Common Logic

A Framework for a Family Of Logic-Based Languages

John F. Sowa

22 October 2010

Presented at RuleML 2010 Symposium
How to say “A cat is on a mat.”

Gottlob Frege (1879):

Charles Sanders Peirce (1885):

\[ \Sigma_x \Sigma_y \text{Cat}_x \cdot \text{Mat}_y \cdot \text{On}_{x,y} \]

Giuseppe Peano (1895):

\[ \exists x \exists y \text{Cat}(x) \land \text{Mat}(y) \land \text{On}(x,y) \]

Frege and Peirce developed their notations independently.

Peano adopted Peirce’s notation, but changed the symbols.

But all three notations have identical semantics.
Some Modern Notations

SQL query:

```sql
SELECT FIRST.ID, SECOND.ID
FROM OBJECTS FIRST, OBJECTS SECOND, SUPPORTS
WHERE FIRST.TYPE = "Cat"
AND SECOND.TYPE = "Mat"
AND SUPPORTS.SUPPORTER = SECOND.ID
AND SUPPORTS.SUPPORTEE = FIRST.ID
```

Common Logic Interchange Format (ISO 24707):

```plaintext
(exists ((x Cat) (y Mat)) (On x y))
```

Conceptual Graph Interchange Format (ISO 24707):

```
[Cat *x] [Mat *y] (On ?x ?y)
```

Conceptual Graph Display Format:

![Graph of a cat on a mat]

Controlled English:

A cat is on a mat.
A Family of Languages

First-order logic is a subset or superset of most logic-based notations. But people are constantly inventing new notations, and they don't want to abandon their favorite notation in favor anybody else's.

The ISO standard 24707 for Common Logic defines a very general semantic foundation for an open-ended family of dialects.

Three dialects are specified in ISO 24707:

* CLIF — Common Logic Interchange Format
* CGIF — Conceptual Graph Interchange Format
* XCL — XML-based notation for Common Logic

But any notation that uses the common semantics can join the family.
Human Interfaces

Controlled English
Controlled Spanish
Controlled Chinese

FLIPP Diagrams
Concept Maps
Topic Maps
UML Diagrams

Common Logic

XCL
CGIF
CLIF

SQL
Prolog
RDF(S)
OWL

OCL
Datalog
RuleML

Machine Interfaces
History

1992: ANSI projects for the Knowledge Interface Format (KIF) and Conceptual Graphs (CGs) started in the X3H4 committee.

1994: X3T2 inherits KIF and CG projects; sponsors workshops on ontology for an ISO project on Conceptual Schemas.

1999: Conceptual Schema project ends with a technical report.

2000: NCITS L8 merges KIF and CG projects in a proposal to SC32 for an ISO standard for Common Logic.

2001: Pat Hayes and Chris Menzel propose a new semantic foundation for Common Logic that is compatible with RDF.

2002: Guha and Hayes use the CL semantics to define the logic base (LBase) for RDF.


2012+: Target date for updates and extensions for a revised version of the CL standard.
Common Logic Controlled English

A dialect of Common Logic that looks like English.
CLCE uses a subset of English syntax and vocabulary.
But the grammar avoids constructions that may cause ambiguities.
CLCE replaces pronouns with temporary names called \textit{variables}.
Examples:

For every company C,
extactly one manager in C is the CEO of C;
every employee of C except the CEO reports to the CEO;
the CEO of C does not report to any employee of C.

If an integer N is 5, then \((N^3 = 125)\).

The scope of variables, such as C or N, extends to the final period.

Note: CLCE is not an ISO standard, but it maps to the CL semantics.
CLCE Semantics

CLCE can express the full semantics of Common Logic.

A recursive definition of "reports" in terms of "directly reports":

Every employee who directly reports to a manager reports to that manager.

If an employee of a company C directly reports to a manager M1 in C, and the manager M1 reports to a manager M2 in C, then the employee reports to the manager M2.

Definitions link CLCE words and phrases to other versions of logic:

Define "x directly reports to y" as (DirectlyReports x y).

Define "x directly reports to y" as SQL(select emp, mgr from employees).
CLCE:  Bob drives his old Chevy to St. Louis.

Conceptual graph display form:

```
Person: Bob  Agnt  Drive  Dest  City: "St. Louis"
      Poss  Thme  Chevy  Attr  Old
```

Conceptual Graph Interchange Format (CGIF):

```
[Drive *x] [Person Bob] [City "St. Louis"] [Chevy *y] [Old *z]
(Agent ?x Bob) (Dest ?x "St. Louis") (Theme ?x ?y) (Poss Bob ?y)
(Attr ?y ?z)
```

Common Logic Interchange Format (CLIF):

```
(exists ((x Drive) (y Chevy) (z Old)))
   (and (Person Bob) (City "St. Louis") (Agent x Bob)
       (Dest x "St. Louis") (Theme x y) (Poss Bob y) (Attr y z))
```
CLCE: If a cat is on a mat, then the cat is a happy pet.

Conceptual graph display form:

CGIF:

[If: [Cat: *x] [Mat: *y] (On ?x ?y)  
  [Then: [Pet: ?x] [Happy: *z] (Attr ?x ?z) ]]

CLIF:

(not (exists (x y) (and (Cat x) (Mat y) (On x y)  
  (not (exists (z) (and (Pet x) (Happy z) (Attr x z))))))))
A Logically Equivalent Variation

CLCE: For every cat \(x\) and every mat \(y\),
if \(x\) is on \(y\), then \(x\) is a happy pet.

CGIF:

\[\text{[Cat: @} \text{every } *x\text{]} \ [\text{Mat: @} \text{every } *y\text{]}\]
\[\text{[If: } (\text{On } ?x \ ?y)\]
\[\quad \text{[Then: } [\text{Pet: } ?x\text{]} [\text{Happy: } *z\text{]} (\text{Attr } ?x \ ?z) ]]\]

CLIF:

\[(\text{forall } ((x \text{ Cat}) \ (y \text{ Mat}))\]
\[\ (\text{if } (\text{On } x \ y)\]
\[\quad \ (\text{and } (\text{Pet } x) \ (\text{exists } ((z \text{ Happy})) \ (\text{Attr } x \ z))))\]

Most dialects of logic and natural languages permit many different ways of expressing semantically equivalent statements.
CLCE: For a number $x$, a number $y$ is $((x + 7) / \sqrt{7})$.

Conceptual graph display form:

CGIF:

$[\text{Number: } *x] [\text{Number: } *y]$

$(\text{Add } ?x 7 | *u) (\text{Sqrt } 7 | *v) (\text{Divide } ?u ?v | ?y)$

CLIF:

$(\text{exists } ((x \text{ Number}) (y \text{ Number})))$

$(= y (\text{Divide} (\text{Add} x 7) (\text{Sqrt} 7)))$
Quantifying Over Relations

Although Common Logic has a first-order semantics, it does permit quantified variables to refer to functions and relations.

English: Bob and Sue are related.

CLCE: There is a familial relation between Bob and Sue.

CGIF:

[Relation: *r] (Familial ?r) (#?r Bob Sue)

CLIF:

(exists ((r Relation)) (and (Familial r) (r Bob Sue)))

Note: the option of quantifying over relations is necessary to support RDF, but it goes beyond KIF.
The word “relation” is not a reserved word in CLCE.

But CLCE allows new words to be defined by their mapping to CGIF, CLIF, or other languages, such as SQL:

Define "familial relation r" as (and (Familial r) (Relation r)).

Define "relation r between x and y" as (and (Relation r) (r x y)).

With these definitions, the following CLCE sentence can be translated to the CLIF and CGIF sentences in the previous slide:

There is a familial relation between Bob and Sue.
Wish List of Extensions to CL

Metalanguage.
  • Names for propositions and statements about propositions.
  • Statements that relate propositions to other propositions.

Nonmonotonic reasoning.
  • Default logics, negation as failure (e.g., SQL and Prolog).
  • Belief or theory revision methods for classical FOL theories.

Uncertainty, vagueness, and fuzziness.

Modality.
  • Alethic (necessity and possibility).
  • Epistemic (knowledge and belief).
  • Deontic (obligation and permission).

Contexts and microtheories.

Can all these features be represented by an extension to CL?
IKRIS Project

DoD-sponsored project: Design an Interoperable Knowledge Language (IKL) as an extension to Common Logic.

Goals:
- Enable interoperability among advanced reasoning systems.
- Test that capability on highly expressive AI languages.

Show that semantics is preserved in round-trip mapping tests:
- Cycorp: Cyc Language $\rightarrow$ IKL $\rightarrow$ CycL
- RPI / Booz-Allen: Multi-Sorted Logic $\rightarrow$ IKL $\rightarrow$ MSL
- Stanford/IBM/Battelle: KIF $\rightarrow$ IKL $\rightarrow$ KIF
- KIF $\rightarrow$ IKL $\rightarrow$ CycL $\rightarrow$ IKL $\rightarrow$ MSL $\rightarrow$ IKL $\rightarrow$ KIF

Conclusion: “IKRIS protocols and translation technologies function as planned for the sample problems addressed.”

The IKL Extension to Common Logic

Common Logic is a superset of the logics used in many semantic systems, but some systems require even more expressive logics.

Only one new operator is needed: a metalanguage enclosure, which uses the keyword 'that' to mark the enclosed statement.

- The enclosed statement denotes a proposition.
- That proposition could be a conjunction of many statements.
- It can be given a name, and other propositions can refer to it.
- In effect, IKL can be used as a metalanguage for talking about and relating packages of IKL statements nested to any depth.

CL with the IKL extensions can represent a wide range of logics for modality, defaults, probability, uncertainty, and fuzziness.

Using CLCE to Express IKL

The operator 'that' of IKL can be used in CLCE:

Tom believes that Mary knows that (2 + 2 = 4).

And in CLIF notation for IKL:

(believes Tom (that (knows Mary (that (= (+ 2 2) 4))))))

The operator 'that' is a powerful metalevel extension.

It enables IKL to specify languages, define their semantics, and specify transformations from one language to another.

Anybody who has not spent years studying logic is unlikely to use CLCE correctly to state all the nuances.

But CLCE can express such nuances in a readable way that a wider audience, including logicians, can appreciate.
The two CGs above show two different interpretations of the sentence *Tom believes that Mary wants to marry a sailor*:

- Tom believes a proposition that Mary wants a situation in which there exists a sailor whom she marries.
- There exists a sailor, and Tom believes a proposition that Mary wants a situation in which she marries the sailor.

The IKL semantics permits the quantifier for “a sailor” to include the enclosed statements within its scope.

For further discussion of these issues, see [http://www.jfsowa.com/pubs/cg4cs.pdf](http://www.jfsowa.com/pubs/cg4cs.pdf)
Representing IKL in CLIF and CGIF

Following is the CGIF representation for the CG on the left of the previous slide:

[Person: Tom] [Believe: *x1] (Expr ?x1 Tom) (Thme ?x1 [Proposition: [Person: Mary] [Want: *x2] (Expr ?x2 Mary) (Thme ?x2 [Situation: [Marry: *x3] [Sailor: *x4] (Agnt ?x3 Mary) (Thme ?x3 ?x4)])])

In CLIF notation, the operator 'that’ applied to a CL or IKL sentence denotes the proposition stated by the sentence:

(exists ((x1 Believe)) (and (Person Tom) (Expr x1 Tom) (Thme x1 (that (exists ((x2 Want) (s Situation)) (and (Person Mary) (Expr x2 Mary) (Thme x2 s) (Dscr s (that (exists ((x3 Marry) (x4 Sailor)) (and (Agnt x3 Mary) (Thme x3 x4) )))))))))))

To represent the CG on the right of the previous slide, move the concept node [Sailor: *x4] in front of the concept [Person: Tom] for CGIF notation. For CLIF, move (x4 Sailor) in front of (x1 Believe).
Semantics for Modal Logic

Saul Kripke: Possible worlds with an accessibility relation.
  • A proposition p is necessary in a world w iff p is true in every world accessible from w.
  • p is possible in w iff p is true in some world accessible from w.

Michael Dunn: Each world is represented by laws and facts.
  • p is necessary in w iff p is provable from the laws of w.
  • p is possible in w iff p is consistent with the laws of w.

Kripke semantics and Dunn semantics are equivalent.
  • Accessibility is a derived relation from the laws and facts.
  • A world w2 is accessible from w1 iff every law of w1 remains true in w2 (but some laws of w1 might only be facts in w2).

Although both versions of semantics are logically equivalent, Dunn's version is easier to map to computer implementations.

For further detail and references, see http://www.jfsowa.com/pubs/laws.htm and http://www.jfsowa.com/pubs/worlds.pdf
Using the Metalanguage Option of IKL

Modal terminology is used to distinguish laws from facts.
  • Unqualified assertions are assumed to be facts.
  • Words 'necessary', 'obligatory', or 'must' indicate laws.

Default logic is a metalevel specification for a family of theories.
  • Ordinary axioms specify a base theory.
  • Each default specifies an optional axiom added to the base theory.
  • A nonmonotonic proof chooses one theory from the entire family.

Uncertainty and fuzziness are also metalevel statements.
  • They define numeric measures over a family of theories.

Context theories also use metalevel markers and reasoning.

IKL, by itself, does not define any of these systems, but it provides the framework and primitives for stating such definitions.
Some minor updates, revisions, and corrections have been suggested for the ISO/IEC standard 24707 for Common Logic.

One major proposal is to adopt the IKL extension for CL 2.0.

Another proposal is to replace the current XCL, which has had very little use, with a notation based on the logic subset of MathML.

The Horn-clause subset of XCL 2.0 should be compatible with the notations for RIF and RuleML.

Any experts in RIF and RuleML would be welcome to collaborate in the design of XCL 2.0. See

http://www.common-logic.org

Begin by subscribing to the email list and take part in the discussion.
Related Readings

ISO/IEC standard 24707 for Common Logic:

“Fads and Fallacies About Logic,” by John F. Sowa

“Conceptual Graphs,” by John F. Sowa

Web site for Common Logic:
   http://www.common-logic.org

Web site for controlled natural languages:
   http://www.ics.mq.edu.au/~rolfs/controlled-natural-languages/