OJ)	

- Standardizing apart eliminates overlap of variables, e.g., Knows(z_{17},OJ)
-

Unification

- We can get the inference immediately if we can find a substitution θ such that *King(x)* and *Greedy(x)* match *King(John)* and *Greedy(y)*
-

θ = {x/John,y/John} works

- Unify(α,β) = θ if αθ = βθ
-

p	q	θ
Knows(John,x)	Knows(John,Jane)	{x/Jane}}
Knows(John,x)	Knows(y,OJ)	{x/OJ,y/John}}
Knows(John,x)	Knows(y,Mother(y))	
Knows(John,x)	Knows(x,OJ)	

- Standardizing apart eliminates overlap of variables, e.g., Knows(z_{17},OJ)

Unification

- We can get the inference immediately if we can find a substitution θ such that *King(x)* and *Greedy(x)* match *King(John)* and *Greedy(y)*
-

θ = {x/John,y/John} works

- Unify(α,β) = θ if αθ = βθ
-

p	q	θ
Knows(John,x)	Knows(John,Jane)	{x/Jane}}
Knows(John,x)	Knows(y,OJ)	{x/OJ,y/John}}
Knows(John,x)	Knows(y,Mother(y))	{y/John,x/Mother(John)}}
Knows(John,x)	Knows(x,OJ)	

- Standardizing apart eliminates overlap of variables, e.g., Knows(z_{17},OJ)

Unification

- We can get the inference immediately if we can find a substitution θ such that *King(x)* and *Greedy(x)* match *King(John)* and *Greedy(y)*
-

θ = {x/John,y/John} works

- Unify(α,β) = θ if αθ = βθ
-

p	q	θ
Knows(John,x)	Knows(John,Jane)	{x/Jane}}
Knows(John,x)	Knows(y,OJ)	{x/OJ,y/John}}
Knows(John,x)	Knows(y,Mother(y))	{y/John,x/Mother(John)}}
Knows(John,x)	Knows(x,OJ)	{fail}

- Standardizing apart eliminates overlap of variables, e.g., Knows(z_{17},OJ)

Unification

- To unify *Knows(John,x)* and *Knows(y,z)*,
- θ = {y/John, x/z } or θ = {y/John, x/John, z/John}
- The first unifier is more general than the second.
- There is a single most general unifier (MGU) that is unique up to renaming of variables.
- MGU = { y/John, x/z }

Generalized Modus Ponens (GMP)

$$\frac{p_1', p_2', \ldots, p_n', (p_1 \wedge p_2 \wedge \ldots \wedge p_n \Rightarrow q)}{q\theta}$$

where $p_i'\theta = p_i\theta$ for all i

p_1' is *King*(*John*) p_1 is *King*(*x*)

p_2' is *Greedy*(*y*) p_2 is *Greedy*(*x*)

θ is {x/John,y/John} q is *Evil*(*x*)

q θ is *Evil*(*John*)

- GMP used with KB of definite clauses (exactly one positive literal)
- All variables assumed universally quantified

Resolution: brief summary

- Full first-order version:
- $$\frac{l_1 \vee \cdots \vee l_k, \qquad m_1 \vee \cdots \vee m_n}{(l_1 \vee \cdots \vee l_{i-1} \vee l_{i+1} \vee \cdots \vee l_k \vee m_1 \vee \cdots \vee m_{j-1} \vee m_{j+1} \vee \cdots \vee m_n)\theta}$$

  where $\texttt{Unify}(l_i, \neg m_j) = \theta$.

- The two clauses are assumed to be standardized apart so that they share no variables.
- For example,
- $$\frac{\neg Rich(x) \vee Unhappy(x) \qquad Rich(Ken)}{Unhappy(Ken)}$$

  with $\theta = \{x/Ken\}$

- Apply resolution steps to CNF(KB $\wedge \neg\alpha$); complete for FOL

Conversion to CNF

- Everyone who loves all animals is loved by someone:
  $\forall x\ [\forall y\ (\ \mathit{Animal}(y) \Rightarrow \mathit{Loves}(x,y)\ )\ ] \Rightarrow [\exists y\ \mathit{Loves}(y,x)]$

- 1. Eliminate biconditionals and implications
- $\forall x\ [\neg\forall y\ (\ \neg\mathit{Animal}(y) \vee \mathit{Loves}(x,y)\ )] \vee [\exists y\ \mathit{Loves}(y,x)]$

- 2. Move $\neg$ inwards: $\neg\forall x\ p \equiv \exists x\ \neg p$, $\neg\ \exists x\ p \equiv \forall x\ \neg p$
- $\forall x\ [\exists y\ \neg(\neg\mathit{Animal}(y) \vee \mathit{Loves}(x,y))] \vee [\exists y\ \mathit{Loves}(y,x)]$
  $\forall x\ [\exists y\ \neg\neg\mathit{Animal}(y) \wedge \neg\mathit{Loves}(x,y)] \vee [\exists y\ \mathit{Loves}(y,x)]$
  $\forall x\ [\exists y\ \mathit{Animal}(y) \wedge \neg\mathit{Loves}(x,y)] \vee [\exists y\ \mathit{Loves}(y,x)]$

Conversion to CNF contd.

3. Standardize variables: each quantifier should use a different one

$\forall x\ [\exists y\ Animal(y) \land \neg Loves(x,y)] \lor [\exists z\ Loves(z,x)]$

4. Skolemize: a more general form of existential instantiation.

Each existential variable is replaced by a Skolem function of the enclosing universally quantified variables:

$\forall x\ [Animal(F(x)) \land \neg Loves(x,F(x))] \lor Loves(G(x),x)$

5. Drop universal quantifiers:

$[Animal(F(x)) \land \neg Loves(x,F(x))] \lor Loves(G(x),x)$

6. Distribute $\lor$ over $\land$:

$[Animal(F(x)) \lor Loves(G(x),x)] \land [\neg Loves(x,F(x)) \lor Loves(G(x),x)]$

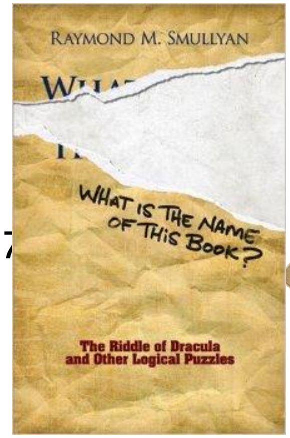
Monotonicity

- If $KB_1 \models a$, then $(KB_1 \cup KB_2) \models a$
- Old theorems are not invalidated by additional axioms.
- Robotics:
  - Inferred results remains valid after expanding the knowledge-base with new facts from observations.
- Non-monotonic logics
  - truth-maintenance systems
  - default logic..

Summary

- First-order logic:
  - objects and relations are semantic primitives
  - syntax: constants, functions, predicates, equality, quantifiers
- Inference
  - Resolution (CNF-based)
  - Semi-decidable

- Suggested reading:
  - Puzzles
    - http://www.greylabyrinth.com/puzzle/puzzle102
    - http://www.greylabyrinth.com/puzzle/puzzle107
  - Interview with R. M. Smullyan
    - http://www.doverpublications.com/mathsci/0227/news.html
  - R. M. Smullyan: *What Is the Name of This Book?*, 197



Abductive inference/reasoning

- C.S.Pierce: inference of the most pragmatical explanation for an observation.
- Types of inference
  - Deduction: model➔observation
  - Induction: observation(s) ➔ model ➔ observation
    - observation(s) ➔ model
    - observation(s) ➔ [model ➔] observation
  - Abduction: observation(s) ➔ model
  - Transduction: observation(s) ➔ observation
  - Causal: intervention ➔ effect
  - Counterfactual: (observation/intervention➔effect) ➔(imagery intervention ➔ imagery effect)
- Related to abduction
  - theories of explanation
  - philosophy of science
  - theories of belief change in artificial intelligence
- Subtypes of abduction
  - Common sense
  - Scientific (Ockham's razor)
  - Logical
  - Probabilistic (most probable explanation)
  - Causal (necessary and sufficient cause)