Jacques Carette

Curriculum Vitae

Revised: February 03, 2010

Contact Information

Address: Department of Computing and Software
McMaster University
1280 Main Street West
Hamilton, Ontario L8S 4K1
Canada

E-mail: carette@mcmaster.ca
Web: http://www.cas.mcmaster.ca/~carette/
Phone: +1-905-525-9140, ext. 26869

Areas of Expertise

Mechanized Mathematics (symbolic computation, rigorous mathematical software, specification/computation environments, mathematical knowledge management).

Meta-programming (code generation, partial evaluation, domain specific languages, generalized interpreters).

Software Engineering (automated testing, static analysis, correct-by-construction techniques).

Education

Ph.D., Mathematics, Université de Paris-Sud, Paris, France, 1997.

M.Sc., Mathematics, Université de Montréal, Montréal, Québec, Canada, 1992.

BMath, Co-op Pure Mathematics and Computer Science, University of Waterloo, Waterloo, Ontario, Canada, 1990.

Employment

2008 – present: Associate Professor, Department of Computing and Software, McMaster University, Hamilton, Ontario, Canada.

2002 – 2008: Assistant Professor, Department of Computing and Software, McMaster University, Hamilton, Ontario, Canada.

1991 – 2002: Product Development Director and Sr. Architect, Maplesoft Inc., Waterloo, Ontario, Canada.

Selected Publications

1. Jacques Carette, Oleg Kiselyov, and Chung-chieh Shan. "Finally tagless, partially evaluated: Tagless staged interpreters for simpler typed languages." *Journal of Functional Programming*, 19(05) 509–543, 2009. *doi:10.1017/S0956796809007205*

2. Jacques Carette and William M. Farmer. "A Review of Mathematical Knowledge Management." In *Proceedings of CICM 2009*, LNCS 5625, 2009. Springer Verlag.

3. J. Carette and W. M. Farmer, "High-level theories", in: F. Wiedijk and J. Rubio, eds., *AISC/Calculemus/MKM 2008*, *Lecture Notes in Computer Science*, Vol. 5144, pp. 232–245, Springer-Verlag, 2008.

4. Jacques Carette and Oleg Kiselyov. "Multi-stage programming with Functors and Monads: eliminating abstraction overhead from generic code." *Science of Computer Programming. doi:10.1016/j.scico.2008.09.008* (27 pages)

5. Jacques Carette, Spencer Smith, John McCutchan, Christopher Anand, and Alexandre Korobkine. "Case Studies in Model Manipulation for Scientific Computing." In *Proceedings of AISC/MKM/Calculemus*, LNCS 5144, pages 24–37, 2008. Springer Verlag.

6. Jacques Carette. “A canonical form for some piecewise defined functions.” In *Proceedings of the 2007 International Symposium on Symbolic and Algebraic Computation (ISSAC)*, pages 77–84, New York, NY, USA, 2007. ACM Press.

7. Jacques Carette and Ryszard Janicki. “Computing properties of numerical imperative programs by symbolic computation.” *Fundamenta Informaticae*, 80, 1-3 (2007), 125–146.

8. Jacques Carette, William M. Farmer, and Volker Sorge. “A rational reconstruction of a system for experimental mathematics.” In *Proceedings of MKM/Calculemus 2007*, LNCS 4573, pages 13–26. Springer Verlag Berlin 2007.

9. Jacques Carette and *Michael Kucera*. “Partial Evaluation for Maple.” In *ACM SIGPLAN 2007 Workshop on Partial Evaluation and Program Manipulation*, 2007. pages 41–50.

10. Jacques Carette. “Gaussian Elimination: a case study in efficient genericity with MetaOCaml.” *Science of Computer Programming*, 62(1):3–24, 2006. Special Issue on the First MetaOCaml Workshop.

11. Jacques Carette and *Stephen Forrest*. “Mining Maple code for contracts.” In Silvio Ranise and Anna Bigatti, editors, *Proceedings of Calculemus 2006*, Electronic Notes in Theoretical Computer Science. Elsevier, 2006. (14 pages)

12. *Michael Kucera* and Jacques Carette. “Partial evaluation and residual theorems in computer algebra.” In Silvio Ranise and Anna Bigatti, editors, *Proceedings of Calculemus 2006*, Electronic Notes in Theoretical Computer Science. Elsevier, 2006. (14 pages)

13. Wolfram Kahl, Jacques Carette, and Xiaoheng Ji. “Bimonadic semantics for basic pattern matching calculi.” In Tarmo Uustalu, editor, *MPC*, volume 4014 of *Lecture Notes in Computer Science*, pages 253–273. Springer, 2006.

14. Jacques Carette and Oleg Kiselyov. “Multi-stage programming with Functors and Monads: eliminating abstraction overhead from generic code.” In *Generative Programming and Component-based Engineering GPCE*, pages 256–274, 2005.

15. Jacques Carette. “Understanding expression simplification.” In *Proceedings of the 2004 International Symposium on Symbolic and Algebraic Computation (ISSAC)*, pages 72–79, New York, NY, USA, 2004. ACM Press.

16. Sergei A. Abramov, Jacques Carette, Keith O. Geddes, and Ha Q. Le. “Telescoping in the context of symbolic summation in Maple”. *Journal of Symbolic Computation*, 38(4):1303–1326, October 2004.

17. J. Carette, W. M. Farmer, and J. Wajs, “Trustable communication between mathematics systems”, in: T. Hardin and R. Rioboo, eds., *Calculemus 2003*, pp. 58–68, Aracne, Rome, Italy, 2003.

Software Released

1. `MapleMix`. A partial evaluator for Maple. Originally released May 2009, updated November 2009. Available from `http://sourceforge.net/projects/maplemix/`.

2. Code for “finally tagless” method of writing interpreters, in both *Haskell* and *metaocaml*. Available from the author’s web site(s). Current version: July 2007.

3. `pa_monad`. Syntax extension for monadic “perform” notation for Objective Caml. Available on J. Carette’s web site and the *Caml Hump*. Joint work with Oleg Kiselyov and Lydia van Dijk. Releases June 2006, April, May and August 2007.

4. Upgrade of `maple.vim`, the syntax rules for *Maple* for the open source editor *vim*. Shipped as part of *vim*.

5. Code for a generative linear algebra library. Available from `www.metaocaml.org`. June 2005.

Unusual Contributions

Co-designed a novel degree program, *Software Engineering and Game Design*, the only one of its kind in Canada.