Towards Component-based Software Development in Computational Logic

Kung-Kiu Lau*
Department of Computer Science
University of Manchester
Manchester M13 9PL, UK
kung-kiu@cs.man.ac.uk

Component-based Software Development (CBD) has been hailed as the Industrial Revolution for IT! However, at present CBD is basically just a new name for (distributed) Object-oriented Analysis/Design (OOA/D), and as such it inherits OOA/D's strengths and weaknesses, and in our opinion it will fail to deliver this revolution.

We believe that the key requirements for CBD to succeed are:

- good (component) interface specifications;
- good composition guide for selecting the right components.

Furthermore, formal correctness and hence formality will be essential for CBD to succeed. Correctness underpins the semantics of interfaces, which in turn should provide a good composition guide.

We also believe that existing Formal Methods cannot provide the necessary formality for CBD, since they lack suitable semantics for components and their composition, and for their interfaces.

In this talk we outline our approach to CDB in Computational Logic, and we shall attempt to show that it has the potential to meet the requirements for the success of CBD.

In our work in formal program development in computational logic, see e.g. [5, 6, 3, 2]:

- components are specification frameworks;

  (these correspond to OOD frameworks in the CBD methodology *Catalysis* [1, 8, 4])
- we have correctness of components and their composition and reuse;
- specifications of components have formal semantics;

  (our approach thus can provide good component interfaces)
- correctness (based on *steadfastness*[7]) is *a priori*

  (our approach can therefore provide a good composition guide).

*I am currently visiting the Department of Computer Engineering, Mälardalen University, Västerås, Sweden. I wish to thank Professors Hans Hansson and Ivica Crnkovic for the invitation and for their hospitality.

References

[1] D.F. D'Souza and A.C. Wills. *Objects, Components, and Frameworks with UML: The Catalysis Approach*. Addison-Wesley, October 1998.

[2] J. Küster Filipe, K.-K. Lau, M. Ornaghi, and H. Yatsu. On Dynamic Aspects of OOD Frameworks in Component-based Software Development in Computational Logic. Extended abstract, to be presented at LOPSTR'99, 22-24 September 1999, Venice, Italy.

[3] J. Küster Filipe, K.-K. Lau, M. Ornaghi, and H. Yatsu, Intra- and Inter-OOD-Framework Interactions in Component-based Software Development in Computational Logic, in this volume.

[4] K.-K. Lau, S. Liu, M. Ornaghi, and A. Wills, Interacting Frameworks in *Catalysis*, in J. Staples, M. Hinchey and S. Liu, editors, *Proceedings of the Second IEEE International Conference on Formal Engineering Methods*, pages 110-119, IEEE Computer Society Press, 1998.

[5] K.-K. Lau and M. Ornaghi, On Specification and Correctness of OOD Frameworks in Computational Logic, in A. Brogi and P. Hill, editors, *Proceedings of the First International Workshop on Software Development in Computational Logic*, pages 59-75, September 1998, Pisa, Italy.

[6] K.-K. Lau and M. Ornaghi, OOD Frameworks in Component-based Software Development in Computational Logic, in P. Flener, editor, *Proceedings of the Eighth International Workshop on Logic-based Program Synthesis and Transformation*, *Lecture Notes in Computer Science* 1559:101-123, Springer-Verlag, 1999.

[7] K.-K. Lau, M. Ornaghi, and S.-Å. Tärnlund. Steadfast logic programs. *J. Logic Programming* 38(3):259-294, March 1999.

[8] K.-K. Lau, M. Ornaghi, and A. Wills. Frameworks in *Catalysis*: Pictorial notation and formal semantics. In M Hinchey and S. Liu, editors, *Proc. 1st IEEE Int. Conf. on Formal Engineering Methods*, pages 213-220, IEEE Computer Society, 1997.