# Simple Common Logic: A Constraint Language for the ODM

David Frankel
David Frankel Consulting
741 Santiago Court
Chico, CA 95973
df@davidfrankelconsulting.com

Pat Hayes
Institute for Human & Machine Cognition
University of West Florida
40 S. Alcaniz Street, Pensacola, FL 32502
phayes@ihmc.us

Elisa Kendall
Sandpiper Software
2053 Grant Road #162
Los Altos, CA 94024
ekendall@sandsoft.com

Deborah McGuinness
Knowledge Systems Laboratory
Stanford University
Gates Bldg. 2A, Stanford CA
dlm@ksl.stanford.edu

## ABSTRACT

*This paper presents the working draft MOF™ (Meta-Object Facility) metamodel for Simple Common Logic (SCL), one of six metamodels currently envisioned for the Ontology Definition Metamodel (ODM) standards effort in the Object Management Group (OMG®). SCL is positioned for use as a constraint language for the ODM, an alternative to OCL for those who need significantly more flexibility or expressivity. We provide historical context and insight into the design process for developing the metamodel, and briefly discuss ongoing and future work required to complete the SCL-related subset of the ODM.*

## INTRODUCTION

Prior to the issuance of the Object Management Group (OMG®)'s Request For Proposal (RFP) for an Ontology Definition Metamodel (ODM) in March 2003 [1], a number of technical meetings and discussions were held to determine the nature of the requirements that said RFP should incorporate. The resulting RFP reflects the urgency felt among Ontology platform special interest group (PSIG) members and others to address the emerging set of XML-based specifications for ontology development recommended by the Semantic Web Activity of the World Wide Web Consortium (W3C) [2]. It also opens the door for addressing knowledge representation formalisms of greater or lesser expressivity than is currently embodied in the Resource Description Framework (RDF) Schema [3] and Web Ontology Language (OWL) [4,5,6] specifications. These specifications represent only a subset of the broad spectrum of expressivity that some definitions of ontology encompass, ranging from simple taxonomies to complex, higher order, modal, intentional, or probabilistic logical theories.

Based on the results of a significant effort on the part of the joint revised submission team to reflect on, and ultimately publish an analysis of the potential usage scenarios for the ODM [7], we determined that there was a clear requirement to provide a mechanism that supported rules representation. That decision supported the desires of some team members:

- To support parallel efforts in the OMG's Agents PSIG for a content language for agent communications that also met FIPA (Foundation for Physical Intelligent Agents) requirements [8].
- To facilitate Semantic Web Services composition, not only for informative services, but for those that may involve side effects, such as e-commerce services requiring credit card transactions [9]
- To facilitate representation of formulas, scientific theories, and other technical knowledge that might be required for bio and medical informatics, engineering applications, financial analysis, and other potential ODM users

Our conclusion, after reviewing a number of candidate knowledge representation formalisms, was that providing support to the level of first order logic in the ODM was sufficient to meet these goals. Reference [10] provides a more comprehensive view of some of the other issues influencing the overall architecture of the ODM and the resulting set of metamodels, mappings and profiles envisioned for the completed specification.

## SIMPLE COMMON LOGIC

Simple Common Logic (SCL) is a first-order logical language intended for information exchange and transmission over an open network [11]. It allows for a variety of different syntactic forms, called dialects, all expressible within a common XML-based syntax and all sharing a single semantics. The language has declarative semantics, which means that it is possible to understand the meaning of expressions written in SCL without requiring an interpreter to manipulate those expressions. SCL is also logically comprehensive – at its most general, it provides for the expression of arbitrary logical expressions. SCL has a purely first-order semantics, and satisfies all the usual semantic criteria for a first-order language, such as compactness and the downward Skolem-Löwenheim property.

In addition to the requirements identified above, motivation for its consideration as an integral component of the ODM includes:

- The fact that normative mappings from SCL to syntactic forms for several commonly used knowledge representation standards are available, defined in ISO 24707, including the Knowledge Interchange Format [12] and Conceptual Graphs [13].
- The availability of a normative XML-based surface syntax for SCL, also defined in ISO 24707, which dramatically increases its potential for use in web-based applications.
- The availability of a direct mapping from OWL to SCL, such that SCL reasoners can leverage both the ontologies expressed in OWL and constraints written

- in SCL to solve a wider range of problems than can be addressed by OWL alone.
- In general, first order logic provides the basis for most of the commonly used knowledge representation languages, including relational databases; more application domains have been formalized using first order logics than any other formalism – its meta-mathematical properties are thoroughly understood. SCL in particular provides a modern form of first order logic that takes advantage of recent insights in some of these application areas including the Semantic Web.

Although a number of proposals have been put forth to support a rule language for OWL, there is currently no standard available from the W3C. Such a standard may be considered for integration with the ODM at a later date. The Object Constraint Language (OCL) [14] was also considered as a basis for representing constraints with OWL ontologies, but was limited in its ability to support (1) quantification, variable binding and certain compound expressions, (2) variadic relations, and (3) variables, formulas, and equations, all of which are critical to knowledge representation in certain conceptual domains.

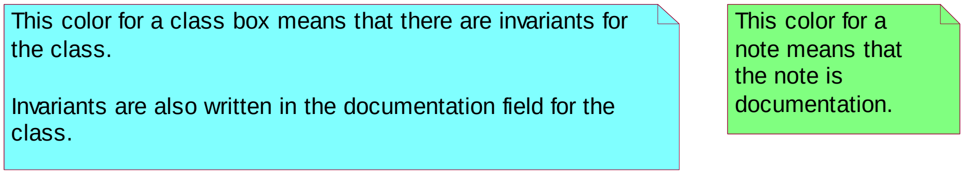
## SCL DEVELOPMENT PROCESS

When the joint submission team was beginning to work on the individual metamodels that would ultimately comprise the ODM, we first considered developing a metamodel for KIF, for which a draft ISO standard was developed in late 2001 [15]. We learned from several of the participants in that ISO effort that their work on KIF had been abandoned in favor of a next generation, XML-savvy language called Simplified Common Logic (since renamed "Simple"), which was still under development. Because the standard was still somewhat fluid at that point (December 2003 and earlier this year), we were able not only to meet with one of the authors to ensure that we had captured the abstract syntax of the language correctly, but to synergistically develop the metamodel with their enthusiastic support – influencing the design of the language itself in the process. For those of us participating directly in those discussions, this was truly a tremendous, and possibly historic opportunity, one which we hope will become more frequent as Model Driven Architecture™ (MDA®) methodologies, and MOF™ (Meta-Object Facility) [16] in particular, become commonplace.

## THE SCL METAMODEL

The SCL metamodel was developed directly from the Extended Backus-Naur Form (EBNF) of the abstract syntax for the language. The diagrams are organized according to the organization of the EBNF, and in order to promote understanding, we have included the EBNF on the diagrams themselves. A complete set of OCL constraints, further refining the metamodel for use by MOF-based repositories and tools, is provided in the draft submission document [17].

Stylistically, color has been used in the notes on the diagrams that follow strictly to provide context; color-coded classes highlight definitions that include OCL constraints, as shown in Figure 1.



This color for a class box means that there are invariants for the class.

Invariants are also written in the documentation field for the class.

This color for a note means that the note is documentation.

*Figure 1. SCL Metamodel Color Conventions*

The metamodel itself is organized in terms of seven diagrams:

- Phrases — Phrases, as shown in Figure 2, provide mechanisms for grouping and scoping the elements that constitute an ontology (or set of constraints associated with an OWL ontology), authored in SCL or any of its syntactic variants.
- Expressions — The Expressions diagram provides a slightly different view of the primary syntactic elements of SCL.
- Terms — The Terms diagram, given in Figure 3, provides additional clarification on the syntactic elements that are valid terms in SCL. These include names, commented terms, and functions.
- Atoms — Atomic sentences are similar in structure to terms; but in addition, the arguments to an atomic sentence may be represented using role-pairs consisting of a role-name and a term. Equations are also considered to be atomic sentences, as shown in Figure 4.
- Sentences — SCL provides facilities for expressing several kinds of sentences, including atomic sentences as well as compound sentences built up from atomic sentences or terms with a set of logical constructors.
- Boolean Sentences — SCL requires implications and biconditional relations (*iff*) to be binary, but allows logical conjunctions and disjunctions to have any number of arguments (including zero— see Fig. 5).

**phrase = sentence | open, 'scl:imports', name, close | open, 'scl:comment', (quotedstring | enclosedname), close | name, phrase ;**
**scltext = { phrase } ;**
**moduledefinition = open, 'scl:module' , name , [open, 'scl:header' , scltext , close ], scltext , close ;**

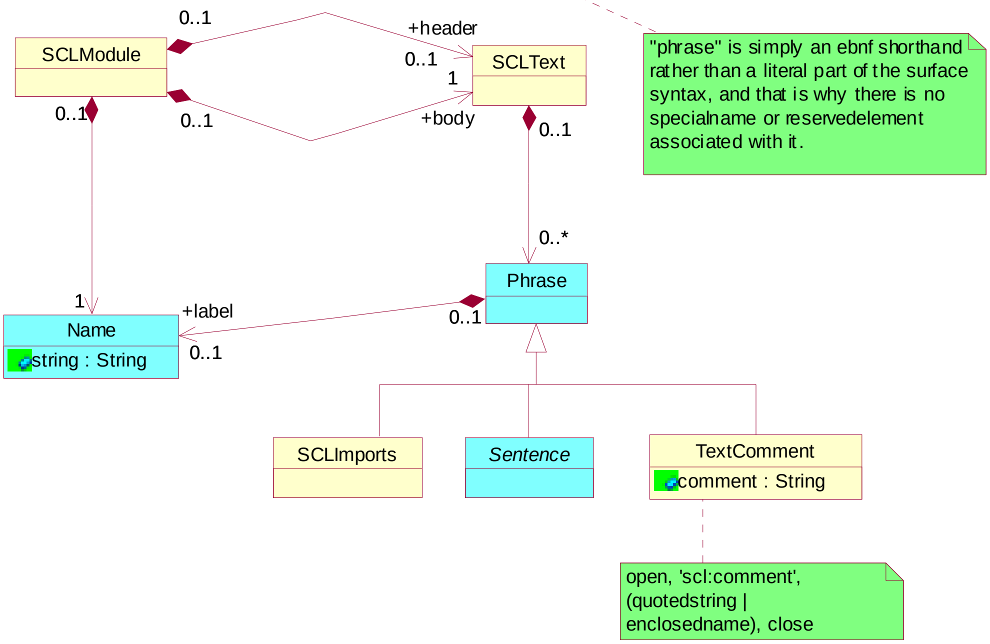


*Figure 2. SCL Phrases Diagram*

**seqvar = '...' , { char } ;**
**open = '(' ;**
**close = ')' ;**

**termseq = { term } , [ seqvar ] ;**
**term = name | ( ( open, term ) | nondenotingnameopen ), termseq, close | ( open, 'scl:comment', ( quotedstring | enclosedname ), term, close ) ;**

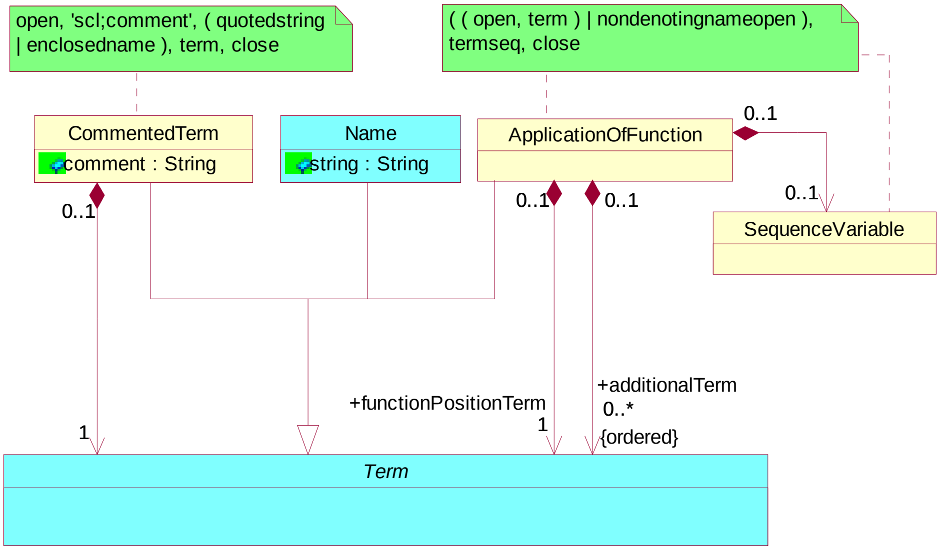


*Figure 3. SCL Terms Diagram*

**equation = open, '=', term, term, close ;**
**atomsent = equation | ( ( open, term ) | nondenotingnameopen ), ( termseq |**
**( open, term, open, 'roleset:' , { open, name, term, close } , close ) ), close ) ;**

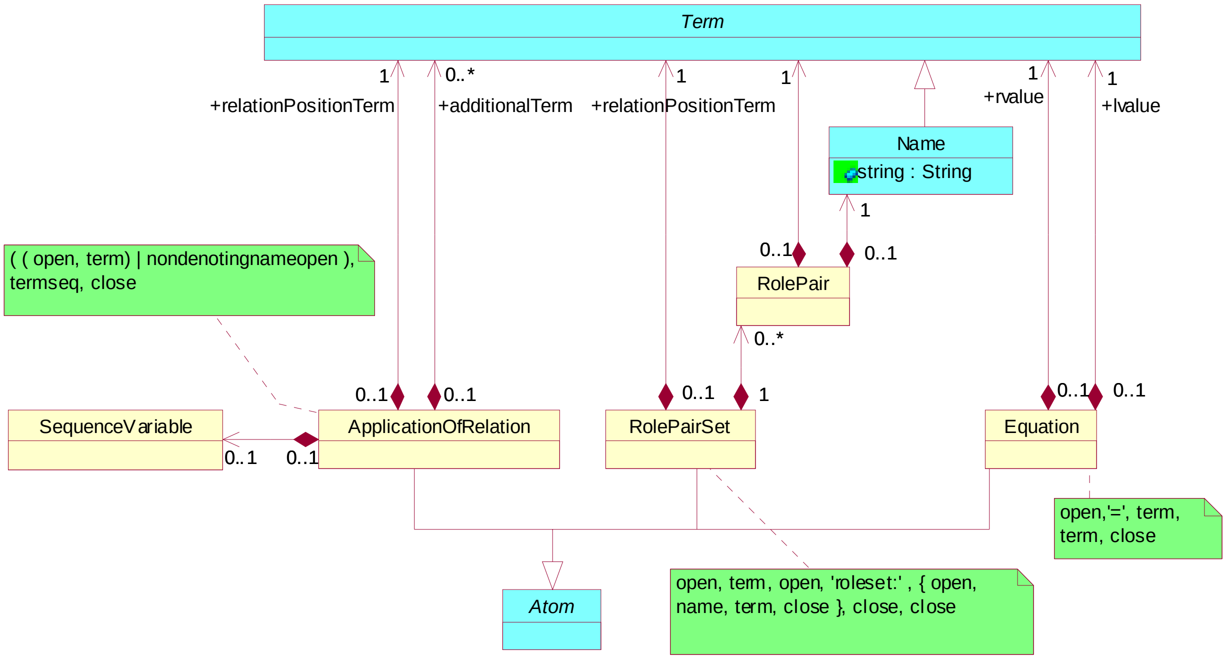
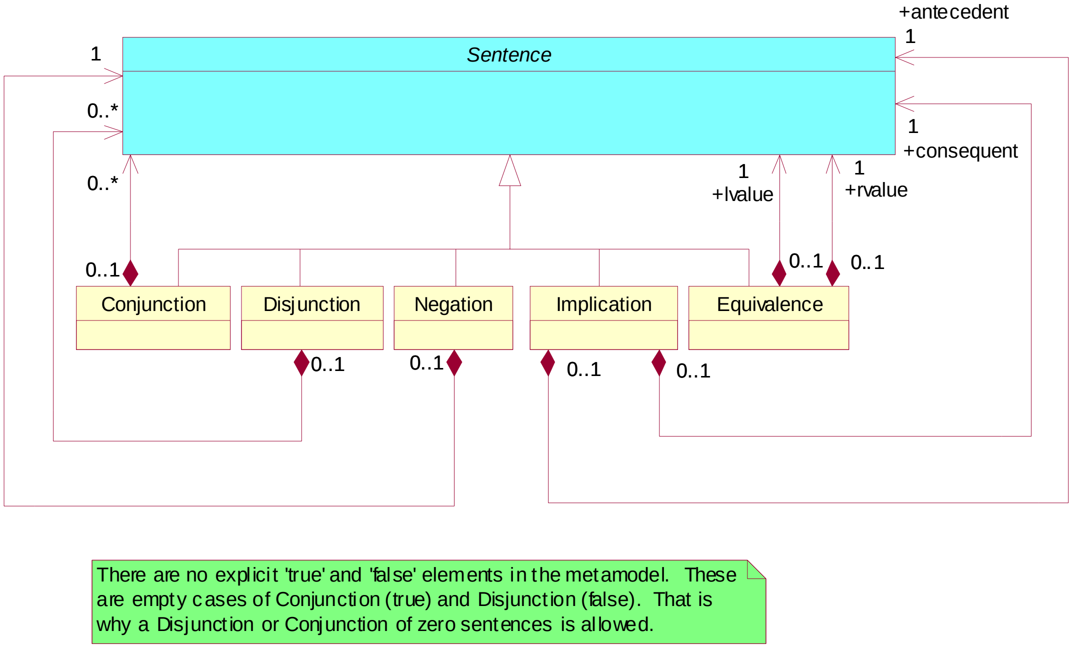


*Figure 4. SCL Atomic Sentences Diagram*

**boolsent = ( open, ('and' | 'or') , { sentence }, close ) |**
**( open, ('implies' | 'iff') , sentence , sentence, close ) |**
**( open, 'not' , sentence, close ) ;**



There are no explicit 'true' and 'false' elements in the metamodel. These are empty cases of Conjunction (true) and Disjunction (false). That is why a Disjunction or Conjunction of zero sentences is allowed.

*Figure 5. SCL Boolean Sentences Diagram*

- Quantified Sentences — The SCL quantifier syntax allows a single quantifier to bind a sequence of names (variables). The quantifier syntax allows any name to be relativized by a term indicating a sort restriction, with a corresponding form for the existential using conjunction, and also allows for an optional *guard*.

As mentioned above, an incomplete draft revised submission document containing these metamodels is available for comment on the OMG Ontology PSIG web site and on the web site given in [17]. In addition to the metamodel, the submitters will provide a mapping strategy with respect to the DL Core metamodel that limits loss due to differences in expressivity using MOF 2.0 Query / View / Transformation (QVT) [18]. A direct mapping from OWL ontologies to SCL will also be provided. As of this writing, the design effort for the SCL language itself and the metamodels presented herein is largely complete and we do not anticipate that major changes will arise. There may be minor adjustments to support the metamodel mappings and related strategy within the context of the ODM, however.

## AREAS FOR FURTHER RESEARCH

Beyond the basic metamodel, many questions remain open with regard to determining strategies for using SCL in conjunction with OWL ontologies, related ontology generation, how theories represented in UML tools should be managed or checked for consistency, validation capabilities, reasoning capabilities, potential use of SCL as a standalone knowledge representation formalism in UML-based tools, and so forth. We are excited and optimistic about the opportunities for use of the metamodels and SCL in general, and plan to address a number of these issues over the coming months.

## ACKNOWLEDGMENT

The authors wish to thank the other ODM submission team members, including Dan Chang, Yiming Ye, Patrick Emery, Lewis Hart, Bob Colomb, and Mark Dutra for their contributions to this effort. The opinions expressed herein represent the opinions of the authors, and do not necessarily reflect the views of the entire ODM submission team.

## REFERENCES

[1] Ontology Definition Metamodel Request for Proposal, OMG Document ad/2003-03-40.

[2] See http://www.w3.org/2001/sw/.

[3] Dan Brickley and R. V. Guha. RDF Vocabulary Description Language 1.0: RDF Schema, W3C Recommendation, World Wide Web Consortium, Amsterdam, The Netherlands, 10 February 2004. Latest version is available at http://www.w3.org/TR/rdf-schema/ .

[4] Deborah L. McGuinness and Frank van Harmelen, eds., *OWL Web Ontology Language Overview*, W3C Recommendation, 10 February 2004, Available at http://www.w3.org/TR/2004/REC-owl-features-20040210/.

[5] Michael K. Smith, Chris Welty, Deborah L. McGuinness, *OWL Web Ontology Language Guide*, W3C Recommendation, 10 February 2004, Available at http://www.w3.org/TR/2004/REC-owl-guide-20040210/

[6] Mike Dean and Guus Schreiber, eds., Sean Bechhofer, Frank van Harmelen, Jim Hendler, Ian Horrocks, Deborah L. McGuinness, Peter F. Patel-Schneider, and Lynn Andrea Stein. OWL Web Ontology Language 1.0 Reference, W3C Recommendation, World Wide Web Consortium, Amsterdam, The Netherlands, 10 February 2004. Latest version is available at http://www.w3.org/TR/owl-ref/ .

[7] Lewis Hart, Patrick Emery, Robert Colomb, Kerry Raymond, Dan Chang, Yiming Ye, Elisa Kendall, and Mark Dutra, “Usage Scenarios and Goals for Ontology Definition Metamodel”, to appear in *Proceedings of the The Fifth International Conference on Web Information Systems Engineering (WISE’04)*, November 22 – 24, 2004.

[8] See http://www.fipa.org/.

[9] Narayanan, S. and McIlraith, S. “Analysis and Simulation of Web Services,” *Computer Networks*, 42 (2003), 675-693, Elsevier Science, 2003.

[10] Daniel T. Chang and Elisa Kendall, “Major Influences on the Design of ODM,” to appear in *Proceedings, the 8th International IEEE EDOC Conference, 1st International Workshop on the Model Driven Semantic Web (MDSW 2004)*, Monterey, CA, September 21, 2004.

[11] Harry Delugach, *ISO/IEC WD 24707 Information technology – Common Logic (CL) – A Framework for a Family of Logic-Based Languages*, Pacific Northwest National Laboratory, Chantilly, VA, 7 June 2004.

[12] Michael Genesereth & Richard Fikes, *Knowledge Interchange Format, Version 3.0 Reference Manual*, KSL Report KSL-92-86, Knowledge Systems Laboratory, Stanford University, June 1992.

[13] Sowa, J.F., *Conceptual Structures: Information Processing in Mind and Machine*. Addison-Wesley, Reading, MA, 1984.

[14] *UML 2.0 OCL Specification*, ptc/03-10-14, Object Management Group, Needham, MA, October 15, 2003.

[15] See http://cl.tamu.edu/discuss/prop.html.

[16] *Meta-Object Facility (MOF™) Specification, Version 2.0*, Object Management Group, Inc., Needham, MA, August 2003.

[17] See http://codip.grci.com/odm/draft/.

[18] MOF 2.0 Query/View/Transformation Revised Submission, ad/04-04-01, April 5, 2004.