Towards Conceptual Structures Interoperability Using Common Logic

Harry S. Delugach

Computer Science Department
Univ. of Alabama in Huntsville
Huntsville, AL 35899
delugach@cs.uah.edu

Abstract. The Common Logic (CL) ISO standard has been officially published and available. While its focus goes beyond the conceptual structures community, one of its components is an interchange format for conceptual graphs. Now the community has an opportunity to leverage the standard for tool usage. Current tools that support pre-ISO versions must now support the standard. Future tools will be much more useful if they too support the standard. This paper describes the CL effort, outlines its main features and issues a call to action.

Keywords: Common Logic, conceptual graph interchange format, CGIF, international standard

1 Introduction

Over the years, a number of knowledge representations have been proposed, e.g., the Knowledge Interchange Format (KIF) [1], conceptual graphs (CGs) [2, 3], the W3C's Resource Description Framework (RDF) [4], various forms of the W3C's Web Ontology Language (OWL) [5] and there have been occasional efforts to establish ways to translate between them, or else to develop techniques for interoperability between systems that use the various representations. Recently one such effort – Common Logic (CL) – has reached the status of an officially published international standard [6]. This standard is freely available to the public.

While the focus of CL goes beyond just the conceptual structures community, one of its components is an interchange format for conceptual graphs. Now for the first time the conceptual graph research community has an opportunity to leverage the standard for tool interoperability and further promulgation of CGs. Current tools that support pre-ISO versions should be updated to support the standard. This paper describes the effort, its main features and then suggests some of the potential benefits of its use.

2 A (partial) interoperability history of conceptual graphs

The story of conceptual structures interoperability has several chapters. Since conceptual graphs were first proposed [2], efforts have been made to support its interoperability. Indeed, that introduction of conceptual graphs contained a set of "standard" concept and relation types, along with a text-based format known as the "linear form" of conceptual graphs (LF). This linear form had the twin goals of being able to be automatically parsed by software and the ability to be understood by human readers. Consider the conceptual graph example in Figure 1:

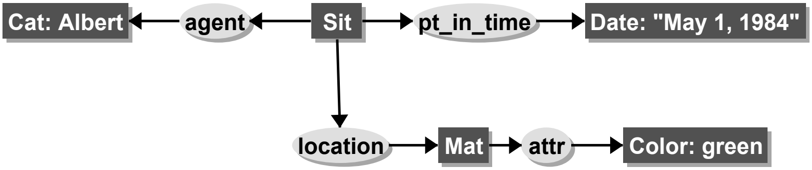

Figure 1. Example Conceptual Graph.

One linear form of this graph is the following:

```
[Cat: Albert]-> (agent) -> [Sit] –
   -> (pt-in-time) -> [Time: "May 1984"]
   -> (location) -> [Mat] -> (attr) -> [Color: green]
```

The linear form has been useful and versatile within online discussions and emails where small graphs need to be shown unambiguously.

The need for interoperable tools was already well understood even before the first ICCS conference. Gerard Ellis and Bob Levinson began an effort known as PEIRCE which was to create a standard workbench for conceptual graphs such that its tools could interoperate with each other [7]. The effort never realized its full potential, for several reasons; one of those was the lack of a standard (other than the LF). John Sowa began to explore options for a more interoperable standard.

A few years after CGs became popular, the Knowledge Sharing Effort (KSE) of the U.S. DARPA research agency was using another knowledge representation language. Their language, known as the Knowledge Interchange Format (KIF) was also a first-order logic language, with extensions for non-monotonic reasoning and definitions [1].

Around 1994, it became clear to researchers in both the CG and KIF communities that they were both addressing issues of interoperability. They realized it was in their best interests to work toward standardizing both CGs and KIF so that (a) they would be interoperable with each other and (b) they would express the same semantics. This effort became known as the Common Logic effort, but it was not successful in reaching the status of an approved standard.

A side effort during these activities was the creation of the CharGer CG editing package [8], along with a published XML version of conceptual graphs which included some meta-information such as graph layout, creation time-date, etc. for a conceptual graph.

One result of that early Common Logic effort was a new textual form of conceptual graphs, known as the Conceptual Graph Interchange Format (CGIF). This version was widely publicized by being available on John Sowa's website for many years (www.jfsowa.com). In fact, many people in the CG community are unaware that this version (which is still online) has been superceded by the ISO standard. Furthermore, many researchers are building tools that still support this superceded standard.

The graph in Figure 1 can be denoted in CGIF as follows:

```
[Cat: Albert ] [Sit: *s ] [Date: *d “May 1, 1984”]
[Mat: *m ] [Color: *c green]
(agent ?s Albert) (pt_in_time ?s ?d) (location ?s ?m )
(attr ?m ?c)
```

The main obstacle preventing the creation of the original Common Logic standard was that participants had different ideas about what features beyond strict first-order logic (modalities, sorts, etc.) ought to be included. In 2003, therefore, a new standardization effort was organized, led by Pat Hayes, Chris Menzel and John Sowa. This new effort was originally called Simplified Common Logic (SCL), which represented both the inauguration of the new effort, as well as a new philosophy of creating a standard with a smaller set of features that could be more easily agreed upon. This effort was aimed toward establishing an international standard, in conjunction with the W3C's desire to create a standard for interoperability of ontology information in conjunction with OWL and RDF.

In June 2003, I was asked to serve as editor for the Common Logic ISO standard. The project was once again called Common Logic (CL, not SCL) in order to match the standard's title. In October 2007, the ISO Common Logic standard became a full-fledged International Standard now designated as ISO/IEC 24707:2007. In April 2008, the standard was designated a publicly available standard, so that one copy can be downloaded from ISO by anyone who wants to use it.

Here is a brief summary of the CG representations discussed in this paper.

Name	**Referred to in this paper**	**Source**	**Status**
Linear Form	Linear Form (LF)	[2]	de facto standard
Sowa's CGIF	Pre-ISO CGIF	http://www.jfsowa.com/cg/cgstandw.htm (still online as of 1 Apr 2008)	Superceded
Conceptual Graph Interchange Format	CGIF	ISO/IEC 24707:2007 [6]	ISO standard freely-available
CharGer-format	CGX	http://projects.sourceforge.net/charger	Freely available

3 Description Of Common Logic

The Common Logic Standard is not just for conceptual graphs; indeed, its main purpose is to provide interoperability for both syntax and semantics for a family of first-order logic languages. The standard's main sections prescribe a set of syntactic and semantic categories: any language dialect that provides both the syntax and semantics is eligible for conformance. The semantics are based on well-understood model theory [9].

3.1 Common Logic Semantics

Common Logic semantics are based on model theory such that any CL text is required to be based on model theoretic interpretations. Simply put, a CL text is associated with a set of individuals, the "universe of discourse," that the text is "about". Because CL also allows functions and relations to be asserted, there is a (potentially larger) set of all things to which the text may refer (including functions and relations); this potentially large set is called the "universe of reference". According to the model theoretic requirements, every symbol in CL is assumed to have some procedure (outside of the model itself) that associates that symbol with specific element(s) in the universe of reference. This association is referred to as the *interpretation* of the symbol.

This means that every model that claims CL conformance must have a model theoretic interpretation over a set of individuals called the *universe of discourse* (*UD*). For CGs, that requirement is addressed by having a set of individual markers in a CG model, each of which denotes a specific individual in *UD*. Every concept has associated with it an (explicit or implicit) individual marker, denoting the individual. Regardless of whether the marker is explicit or implicit, it is required for a CG model that there exist some consistent means of distinguishing the individual to which a marker refers. For example, the procedure must associate marker `#483` with some individual in *UD*; whenever marker `#483` appears, it must be associated with that same individual.

The impact of the model theory on the use of CG models is straightforward: every concept denotes an individual in some *UD*, so that there must always be a *UD* assumed for any given CG model. The impact on interoperability is more subtle: when a CG model is transferred, its particular *UD* is not explicitly included. Any other system that uses the model may only assume that there exists such a *UD*. This is further explained below.

An important feature for interoperability is CL's ability to incorporate comments in a CL text. These can serve much the same purpose as in any programming or formal specification language – as uninterpreted human-readable text to aid in understanding – but they can also serve as a means to include extra-logical information (i.e., beyond CL semantics). For example, a CGIF text could include any of the following (suitably structured for automated processing, of course) as comments:

- Information about the graphical layout of graph elements (relative page locations, color, fonts, etc.),
- Knowledge about the provenance, origin or custodian of the knowledge being represented
- Executable code for the specification of actors

Though CL comments are necessarily uninterpreted, CL's semantics nevertheless allow comments to be attached to particular phrases in a text. This means they can effectively annotate concepts, relations, or contexts.

3.2 Common Logic Dialects

The Common Logic standard does not prescribe a single syntax, but instead prescribes a common abstract semantics for CL syntaxes. Three separate concrete syntaxes are specified in the CL standard as appendices (each called an "annex" in ISO style).

Common Logic Interchange Format (CLIF)

This dialect is based on KIF and resembles LISP syntax. It is specified in Annex A of [6].

Conceptual Graph Interchange Format (CGIF)

This dialect is based on CGs; its syntax is based on the original pre-ISO CGIF which somewhat resembles the linear form but carefully crafted so that is can be parsed in one pass. It is specified in Annex B of [6].

Extended Common Logic Markup Language (XCL)

This is a new dialect expressed in XML but developed especially for this standard. It is specified in Annex C of [6].

As for all ISO/IEC standards, there is an expected review every five years. This gives a nice time frame during which we can explore the standard and have a better idea of how it might be revised.

3.3 Relationship to other representations

Since Common Logic grew partly out of the W3C community, it fills a technical niche that positions it next to the most popular representations of RDF [4] and OWL [10] [5]. In brief, since RDF expresses only binary relations, CL is more expressive. For two of the three main “species” of OWL, the comparison is straightforward. OWL Lite is primarily for describing classification hierarchies while OWL DL is based on description logics [11] which are a set of languages, each of which is a decidable subset of first-order logic. CL is therefore more expressive than any of these.

The third OWL species, OWL Full, is meant to be more expressive and therefore possess no computational guarantees; i.e., some queries on an OWL knowledge base are undecidable in polynomial time. OWL Full’s expressiveness is comparable to CL’s, but without the benefit of a century of study in dealing with first-order logic.

One difficulty with the OWL species is that practitioners must decide which of them to use for their purposes. Given a limited domain, it is certainly preferable to use a less expressive (and thereby more computable) representation, if that’s all that is needed. It is often the case, however, that a domain’s limited purposes may become expanded over time, requiring some rework of the knowledge base into some more expressive form.

Expressivity is obviously important for interoperability: if we transfer some text to another knowledge system that is less expressive, then we either must accept that some of our knowledge will be lost or (worse) that it will be misinterpreted in the less expressive system and lead to incorrect inferences. CL’s expressiveness means that we can be sure that the meaning of any RDF, OWL Lite or OWL DL representation can be preserved. Efforts to accommodate OWL Full are ongoing.

4 Interoperability Issues

There are some significant differences between Common Logic and the CG theory as described in [2]. These differences may affect some current semantics of CG tools as they are currently constituted.

Implicit UD

It was mentioned above that when a CG model is transferred, its particular *UD* is not explicitly included. Any other system that uses the model may only assume that there exists such a *UD*. For example, if a CG model is transferred from its origin to a new system, the marker **`#483`** must denote some individual in the originating system's assumed *UD*, but the new system cannot "know" to what individual (e.g., some actual person, location, point-in-time, etc.) the marker refers. If the new system performs logical inferences on the model, e.g., inferring that the individual denoted by marker **`#483`** has the color green, then the results of that inference must be true in the originating system as well. Thus CL's semantics under interoperability are limited to knowing that there is *some* model theoretic interpretation that is valid for the model, without knowing exactly *what* that interpretation is.

Untyped logic

CL does not support types (technically referred to as "sorts" in logic; see [12]), nor any notion of a type hierarchy. In theory, we can treat a type as simply a function on individuals that returns their type; e.g., **`type(Albert)`** would return **`Cat`**. This does not, however, address the issue of how to declare type symbols in the first place, how to associate those type names with particular individuals, how to specify subsumption rules or how to specify reasoning with subsumption. All these issues need to be addressed.

Functions

CL includes the ability to specify functions, which are not specifically supported in CGs. A CL function is a formula whose evaluation results in a denotation of some individual in the *UD* (i.e., it "returns" that individual's identity). It is likely that CG actors can provide this capability, but this still must be clearly demonstrated to the research community, which has been reluctant to adopt actors into most CG tools.

5 Community Research Challenges

Considerable effort has been expended toward the creation of the standard. A dozen or so people have contributed or made substantive technical critique. Some dozen or so meetings in several countries have considered various aspects of the standard. A number of other organizations and initiatives have expressed interest in CL as a useful component for ontologies, knowledge-based systems, automated reasoners, metadata registries, etc.

Explore the standard!

There are some issues with respect to the standard that will need addressing. For example, do CGs provide a model theoretic interpretation of conceptual relations in accordance with the model theory prescribed? Do actors truly provide the capability to represent functions as specified in the standard? How should types and type hierarchies be specified in a standard way, since they are not covered in CL? These and other questions will need the attention of the research community.

Support the standard!

Many researchers are interested in using CGIF (or already are!) as an interchange format. Obviously the more people who are interested, the more tools and software will emerge that support the standard. A standard is not like a legal statute – it is actually an agreement and a commitment: an agreement that it provides useful features and a commitment to adhere to the standard so that its utility can be realized. While the approval of a standard may seem like the end of a process (especially to those involved in the meetings and discussions!), in fact, it is really a beginning.

Critique the standard!

Users of a standard sometimes see it as a "done deal" – a process that takes place out of sight and whose participants are trying to persuade others to buy into their ideas. But a standard is only as good as its users think it is. Since this is the first version of CL, there are no doubt some issues to be identified and discussed.

Contribute to the standard!

One of the things I have learned is that anyone may contribute to the making of a standard. If you are interested, you can easily communicate with the people involved in the standard's development. Most standards really have only a few people involved in the technical details; if these interest you, then you are encouraged to get involved.

6 Conclusion

The completion of the CL effort is an exciting development for conceptual structures. Now that a completed standard is widely available, the conceptual structures community, especially those interested in conceptual graphs, have an opportunity to build on its potential. One measure of success in a community is whether it supports tools and interoperability. It is therefore imperative that the community rises to the challenge and show all that a standard can help us accomplish.

Acknowledgements. I would to thank the anonymous reviewers for their helpful comments and suggestions. I would also like to thank Pat Hayes, John Sowa and Chris Menzel for their guidance and patience during the lengthy standardization process, in both helping me understand the standard itself and also understanding the power and pitfalls of logic.

References

1. M.R. Genesereth and R.E. Fikes, *Knowledge Interchange Format Version 3.0 Reference Manual*, Technical Report KSL-92-86, Computer Science Department, Stanford University, 1992.
2. J.F. Sowa, *Conceptual Structures: Information Processing in Mind and Machine*, Addison-Wesley, 1984, p. 481.
3. J.F. Sowa, *Knowledge Representation: Logical, Philosophical, and Computational Foundations*, Brooks/Cole, 2000.
4. D. Brickley and R.V. Guha, "RDF Vocabulary Description Language 1.0: RDF Schema," 2004; http://www.w3.org/TR/2004/REC-rdf-schema-20040210.
5. G. Antoniou and F. van Harmelen, "Web Ontology Language: OWL," *Handbook on Ontologies*, S. Staab and R. Studer, eds., Springer, 2004, pp. 67-92.
6. ISO/IEC, "ISO/IEC 24707:2007 - Information technology - Common Logic (CL) - A framework for a family of logic-based languages," 2007; http://standards.iso.org/ittf/PubliclyAvailableStandards/c039175_ISO_IEC_24707_2007(E).zip.
7. G. Ellis and R. Levinson, "The Birth of PEIRCE: A Conceptual Graphs Workbench," *Conceptual Structures: Theory and Implementation*, H. D. Pfeiffer and T. E. Nagle, eds., Spinger-Verlag, 1992, pp. 219-228.
8. H. Delugach, "CharGer: Some Lessons Learned and New Directions," *Working with Conceptual Structures: Contributions to ICCS 2000*, G. Stumme, ed., Shaker Verlag, 2000, pp. 306-309.
9. W. Hodges, *A Shorter Model Theory*, Cambridge University Press, 1997.
10. M.K. Smith, et al., "OWL Web Ontology Language Guide," 2004; http://www.w3.org/TR/owl-guide/.
11. F. Baader, et al., "Description Logics," *Handbook on Ontologies*, S. Staab and R. Studer, eds., Springer, 2004, pp. 3-28.
12. M. Davis, "First Order Logic," *Handbook of Logic in Artificial Intelligence and Logic Programming*1, D. M. Gabbay, et al., eds., Oxford Univ. Press, 1993, pp. 31-65.