

Curriculum Vitae

James Noble

Full Name Robert James Noble.
Qualifications **B.Sc.(Hons) Ph.D. VUW**
Nationality New Zealand Citizen, British Citizen.
Date and place of birth 17 August 1967, London, England.

Address for correspondence

James Noble,
Computer Science,
Victoria University of Wellington,
P O Box 600 Wellington,
New Zealand.

Telephone: +64 4 463 6736
Facsimile: +64 4 463 5045
Internet Mail: kjx@mcs.vuw.ac.nz
WWW: <http://www.mcs.vuw.ac.nz/~kjx>

Present Position

I am currently engaged as Professor of Computer Science (Software Engineering) within the School of Engineering and Computer Science at Victoria University of Wellington, New Zealand.

Research Areas

I conduct research in the following areas of Software Engineering and Computer Science:

Software Design especially Object-Oriented and Aspect-Oriented approaches to design.
Programming Languages especially Ownership Types and Pluggable Type systems.
Design Patterns including Small Memory Systems; User Interaction; Agile Development.
Human Computer Interaction including Interface Design, Interaction Design, Usability.
Software Visualisation and **Visual Languages** including language design and analysis .
Development Methodologies especially Agile approaches involving users and customers.
Philosophy of Computer Science and Software Engineering, in particular semiotics of computation and postmodern approaches to software design.

Teaching Experience

Since 2000 I have developed and taught courses from introductory computer science to specialised international tutorials:

- Lead development of a innovative new first-year course on Software Modelling.
- Lead development of a two second-year courses on Software Design and Engineering
- Lead development of new third year courses on User Interface Design and Agile Methods
- Redeveloped a fourth year course on Object-Oriented Paradigms
- Taught international tutorials on Agile Software Engineering, Usage-Centred User Interface Design, Visual Languages, Software Visualisation, Semiotics, Design Patterns, Small Memory Systems, Postmodern Software Development.

Appointments Held

(Teaching appointments marked *)

Visiting Professor

Department of Computing, Imperial College, London
October 2009 – September 2012

* Professor

Computer Science, Victoria University of Wellington
December 2003 — Present

* Associate Professor

Computer Science, Victoria University of Wellington
January 2003 — November 2003

* Senior Lecturer

Computer Science, Victoria University of Wellington
January 2002 — December 2002

* Lecturer

Computer Science, Victoria University of Wellington
October 1999 — December 2001

* Research Scientist,

Microsoft Research Institute, Macquarie University,
July 1996 — October 1999.

* Postdoctoral Research Fellow,

Center for Object Technology, Applications, and Research,
Department of Computing Sciences, University of Technology, Sydney,
May 1995 — June 1996.

* Lecturer,

Department of Computing Sciences, University of Technology, Sydney,
August 1995 — November 1995.

* Assistant Lecturer,

Victoria University of Wellington,
August 1992 — January 1993.

* Tutor, Victoria University of Wellington,

1993 — 1994.

* Teaching Assistant, Victoria University of Wellington,

February 1989 — July 1992.

Programmer, Coopers & Lybrand,

November 1986 - November 1988

Scholastic and University Record

Ph.D., Victoria University of Wellington, 1996.

J.L. Stewart Scholarship, 1989.

IBM Postgraduate Scholarship, 1989 — 1992.

UGC Postgraduate Scholarship, 1989 — 1992.

B.Sc.(Hons) 1st class, Victoria University of Wellington, 1988.

B.Sc., Victoria University of Wellington, 1985 — 1987.

Prizes and Awards

VUW PGSA Award: Best Supervisor in the Faculty of Engineering 2010.

Asia-Pacific Software Engineering Conference “Best Paper Award” 2010.

Association for Computing Machinery (ACM) SIGPLAN “Most Influential OOPSLA Paper in the last 10 years” Award 2008

Australian Software Engineering Conference “Most Influential Paper in the last 10 years” Award 2008

VUW Research Excellence Award 2008

Neil B. Harrison Shepherding Award, Ninth European Conference on Pattern Languages of Program Design (EuroPLoP) 2004

Best Paper Award, Eighth European Conference on Pattern Languages of Program Design (EuroPLoP) 2003

Best Presentation Award, IEEE Symposium on Visual Languages 2000

Best Paper Award, Australian Software Engineering Conference 1998

External Research Support

Design patterns and ownership types for reusable program verification. James Noble, Sophia Drossopoulou. RSNZ Marsden Fund, 2011-2013. NZD 640,000

More Expressive Ownership Types. James Noble. Build IT Postdoctoral Fellowship. NZD 116,000, July 2008-2010

Scoped Memory Types for Safety Critical RTSJ. James Noble. IBM Real Time Innovation Award. USD 16,000. Dec 2007

Software Product and Process Improvement for New Zealand Industry. John Grundy (Science Leader). John Hosking, Emilla Mendes, Stephen MacDonell, James Noble (Key Researchers). FRST New Materials, Technologies and Services. October 2007—June 2011.

Practical Ownership Types for Object and Aspect Programs Sophia Drossopoulou, James Noble and Dave Clarke. EPSRC, GBP 52,000, 2006.

Integrated Ownership and Debugging in Eclipse. James Noble, Robert Biddle. IBM Eclipse Innovation Award. USD 15,000. 2005

Eclipse support for Ownership Types. James Noble, Robert Biddle. IBM Eclipse Innovation Award. USD 20,000. 2004

Learning the Aspects of Object-Oriented Programming. Robert Biddle, James Noble. IBM Eclipse Innovation Award. USD 20,000. 2004

Ownership types for Object- and Aspect-Oriented Programming. James Noble, David Clarke, Robert Biddle. Marsden Grant. NZD 327,678 2004-2006.

Domain Specific Software Tools. John Hosking, John Grundy, Ewan Tempero, Emilia Mendes, Anna Liu, James Noble, Mark Apperly, Robert Biddle, Robert Amor, Nianping Zhu. Public Good Science Fund. NZD 762,000. 2002-2005.

Aliasing and Ownership in Object-Oriented Systems. James Noble. Fast Start Marsden Grant NZD 100,000. 2002-2003.

Object-Oriented Frameworks. Robert Biddle, James Noble, and Information Power Ltd. Technology for Business Growth Grant NZD 50,000. 2002-2003.

Interactive Metrics Visualisation for User Interface Design. James Noble and Larry Constantine. ARC Small Grant, 1998. AUD 16 000.

External Service

Appointments

Panel member for Royal Society of New Zealand Marsden Fund Mathematical and Information Sciences (MIS) panel, 2006-2008.

Foundation Editor-In-Chief, Springer-Verlag Transactions on Pattern Languages of Programming. 2007-present

Editorial Board member of journals: IET Software, Systems Signs and Actions, Intl. Journal Agile and Extreme Software development..

Director of the Hillside Group, a US-based non-profit organization that promotes the study of design patterns worldwide, 2001-2008.

Invited Presentations / Keynotes

We Demand Rigidly Defined Areas of Doubt and Uncertainty. James Noble. Keynote presentation to the 5th Asian Workshop on Aspect-Oriented and Modular Software Development (AOAsia 2009), at ASE 2009, November 2009

The Myths of Object-Oriented Programming. James Noble. Invited presentation, ECOOP, Paphos. 2008.

We Demand Rigidly Defined Areas of Doubt and Uncertainty. James Noble. Keynote presentation to the ECOOP 2008 Workshop on Aspect Dependencies and Interactions, July 2008

The Lego Hypothesis. James Noble. Keynote presentation to QCon San Francisco, Nov 2007.

Roles and Relationships James Noble. Keynote presentation to ECOOP Workshop on Roles and Relationships, Berlin, July 2007.

The Lego Hypothesis. James Noble. Keynote presentation to OOP, Munich, Jan 2007.

Love in the Age of Software. James Noble and Robert Biddle. Keynote Presentation to XPDay, London, Nov 2006.

Postmodern Prospects for Conceptual Modelling. James Noble, Robert Biddle. Invited Presentation to Asian-Pacific Conference on Conceptual Modelling (APCCM), Hobart, Jan 2006.

The Lego Hypothesis. James Noble. Keynote presentation to JAOO 2004, Aarhus, Denmark, Sept 2004.

External Examiner for Masters & Doctoral Theses

Queensland University of Technology, 2010
Massey University 2009, 2010
The University of Auckland, 2008, 2009
The University of New South Wales, 2007
The University of Madeira, Portugal, 2006
Monash University, 2005
The University of Sheffield, 2005
Flinders University of South Australia, 2005
The University of Auckland 2004, 2003
Dublin City University, 2004
The Royal Melbourne Institute of Technology, 2002

Research Assessments

External Research Quality Evaluator, The Australian National University, 2010.
Reviewer for Canadian MITACS grant body, 2010.
Reviewer for ETH Zurich Research Commission, 2010
Reviewer for Hasler Stiftung, 2010
Reviewer for Hong Kong Research Grants Committee, 2009

Conference Committees

Object-Oriented Programming Systems, Languages, and Application (OOPSLA)

- Programme Committee OOPSLA 2001, 2002, 2005, 2006,2009
- Organising Committee OOPSLA 2002, 2005.
- Co-Chair, OOPSLA 2005 Onward! Stream
- Chair, OOPSLA 2002 Doctoral Symposium
- Doctoral Symposium Faculty Member 2001,2003,2006,2007

European Conference on Object-Oriented Programming (ECOOP)
2001, 2005, 2007, 2008, 2009. 2010, 2011

- Inaugural Summer School Chair 2008; Summer School Committee 2009, 2011
- Doctoral Symposium Academic Panel Member 2008.

International Symposium On Wikis (WikiSym)

- Steering Committee 2005-present
- Programme Chair WikiSym 2006
- Programme Committee WikiSym 2005-2008,2010.

Australian Software Engineering Conference (ASWEC)

- Programme co-Chair 2010, 2012

ACM Principles Of Programming Languages (POPL) 2007

Aspect Oriented Software Development (AOSD) 2008

ACM Hypertext 2007

Programme Chair for forUSE 2002.

Technology of Object-Oriented Languages and Systems (TOOLS Europe), 2010, 2011.

Technology of Object-Oriented Languages and Systems (TOOLS Pacific), 1997-2002.

- Program chair for TOOLS Pacific 2002.

Asia-Pacific Conference on Conceptual Modelling (ACCPM) 2009

Australian User Interface Conference 2000, 2002, 2004, 2005

Australian Computer Science Conference 2002, 2003, 2004.

Computing: Australian Theory Symposium (CATS) 2009.

Dynamic Language Symposium (DLS) Programme Chair 2009

European Conference on Pattern Languages of Programming (EuroPLOP) from 1998 to 2001.

- Focus Group Chair for EuroPLOP 2000.

European Symposium on Programming (ESOP) 2009

Asian-Pacific Pattern Languages of Programming Conference (KoalaPLoP) 2000-2002.

- Conference Chair 2001
- Program Chair 2002.

Visual Languages/Human Centric Communication (VL/HCC) 2006, 2007, 2008

Diagrams 2008

Software Engineering - SIENZ 2007 Research Stream Programme Panel

INTERACT'99, 2001.

Workshop Committees

ECOOP 2011 Intercontinental Workshop on Aliasing, Confinement, and Ownership in Object-Oriented Systems (IWACO): Twenty Years of Aliasing in Object-Oriented Languages. Nick Cameron, Dave Clarke, James Noble, Matthew Parkinson.

Dagstuhl 2010 Workshop on Relationships, Objects, Roles, and Queries in Modern Programming Languages, Stephanie Balzer, Gavin Bierman, Stephan Hermann, Stephen Nelson, Erik Meijer, James Noble, David Pearce, Friedrich Steimann, Frank Tip.

OOPSLA 2008 Relationships and Associations in Object-Oriented Languages. Stephen Nelson, Stephanie Balzer, Gavin Bierman, Erik Meijer, James Noble, David Pearce, Jiri Soukup, Frank Tip.

ECOOP 2007 International Workshop on Aliasing, Confinement, and Ownership in Object-

Oriented Systems (IWACO), Tobias Wrigstad, Dave Clarke, Sophia Drossopoulou, James Noble.

FOOL/WOOD Foundations of Object-Oriented Languages workshop and the Workshop on Object-Oriented Developments 2006

ECOOP FTfJP Workshop: Formal Techniques for Java-like Programs: 2003, 2004.

ECOOP 2003 Intercontinental Workshop on Aliasing, Confinement, and Ownership in Object-Oriented Systems (IWACO). Dave Clarke, Sophia Drossopoulou, James Noble

OOPSLA 2002 Workshop on Refactoring Design Patterns. Brian Foote, James Noble, Charles Weir.

ECOOP'99 International Workshop on Aliasing in Object-Oriented Systems. James Noble, Jan Vitek, Doug Lea, Paolo Sergio Almeida.

OOPSLA'98 Workshop on Prototype-Based Object-Oriented Programming. Antero Taivalsaari and James Noble.

ECOOP'97 Workshop on Prototype-Based Object-Oriented Programming. James Noble and Ivan Moore.

ECOOP'96 Workshop on Prototype-Based Object-Oriented Programming. Antero Taivalsaari and James Noble.

Journal and Conference Reviewing

IEEE Transactions on Software Engineering. 2007, 2008

IEEE Software 2008

Software—Practice & Experience 2008

ISSTA 2008

Journal of Visual Languages and Computation

MODELS 2008

Conference Tutorials

Customers are from Venus, Programmers are from Mars. Angela Martin, Robert Biddle, James Noble. XP 2006, Agile 2006, OOPSLA 2006, XP 2006, XPDay 2006. Agile 2007.

The Six Million Dollar Customer. Angela Martin, Robert Biddle, James Noble. OOPSLA 2005.

Semiotics of Visual Languages. James Noble, Robert Biddle. VL/HCC2004.

Object-Oriented Techniques That Really Work. Robert Biddle, James Noble, Ewan Tempero.. forUSE2002.

Patterns for Small Machines. Charles Weir and James Noble. OOPSLA 2000-2002, TOOLS Europe 2000, JAOO 2004

Playing with Patterns. Charles Weir and James Noble. OOPSLA'98, 99, 2000. TOOLS Europe 2000

Usage-Centered Design. Larry Constantine, Lucy Lockwood, and James Noble. OzCHI'98, OOPSLA'99, 2000. 2001. 2002, forUSE 2002.

Applying Design Patterns, John Potter and James Noble. TOOLS Pacific, Melbourne, 1997, ASWEC, Adelaide, 1998.

Industrial Courses

Object-Oriented Techniques That Really Work. Robert Biddle, James Noble, Ewan Tempero.
Presented to various New Zealand companies, 2000-2002.

Research and Publications

Book

1. *Small Memory Software: Patterns for Systems with Limited Memory*. James Noble and Charles Weir. Addison-Wesley, London. October 2000.

Edited Books

2. *Pattern Languages of Program Design 5*. Dragos Manolescu, Marcus Volter, and James Noble, editors. Addison-Wesley, 2006.
3. *Prototype-Based Programming: Concepts, Languages, Applications*. James Noble, Antero Taivalsaari, Ivan Moore, editors. Springer-Verlag, 1999.

Edited Journal Issue

4. Transactions on Pattern Languages of Programming I.. James Noble, Ralph Johnson, editors. Springer-Verlag Lecture Notes in Computer Science (LNCS 5770) 2010.
5. Special issue of Software—Practice and Experience on Aliasing in Object-Oriented Systems. James Noble and Doug Lea, editors. 31:1, 2001.

Edited Conference Proceedings

6. Proceedings of the 21st Australian Software Engineering Conference (ASWEC 2010). James Noble and Colin Fidge, Editors. IEEE Press, 2010.
7. Proceedings of the 5th Symposium on Dynamic languages (DLS 2009). James Noble, Editor. ACM Press, 2009
8. Proceedings of the 2006 International Symposium on Wikis (WikiSym 2006). Dirk Riehle, James Noble, Editors. ACM Press, 2006.
9. Proceedings of the Third Asian-Pacific Conference on Pattern Languages of Programs KoalaPloP 2002. James Noble and Paul Taylor, Editors. Volume 16 of *Conferences on Research and Practice in Information Technology*, Australian Computer Society, 2002.
10. Objects for Mobile, Web, and Internet Computation: Proceedings of the Fortieth International Conference on Technology of Object-Oriented Languages and Systems - TOOLS Pacific 2002, Sydney, Australia, 2002. James Noble and John Potter. Editors. Volume 10 of *Conferences in Research and Practice in Information Technology*.
11. Proceedings of the Second Asian-Pacific Conference on Pattern Languages of Programs KoalaPloP 2001. James Noble and Brian Wallis, Editors. Technical Report CS-TR-01-8, Computer Science, Victoria University of Wellington, New Zealand.

Refereed Book Chapters

12. An Ideal Customer: A Grounded Theory of Requirements Elicitation, Communication and Acceptance on Agile Projects. Angela Martin, Robert Biddle, James Noble. To appear in *Agile Software Development: Current Research and Future Directions*. Nils Brede Moe, Tore Dybå, Torgeir Dingsøy, Editors. Springer Verlag. 2010.
13. A Semiotic Model of User-Interface Metaphor. Pippin Barr, Robert Biddle, James Noble. Chapter in *Virtual, Distributed and Flexible Organisations*, Kluwer Academic Publishers, 2004.

14. Visual Program Visualisation. James Noble and Robert Biddle. Chapter 12, pages 359-388 in *Software Visualisation: From Theory to Practice*. Kang Zhang, editor. Kluwer. 2003
15. Visualising Objects: Abstraction, Encapsulation, Aliasing and Ownership. James Noble. In Chapter 1, pages 58-72 of *Software Visualisation, State of the Art Survey*. Springer-Verlag Lecture Notes in Computer Science (LNCS 2269). 2002.
16. The Object System Pattern. James Noble. Chapter 5, pages 53-72 in *Pattern Languages of Program Design 4*. Neil Harrison, Brian Foote and Hans Rohnert, editors. Addison-Wesley, 2000.
17. Basic Relationship Patterns. James Noble. Chapter 6, pages 73-89 in *Pattern Languages of Program Design 4*. Neil Harrison, Brian Foote and Hans Rohnert, editors. Addison-Wesley, 2000.
18. High-level and Process Patterns from the Memory Protection Society. James Noble and Charles Weir. Chapter 12, pages 221-238 in *Pattern Languages of Program Design 4*. Neil Harrison, Brian Foote and Hans Rohnert, editors. Addison-Wesley, 2000.
19. Prototype-Based Programming for Abstract Program Visualisation. James Noble. In *Prototype-Based Programming: Concepts, Languages, Applications*. James Noble, Antero Taivalsaari, Ivan Moore, editors. Springer-Verlag, 1999.

Refereed Journal Articles

20. Developing a Grounded Theory to Explain the Practices of Self-Organizing Agile Teams. Rashina Hoda, James Noble, Stuart Marshall. *Empirical Software Engineering*. To appear 2011.
21. The Impact of Inadequate Customer Involvement on Self-Organizing Agile Teams. Rashina Hoda, James Noble, Stuart Marshall. *Information Science and Technology*. Vol 53 Issue 5 pp.521-534. 2011.
22. Abstraction and Activity in Computer Mediated Music Production. Matthew Duignan, James Noble, Robert Biddle. In the *Computer Music Journal*, 34:4, Winter 2010.
23. JavaCOP: Declarative Pluggable Types for Java. Shane Markstrum, Daniel Marino, Matthew Esquivel, Todd Millstein, Chris Andreae, James Noble. In the *ACM Transactions on Programming Languages and Systems (TOPLAS)* 23:2, Article 4, 2010.
24. Implicit ownership types for memory management. Tian Zhao, Jason Baker, James Hunt, James Noble, Jan Vitek. *Science of Computer Programming*. 71(3): pp213-241. 2008
25. Point/Counterpoint (Every Good Designer Uses Patterns). Peter Sommerlad, James Noble. *IEEE Software* 24(4) 2008. pp68-71.
26. Scoped types and aspects for real-time Java memory management. Chris Andreae, Yvonne Coady, Celina Gibbs, James Noble, Jan Vitek, Tian Zhao. *Real-Time Systems*. Vol 37 No 1, pp 1-44. Oct 2007.
27. Videogame Values: Human-Computer Interaction and Games. Barr, P., Biddle, R., and Noble, J. *Interacting with Computers: HCI Issues in Computer Games*, 19(2): pp180-195, 2007.
28. Featherweight Generic Confinement. Alex Potanin, James Noble, Dave Clarke, and Robert Biddle. *Journal of Functional Programming*. Vol 16 Issue 6, pp 793-811. Sep 2006.
29. Patterns as Signs: A Semiotics of Object-Oriented Design Patterns. James Noble, Robert Biddle, Ewan Tempero. *Systems, Signs, and Actions*. 2:1, pp3-40, 2006.
30. Exclusion and Potential Concurrency for Composite Objects. Abdelsalam Shanneb, John Potter and James Noble. *Science of Computer Programming*. 38:3, 344-365, Dec 2005

31. Scale-Free Geometry of Object-Oriented Programs. Alex Potanin, James Noble, Marcus Freen, Robert Biddle. *Communications of the ACM*. 48:5, 99-103, May 2005
32. Checking Ownership and Confinement. Alex Potanin and James Noble. *Concurrency and Computation, Practice and Experience* Volume 16, Issue 7, pp671 - 687, 2004.
33. Scalable Visualisations of Object-Oriented Systems using Ownership Trees. Trent Hill, James Noble, and John Potter. In the *Journal of Visual Languages and Computation*, vol 13 no 3 June 2002.
34. Aliasing in Object-Oriented Systems: Introduction to the Special Issue. James Noble and Doug Lea. *Software—Practice and Experience*. 31:1, p.1565, 2001. Reprinted in *Software Focus*. Wiley. Jan 2001.
35. Arguments and Results. James Noble. *The Computer Journal*. vol 46 no 6. pp.439-450. 2001.
36. Object Oriented Program Visualisation in Tarraingím. James Noble, Lindsay Groves, and Robert Biddle. *Australian Computer Journal*, Special Issue on Software Visualisation. 27(4), November 1995.
37. Tarraingím— A Program Animation Environment. R. J. Noble and L. J. Groves. *New Zealand Journal of Computing*, 4(1), December 1992.

Refereed Conference Papers

Conferences marked ** are highly prestigious: accepting less than 1/5 of submitted papers (24 papers)
Papers marked # won best paper or best presentation awards. (4 papers)

38. Effective Communication in Distributed Agile Software Development Teams. Siva Dorairaj, Petra Malik, James Noble. In the Conference on Agile Processes and eXtreme Programming in Software Engineering, XP 2011, pp.102-166
39. Supporting Self-Organizing Agile Teams: What's Senior Management Got To Do With It. Rashina Hoda, Stuart Marshall, James Noble. In the Conference on Agile Processes and eXtreme Programming in Software Engineering, XP 2011.
40. Bridging Cultural Differences — A Grounded Theory. Siva Dorairaj, Petra Malik, James Noble. In the Indian Software Engineering Conference (ISEC) 2011 p3-10.
41. # Qualitas Corpus: A Curated Collection of Java Code for Empirical Studies. Ewan Tempero, Craig Anslow, Jens Dietrich, Ted Han, Jing Li, Markus Lumpe, Hayden Melton and James Noble. In the Asia Pacific Software Engineering Conference (APSEC2010), December 2010
42. ** Tribal Ownership. Nicholas Cameron, James Noble, Tobias Wrigstad. In the ACM Conference on Object-Oriented Programming, Systems, Languages and Applications (OOPSLA) 2010, p618-633.
43. ** Agility in Context. Rashina Hoda, Philippe Kruchten, James Noble, Stuart Marshall. In the ACM Conference on Object-Oriented Programming, Systems, Languages and Applications (OOPSLA) 2010, p74-88.
44. ** Organizing Self-Organizing Teams. Rashina Hoda, James Noble, Stuart Marshall. To appear in the International Conference on Software Engineering (ICSE) 2010, p285-294.
45. An empirical study of overriding in open source Java. Ewan D. Tempero, Steve Counsell, James Noble. In Australian Computer Science Conference (ACSC) 2011. Pp366-345.
46. Craig Anslow, Stuart Marshall, James Noble, Ewan D. Tempero, Robert Biddle: User Evaluation of Polymetric views using a large visualization wall. ACM Symposium on Software Visualisation (SOFTVIS) 2010, p25-34.

47. Haowei Ruan, Craig Anslow, Stuart Marshall, James Noble. Exploring the Inventor's Paradox: Applying Jigsaw to Software Visualization. ACM Symposium on Software Visualisation (SOFTVIS) 2010, p83-92.
48. Encoding Ownership Types in Java. Nicholas Cameron, James Noble. In Technology of Object-Oriented Languages and Systems (TOOLS 48) 2010, p271-290.
49. Understanding the Impact of Collection Contracts on Design. Stephen Nelson, David Pearce, James Noble. In Technology of Object-Oriented Languages and Systems (TOOLS 48) 2010, p61-78.
50. An Empirical Study of Overriding in Open Source Java. Ewan Tempero, Steve Counsell and James Noble. In Australasian Computer Science Conference (ACSC) 2010.
51. Agile Undercover: When Customers Don't Collaborate. Rashina Hoda, James Noble and Stuart Marshall. In the Conference on Agile Processes and eXtreme Programming in Software Engineering, XP 2010, 73-87.
52. What Language Does Agile Speak?. Rashina Hoda, James Noble and Stuart Marshall. To appear in the Conference on Agile Processes and eXtreme Programming in Software Engineering, XP 2010, p387-388.
53. Understanding the Importance of Trust in Distributed Agile Projects: A Practical Perspective. Siva Dorairaj, James Noble and Petra Malik. In the Conference on Agile Processes and eXtreme Programming in Software Engineering, XP 2010, p172-177.
54. The XP Customer Team: A Grounded Theory. Angela Martin, Robert Biddle, James Noble. In the Agile Development Conference (AGILE) 2009.
55. XP Customer Practices: A Grounded Theory. Angela Martin, Robert Biddle, James Noble. In the Agile Development Conference (AGILE) 2009.
56. ** The Myths of Object-Orientation. James Noble. In the Proceedings of the European Conference on Object-Oriented Programming (ECOOP) 2009
57. Negotiating Contracts for Agile Projects: A Practical Perspective. Rashina Hoda, James Noble and Stuart Marshall. In the Conference on Agile Processes and eXtreme Programming in Software Engineering, XP 2009.
58. Keith Cassell, Peter Andreae, Lindsay Groves, James Noble: Towards Automating Class-Splitting Using Betweenness Clustering. ASE 2009: 595-599
59. ** Multiple Dispatch in Practice. Radu Muschevici, Alex Potanin, Ewan Tempero, and James Noble. In the ACM Conference on Object-Oriented Programming, Systems, Languages and Applications (OOPSLA) 2008.
60. ** Caching and Incrementalization for the Java Query Language. Darren Willis, David J. Pearce, and James Noble. In the ACM Conference on Object-Oriented Programming, Systems, Languages and Applications (OOPSLA) 2008.
61. ** UpgradeJ: Incremental Typechecking for Class Upgrades. Gavin Bierman, Matthew Parkinson, James Noble. . In Proceedings of the European Conference on Object-Oriented Programming (ECOOP) 2008 pp235-295.
62. ** How do Java Programs use Inheritance? Ewan Tempero. James Noble, Hayden Melton. In Proceedings of the European Conference on Object-Oriented Programming (ECOOP) 2008, pp667-691.
63. A Qualitative Study of Culture and Persuasion in a Smoking Cessation Game. Rilla Khaled, Ronald Fischer, James Noble, Robert Biddle. PERSUASIVE 2008. pp224-236.
64. Patterns for Program Query Optimisation. David J. Pearce, James Noble. To appear in the Proceedings of the Conference on Pattern Languages of Programs (PLOP) 2008.

65. Exception Handling Bug Patterns in Aspect Oriented Programs. Roberta Coelho, Awais Rashid, Uira Kulesza, Arndt von Staa, Carlos Lucena, James Noble To appear in the Proceedings of the Conference on Pattern Languages of Programs (PLOP) 2008.
66. **Multiple Ownership. Nicholas Cameron, Sophia Drossopoulou,, James Noble, Matthew Smith. In ACM Conference on Object-Oriented Programming, Systems, Languages and Applications (OOPSLA) 2007. pp441-460.
67. **MAO; Ownership and Effects for more Efficient Reasoning about Aspects. Curtis Clifton, Gary T. Leavens, James Noble. In Proceedings of the European Conference on Object-Oriented Programming (ECOOP).2007, pp451-475.
68. Fine Tuning the Persuasion in Persuasive Games. Rilla Khaled, Pippin Barr, James Noble, Ronald Fischer, Robert Biddle . PERSUASIVE 2007. pp36-47.
69. Agile Development Iterations and UI Design. Jennifer Ferreira, James Noble, Robert Biddle. Agile 2007, pp50-58.
70. Up-Front Interaction Design in Agile Development. Jennifer Ferreira, James Noble, Robert Biddle. XP 2007. pp9-16
71. Dynamic Ownership in a Dynamic Language. Donald Gordon and James Noble. Dynamic Languages Symposium (DLS2007), pp9-16.
72. Tribe: a simple virtual class calculus . Dave Clarke, Sophia Drossopoulou, James Noble, Tobias Wrigstad.. AOSD 2007. pp121-134
73. Problem Frame Patterns. James Noble, Rebecca Wirfs-Brock, Paul Taylor. In the Proceedings of the Conference on Pattern Languages of Programs (PLOP) 2006.
74. Programmers are from Mars: Customers are from Venus: A practical guide to working with customers on XP Projects. Angela Martin, James Noble, Robert Biddle. In the Proceedings of the Conference on Pattern Languages of Programs (PLOP) 2006.
75. ** Generic Ownership for Generic Java. Alex Potanin, James Noble, Dave Clarke, Robert Biddle. ACM Conference on Object-Oriented Programming, Systems, Languages and Applications (OOPSLA). Portland, 2006. pp311-324.
76. ** A Framework for Implementing Pluggable Type Systems. Chris Andreae, James Noble, Shane Markstrum, Todd Millstein. ACM Conference on Object-Oriented Programming, Systems, Languages and Applications (OOPSLA). Portland, 2006. pp57-74.
77. ** Understanding the Shape of Java Software. Gareth Baxter, Marcus Frean, Hayden Melton, James Noble, Mark Rickerby, Hayden Smith, Ewan Tempero, Matt Visser. . ACM Conference on Object-Oriented Programming, Systems, Languages and Applications (OOPSLA). Portland, 2006. pp397-412.
78. Changing the Virtual Self: The Avatar Transformation Activity in Popular Games. Barr, P., Brown, J., Biddle, R., Khaled, R., and Noble, J, Joint International Conference on CyberGames and Interactive Entertainment, 2006.
79. Patterns for Relationship Aspects. David Pearce and James Noble. In Proceedings of European Conference on Pattern Languages of Programs. Irsee, Germany, 2006
80. ** STARS: Scoped Types and Aspects for Real-Time Systems. Chris Andreae, Yvonne Coady, Celina Gibbs, James Noble, Jan Vitek, Tian Zhao. In Proceedings of the European Conference on Object-Oriented Programming (ECOOP). Nantes, 2006.
81. ** Efficient Object Querying for Java. Darren Willis, David Pearce. James Noble. In Proceedings of the European Conference on Object-Oriented Programming (ECOOP). Nantes, 2006.

82. ** Relationship Aspects. David Peace and James Noble. In Proceedings of Aspect-Oriented Software Development (AOSD). Bonn, 2006.
83. Investigating Social Software as Persuasive Technology, Rilla Khaled, Pippin Barr, James Noble, Robert Biddle, in Proceedings of First International Conference on Persuasive Technology for Human Well-Being (Persuasive06), Eindhoven, Netherlands, 2006.
84. Our Place or Mine?: Exploration into Collectivism-Focused Persuasive Technology Design, Rilla Khaled, Pippin Barr, James Noble, Robert Biddle, in Proceedings of First International Conference on Persuasive Technology for Human Well-Being (Persuasive06), Eindhoven, Netherlands, 2006.
85. Postmodern Prospects for Conceptual Modelling. James Noble and Robert Biddle. Invited Presentation in Proceedings of the Asian-Pacific Conference on Conceptual Modelling (APCCM). Hobart. pp11-20. 2006
86. From Pushing Buttons to Play and Progress: Value and Interaction in Fable. Pippin Barr, James Noble, Robert Biddle, Rilla Khaled. In Proc. Seventh Australasian User Interface Conference (AUIC), Hobart, Australia. CRPIT, 50. Piekarski, W., Ed. ACS. 61-68. 2006.
87. A Case for Iconic Icons. Jennifer Ferreira, James Noble, Robert Biddle. In Proc. Seventh Australasian User Interface Conference (AUIC), Hobart, Australia. CRPIT, 50. Piekarski, W., Ed. ACS. 97-100. 2006
88. Persuasive Interaction for Collectivist Cultures. Rilla Khaled, Robert Biddle, James Noble, Pippin Barr, Ron Fischer. In Proc. Seventh Australasian User Interface Conference (AUIC), Hobart, Australia. CRPIT, 50. Piekarski, W., Ed. ACS. 73-80. 2006
89. **Efficient Multimethods in a Single-Dispatch Language. Brian Foote, Ralph Johnson, James Noble. In Proceedings of the European Conference on Object Oriented Programming (ECOOP). July 2005.
90. A Taxonomy of Sequencer User-Interfaces, Matthew Duignan, James Noble, Robert Biddle. In Proceedings of the International Computer Music Conference (ICMC). 2005.
91. Oppositional Play: Gathering Negative Evidence for Computer Game Values. Barr, P., Marsen, S., Noble, J., and Biddle, R. In Proceedings of the Second Australasian Conference on Interactive Entertainment (IE2005), 2005.
92. amethodology.. Charles Weir, James Noble. In Proceedings of the Tenth European Conference on Pattern Languages of Programs. Irsee, Germany, 2005.
93. Clicki: A framework for light-weight web-based visual applications. Donald Gordon, James Noble, Robert Biddle. In Proceedings of Sixth Australasian User Interface Conference (AUIC). Newcastle, Australia, January 2005.
94. The Semiotics of User Interface Redesign. Jennifer Ferreira, Pippin Barr, James Noble. In Proceedings of Sixth Australasian User Interface Conference (AUIC). Newcastle, Australia, January 2005.
95. A Practical Application of a Taxonomy of User-Interface Metaphor. Pippin Barr, Rilla Khaled, James Noble, Robert Biddle. In Proceedings of Sixth Australasian User Interface Conference (AUIC). Newcastle, Australia, January 2005.
96. Using a System of Tutorials and Groups to Teach User Interface Design. Wright, T., Noble, J. and Marshall, S. In Proc. Seventh Australasian Computing Education Conference (ACE2005), Newcastle, Australia. CRPIT, 42. Young, A. and Tolhurst, D., Eds., ACS. p187-19. January 2005.
97. Scoped Types for Real-Time Java. Tian Zhao, James Noble, and Jan Vitek. In The 25th IEEE Real-Time Systems Symposium (RTSS 2004). Lisbon, Portugal, December 2004.

98. Notes on Notes on Postmodern Programming. James Noble and Robert Biddle. In Proceedings of the Onward! stream at the Conference on Object-Oriented Programming, Systems, Languages and Applications (OOPSLA). Vancouver, October, 2004.
99. My Friend the Customer. Charles Weir, James Noble, Angela Martin, Robert Biddle. In Proceedings of the Ninth European Conference on Pattern Languages of Programs. Irsee, Germany, 2004.
100. Metaphorscape. Patterns for XP System Metaphors. Rilla Khaled, Pippin Barr, James Noble, Robert Biddle. In Proceedings of the Ninth European Conference on Pattern Languages of Programs. Irsee, Germany, 2004.
101. Metaphors for Electronic Music Production in *Reason* and *Live*. Matthew Duignan, Pippin Barr, Robert Biddle. In Proceedings of the 6th Asian Pacific Conference on Computer Human Interaction (APCHI 2004), pages 111-120. IEEE Computer Society. Springer-Verlag. 2004.
102. Towards a General Model for Assisting Navigation. Mike McGavin, James Noble, Robert Biddle and Judy Brown. In Proceedings of the 6th Asian Pacific Conference on Computer Human Interaction (APCHI 2004), pages 262-271. IEEE Computer Society. Springer-Verlag. 2004.
103. Snapshot Query-Based Debugging. Alex Potanin, James Noble, Robert Biddle. In the Australasian Software Engineering Conference (ASWEC 2004). Melbourne, Australia. IEEE CS Press. April 2004.
104. XML Database Support for Program Trace Visualisation. Craig Anslow, Stuart Marshall, Robert Biddle, James Noble, and Kirk Jackson. Australasian Symposium on Information Visualisation (INVIS 2004). Neville Churcher and Clare Churcher (eds). January 2004.
105. Delegation Diagrams: Visual Support for the Development of Object-Oriented Designs. Ewan Tempero, James Noble, Robert Biddle. In Proc. Fifth Australasian User Interface Conference (AUIC2004), Dunedin, New Zealand. CRPIT, 28. Cockburn, A., Ed. ACS. 83-89. 2004.
106. A Web User Interface For An Interactive Software Repository. Stuart Marshall and Robert Biddle and James Noble. In Proc. Fifth Australasian User Interface Conference (AUIC2004), Dunedin, New Zealand. CRPIT, 28. Cockburn, A., Ed. ACS. 57-64. 2004.
107. Less Extreme Programming. James Noble, Stuart Marshall, Stephen Marshall, Robert Biddle. In Proc. Sixth Australasian Computing Education Conference (ACE2004), Dunedin, New Zealand. CRPIT, 30. Lister, R. and Young, A. L., Eds., ACS. 217-226. 2004
108. ** Saving the World from Bad Beans: Deployment-time Confinement Checking. Dave Clarke, Michael Richmond, James Noble. To Appear in the ACM Conference on Object-oriented Programming, Languages, Systems and Applications (OOPSLA), October, 2003. Anaheim.
109. No Name: Notes on Reuse. Robert Biddle, Angela Martin, James Noble and Ewan Tempero. To Appear In Proceedings of the Onward! stream at the Conference on Object-Oriented Programming, Systems, Languages and Applications (OOPSLA). Anaheim, October, 2003.
110. Use Case Analysis with Narrative Semiotics. Sky Marsen, Robert Biddle, James Noble. In Proceedings of the Australian Conference on Information Systems (ACIS), Perth, 2003.
111. A technology for lightweight web-based visual applications. Donald Gordon, Robert Biddle, James Noble, and Ewan Tempero. To Appear in Margaret Burnett and John

- Grundy, editors, IEEE Symposium on Visual and Multimedia Software Engineering, Auckland, New Zealand, 2003.
112. # The Hitch-Hiker's Guide to Google. Charles Weir and James Noble. In Proceedings of the Eighth European Conference on Pattern Languages of Program Design (EuroPLOP). Irsee. Universitäts Verlag Konstanz. 2003.
 113. A Window in Your Pocket. Charles Weir, James Noble. In Proceedings of the Eighth European Conference on Pattern Languages of Program Design (EuroPLOP). Irsee. Universitäts Verlag Konstanz. 2003.
 114. Interface Ontology: Creating a World for Computer Interfaces. Pippin Barr, Robert Biddle, James Noble. In Proceedings of the Eighth European Conference on Pattern Languages of Programs. Irsee, Germany.2003
 115. Software Visualisation of Java Programs in InspectJ. Rilla Khaled, James Noble, Robert Biddle. In Proceedings of the Eighth European Conference on Pattern Languages of Programs. Irsee, Germany. 2003
 116. Being Jane Malkovich: a Look into the World of an XP Customer. Angela Martin, James Noble, Robert Biddle. In Proceedings of the Fourth International Conference on eXtreme Programming and Agile Processes in Software Engineering, Lecture Notes in Computer Science, Springer-Verlag. 2003
 117. InspectJ: Program monitoring for visualisation using AspectJ. Rilla Khaled, James Noble, and Robert Biddle. In Michael Oudshoorn, editor, Proceedings of the 26th Australasian Computer Science Conference, Conferences in Research and Practice in Information Technology, Vol 16, Adelaide, South Australia, Australian Computer Society 2003.
 118. Spreadsheet structure inspection using low level access and visualization. Daniel Ballinger, Robert Biddle, and James Noble. In Robert Biddle and Bruce Thomas, editors, Proceedings of the 4th Australasian User Interface Conference, Conferences in Research and Practice in Information Technology, Vol 18 , Adelaide, South Australia. Australian Computer Society. 2003.
 119. Icons R icons: User interface icons, metaphor and metonymy. Pippin Barr, Robert Biddle, and James Noble. In Robert Biddle and Bruce Thomas, editors, Proceedings of the 4th Australasian User Interface Conference, Conferences in Research and Practice in Information Technology, Vol 18, Adelaide, South Australia. Australian Computer Society 2003.
 120. A lightweight web based case tool for UML class diagrams. Daniel Mackay, Robert Biddle, and James Noble. In Robert Biddle and Bruce Thomas, editors, Proceedings of the 4th Australasian User Interface Conference, Conferences in Research and Practice in Information Technology, Vol 18, Adelaide, South Australia, Australian Computer Society 2003.
 121. Spreadsheet visualisation to improve end-user understanding. Daniel Ballinger, Robert Biddle, and James Noble. In Tim Pattison and Bruce Thomas, editors, Proceedings of the Australasian Symposium on Information Visualisation, Conferences in Research and Practice in Information Technology, Vol 24, Adelaide, South Australia, Australian Computer Society 2003.
 122. A case study of visualisation using AspectJ. Rilla Khaled, James Noble, and Robert Biddle. In Tim Pattison and Bruce Thomas, editors, Proceedings of the Australasian Symposium on Information Visualisation, Conferences in Research and Practice in Information Technology, Vol 24, Adelaide, South Australia, Australian Computer Society 2003.

123. Teaching the evaluation of object-oriented designs. Robert Biddle, James Noble, and Ewan Tempero. In Tony Greening and Raymond Lister, editors, Proceedings of the 5th Australasian Computing Education Conference, Conferences in Research and Practice in Information Technology, Vol 20, Adelaide, South Australian Computer Society 2003.
124. A Lightweight Case Tool for UML Sequence Diagrams. Rilla Khaled, Dan Mackay, Robert Biddle, James Noble, Ewan Tempero. SIGCHINZ Workshop on Human-Computer Interaction. 2002.
125. A Taxonomy of User-Interface Metaphors. Pippin Barr, Robert Biddle, James Noble. SIGCHINZ Workshop on Human-Computer Interaction. 2002.
126. Notes on Postmodern Programming. James Noble and Robert Biddle. In Proceedings of the Onward! stream at the Conference on Object-Oriented Programming, Systems, Languages and Applications (OOPSLA). Seattle, October, 2002.
127. Patterns for Self. James Noble and Brian Foote. To appear in Proceedings of the Third Asian-Pacific Conference on Pattern Languages of Programs (KoalaPLoP). James Noble and Paul Taylor, Editors. Australian Computer Society, 2002.
128. Essential Use Case Bodies. Robert Biddle, James Noble, Ewan Tempero. To appear in Proceedings of the Third Asian-Pacific Conference on Pattern Languages of Programs (KoalaPLoP). James Noble and Paul Taylor, Editors. Australian Computer Society, 2002.
129. ** Patterns as Signs. James Noble, Robert Biddle. In Proceedings of the European Conference on Object Oriented Programming, Spain. (ECOOP). pp368-391. Springer-Verlag, 2002.
130. Supporting Reusable Use Cases. Robert Biddle, James Noble, Ewan Tempero. In Proceedings of the International Conference on Software Reuse (ICSR). Austin. pp 210-226 Springer-Verlag 2002
131. Sokoban: A System Object Case Study. Robert Biddle, James Noble, Ewan Tempero. In Proceedings of Technology of Object-Oriented Languages and Systems (TOOLS 37) Sydney. pp103-114. Australian Computer Society. 2002.
132. Reflections on CRC Cards and OO Design. Robert Biddle, James Noble, Ewan Tempero. In Proceedings of Technology of Object-Oriented Languages and Systems (TOOLS 37) Sydney. pp201-206. Australian Computer Society. 2002.
133. Metaphor and Metonymy in Object-Oriented Design Patterns. James Noble, Robert Biddle and Ewan Tempero. In Proceedings of the Australian Computer Science Conference (ACSC), Melbourne Australia, pp187-196. 2002.
134. Essential Use Cases and Responsibility in Object-Oriented Development. Robert Biddle, James Noble, and Ewan Tempero. In Proceedings of the Australian Computer Science Conference (ACSC), Melbourne Australia, pp 7-16. 2002.
135. Program Visualisation for Visual Programs. James Noble and Robert Biddle. In Proceedings of the Australian User Interfaces Conference (AUIC), Melbourne, Australia, pp 29-38. 2002.
136. Visualising 1,051 Visual Programs: Module Choice and Layout In Nord Modular Patch Language. James Noble and Robert Biddle. In Proceedings of INVIS 2001, Sydney, Australia, pp 121-128. 2001.
137. Role-play and Use Case Cards for Requirements Review. Robert Biddle, James Noble, Ewan Tempero. In Proceedings of the Australian Conference on Information Systems (ACIS), Coffs Harbour, Australia, 2001.
138. ** Simple Ownership Types for Object Containment. David Clarke, James Noble, John Potter. In Proceedings of the European Conference on Object Oriented Programming, Brussels. (ECOOP) pp 2-27. Springer-Verlag, 2001.

139. ** Capabilities for Sharing: A Generalization of Uniqueness and Read-Only. John Boyland, James Noble, William Retert. In Proceedings of the European Conference on Object Oriented Programming, Brussels. (ECOOP) pp 53-76 Springer-Verlag, 2001.
140. Need to Know: Patterns for Coupling Contexts and Components. James Noble. In Proceedings of the Second Australasian Conference on Pattern Languages for Program Design (KoalaPLOP'01). pp 181-195. 2001
141. Patterns for Essential Use Cases. Robert Biddle, James Noble, Ewan Tempero. In Proceedings of the Second Australasian Conference on Pattern Languages for Program Design (KoalaPLOP'01) pp 196-206. 2001.
142. Reflections on Remote Reflection. Michael Richmond and James Noble. In Proceedings of the Australian Computer Science Conference (ACSC). Gold Coast, January 2001. IEEE Computer Society Press.
143. ** Exclusion for Composite Objects. James Noble, David Holmes, John Potter. In Proceedings of the Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA) Minneapolis. pp 13-28. ACM Press, 2000.
144. Scalable Visualisations with Ownership Trees. Trent Hill, James Noble, John Potter. In Proceedings of Technology of Object-Oriented Languages and Systems (TOOLS 37) Sydney. IEEE Computer Society Press, 2000.
145. # Visualising the Structure of Object-Oriented Systems. Trent Hill, James Noble, John Potter. In Proceedings of the IEEE Symposium on Visual Languages (VL). Seattle. pp 191-198 IEEE Computer Society Press, 2000.
146. Small User Interfaces. James Noble and Charles Weir. In Proceedings of the European Conference on Pattern Languages of Program Design, Irsee. pp 327-343. Universitäts Verlag Konstanz 2000.
147. Process Patterns for Small Systems. Charles Weir and James Noble. In Proceedings of the European Conference on Pattern Languages of Program Design. Irsee. pp 344-356. Universitäts Verlag Konstanz. 2000.
148. Iterators and Encapsulation. James Noble. In Proceedings of Technology of Object-Oriented Languages and Systems (TOOLS 33) Mont-St-Michel. pp 431-442. IEEE Computer Society Press, 2000.
149. Natural Creation. James Noble. In Proceedings of KoalaPLOP 2000, The First Asian Pacific Conference on Patterns Languages of Programs. Melbourne, May 2000.
150. Small Interfaces. James Noble and Charles Weir. In Proceedings of KoalaPLOP 2000, The First Asian Pacific Conference on Patterns Languages of Programs. Melbourne, May 2000.
151. Conglomeration: Realising Aliasing Protection. John Potter and James Noble. In Proceedings of the Australian Computer Science Conference (ACSC). Canberra, January 2000. IEEE Computer Society Press.
152. Natural Creation: A Composite Pattern for Creating Objects. James Noble. In Proceedings of Technology of Object-Oriented Languages and Systems (TOOLS 32) Melbourne, pp 78-88. IEEE Computer Society Press, 1999.
153. Object Ownership for Dynamic Alias Protection. In Proceedings of Technology of Object-Oriented Languages and Systems (TOOLS 32) Melbourne, pp 176-187. IEEE Computer Society Press, 1999.
154. Using Secondary Storage. James Noble and Charles Weir. In Proceedings of the Fourth European Conference on Pattern Languages of Programming and Computing. Irsee, Germany, 1999.

155. Process Patterns for Personal Practice. Charles Weir and James Noble. In Proceedings of the Fourth European Conference on Pattern Languages of Programming and Computing. Irsee, Germany, 1999.
156. ** Flexible Alias Protection. James Noble, Jan Vitek, John Potter. In Proceedings of the European Conference on Object Oriented Programming, Brussels. (ECOOP) Springer-Verlag, 1998.
157. ** Ownership Types for Flexible Alias Protection. David Clarke, John Potter, James Noble. In Proceedings of the Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA) Vancouver. ACM Press, 1998.
158. Objects and Constraints. James Noble. In Proceedings of Technology of Object-Oriented Languages and Systems (TOOLS Pacific), Melbourne, IEEE Computer Society Press, 1998.
159. Towards a Pattern Language for Object-Oriented Design. James Noble. In Proceedings of Technology of Object-Oriented Languages and Systems (TOOLS Pacific), Melbourne, IEEE Computer Society Press, 1998.
160. Integrating Metric Visualisation into a User-Interface Builder. James Noble. In Proceedings of OzCHI'98.
161. # The Ins and Outs of Objects. John Potter, James Noble, David Clarke. In Proceedings of the Australian Software Engineering Conference (ASWEC), Adelaide, IEEE Computer Society Press, 1998.
162. Classifying Relationships between Object-Oriented Design Patterns. James Noble. In Proceedings of the Australian Software Engineering Conference (ASWEC), Adelaide, IEEE Computer Society Press, 1998.
163. A Mode System for Flexible Alias Protection. John Potter, David Clarke, James Noble. In Proceedings of Formal Methods Pacific, Canberra, 1998.
164. Patterns for Small Machines. James Noble and Charles Weir. In Proceedings of the European Conference on Pattern Languages of Program Design, Irsee, Germany 1998. Universitäts Verlag Konstanz.
165. The Object System Pattern. James Noble. In Proceedings of the European Conference on Pattern Languages of Program Design, Irsee, Germany 1998. Universitäts Verlag Konstanz.
166. The Evolution of Tarrainím. James Noble, Robert Biddle, and Lindsay Groves. In Proceedings of the Australian Computer Science Conference (ACSC), Perth. Chris McDonald (ed). Springer-Verlag, 1998. pp 307-318.
167. Prototype-Based Programming for Abstract Program Visualisation. James Noble, In Proceedings of the Australian Software Visualisation Workshop, Adelaide. Chris Marlin (ed) Flinders University. ISBN 0-7258-0630-3. 1997.
168. Aspects of Synchronisation. David Holmes, James Noble, John Potter. In Proceedings of Technology of Object-Oriented Languages and Systems (TOOLS Pacific 25) Pacific, Melbourne, 1997. pp 7-18. IEEE Computer Society Press.
169. Patterns for Finding Objects within Designs. James Noble. In Proceedings of Technology of Object-Oriented Languages and Systems (TOOLS Pacific 25) Pacific, Melbourne, 1997. pp 103-110. IEEE Computer Society Press.
170. Change Detection for Aggregate Objects with Aliasing. James Noble and John Potter. In Proceedings of the Australian Software Engineering Conference (ASWEC), Sydney, 1997. IEEE Computer Society Press.

171. Visualising Alignment with Adaptive Gridlines. James Noble. (short paper).In Proceedings of the 13th IEEE Symposium on Visual Languages, 1997.
172. Arguments and Results. James Noble. In Proceedings of Conference on Pattern Languages of Program Design. Monticello, Illinois. 1997.
173. GOF Patterns for GUI Design. James Noble. In Proceedings of the European Conference on Pattern Languages of Program Design. Irsee, Germany. 1997.
174. Basic Relationship Patterns. James Noble. In Proceedings of the European Conference on Pattern Languages of Program Design. Irsee, Germany.1997.
175. Interactive Design Metric Visualisation: Visual Metric Support for User Interface Design. James Noble and Larry L. Constantine. In Proceedings of OzCHI'96, Hamilton, NZ. 1996.
176. Scribble: A Diagram Editor with a Minimal Interface. James Noble. In Proceedings of OzCHI'96, Hamilton, NZ. 1996.
177. Some Patterns for Relationships. James Noble. In Proceedings of Technology of Object-Oriented Languages and Systems (TOOLS Pacific 21), Melbourne, 1996. Prentice-Hall.
178. Found Objects: A Small Pattern Language for Finding Objects in Designs. James Noble. In Proceedings of the European Conference on Pattern Languages of Program Design. Irsee, Germany. 1996.
179. Explicit Relationships in Object Oriented Development. James Noble and John Grundy. In Proceedings of Technology of Object-Oriented Languages and Systems (TOOLS Pacific 18), Melbourne, 1995.
180. Tarraingím— A Program Animation Environment. R. James Noble and Lindsay J. Groves. In Proceedings of the 12th New Zealand Computer Conference, Dunedin, NZ. 1991.
181. An Introduction to the Tarraingím Program Animation Environment. R. James Noble and Lindsay J. Groves. In Proceedings of Technology of Object-Oriented Languages and Systems (TOOLS Pacific 6), Sydney, 1991.

Refereed Workshop Papers And Reports

182. Balancing Acts: Walking the Agile Tightrope. Rashina Hoda, James Noble, Stuart Marshall. In the Proceedings of the Workshop on Cooperative and Human Aspects of Software Engineering (CHASE) at ICSE2010.
183. Does Size Matter? A Preliminary Investigation of the Consequences of Powerlaws in Software. Joshua Lindsay, James Noble, Ewan Tempero. In the Proceedings of the Workshop on Emerging Trends in Software Metrics at ICSE 2010.
184. Balancing Acts: Walking the Agile Tightrope. Rashina Hoda, James Noble, Stuart Marshall. To appear in the Proceedings of the Workshop on Cooperative and Human Aspects of Software Engineering at ICSE 2010.
185. Understanding the Importance of Trust in Distributed Agile Projects: A Practical Perspective. Siva Dorairaj, James Noble and Petra Malik. To Appear in the Conference on Agile Processes and eXtreme Programming in Software Engineering, XP 2010.
186. OGJ gone Wild. Nicholas Cameron and James Noble. To be Presented at the International Workshop on Aliasing, Confinement, and Ownership in Object-Oriented Programming (IWACO) at ECOOP 2009.

187. From Java to UpgradeJ: An empirical study. Ewan Tempero, Gavin Bierman, James Noble, Matthew Parkinson. In the Proceedings of the First ACM Workshop on Hot Topics in Software Upgrades (HotSWUp), 2008.
188. Visualizing the word structure of Java class names. Craig Anslow, James Noble, Stuart Marshall, Ewan D. Tempero. Poster presentation in the OOPSLA Companion, 2008. pp777-778.
189. Relationships and Associations in Object-Oriented Languages. Stephen Nelson, Stephanie Balzer, Gavin M. Bierman, Erik Meijer, James Noble, David Pearce, Jiri Soukup, Frank Tip. Workshop abstract in the OOPSLA Companion, 2008. 855-856.
190. A for Agile, Issues with Awareness and Adoption. Rashina Hoda, James Noble, and Stuart Marshall. In the Proceedings of the Research-In-Progress Track at Agile 2008.
191. First Class Relationships for OO Languages. Stephen Nelson, David J. Pearce, James Noble. ECOOP 2008 Workshop on Multiparadigm Programming with Object-Oriented Languages (MPOOL) 2008.
192. Object State Querying for Optimisation. David J. Pearce, James Noble. ECOOP 2008 Workshop on Multiparadigm Programming with Object-Oriented Languages (MPOOL) 2009.
193. Web Software Visualization Using Extensible 3D (X3D) Graphics. Craig Anslow, James Noble, Stuart Marshall, and Robert Biddle. Poster, SoftVis, 2008.
194. Towards End-User Web Software Visualization. Craig Anslow, James Noble, Stuart Marshall, and Ewan Tempero. Graduate Consortium, VLHCC, 2008.
195. Towards Unifying Immutability and Ownership. Paley Li, Alex Potanin, James Noble, and Lindsay Groves. International Workshop on Aliasing, Confinement, and Ownership (IWACO), ECOOP 2008.
196. Introducing Alloy in a Software Modelling Course. James Noble, David J. Pearce, Lindsay Groves. ETAPS 2008 Workshop on Formal Methods in Computer Science Education (FORMED) 2008. pp81-91.
197. Aliasing, Confinement, and Ownership in Object-Oriented Programming Dave Clarke, Sophia Drossopoulou, James Noble, Tobias Wrigstad. ECOOP Workshops 2007: 40-49
198. A High Integrity Profile for Memory Safe Programming in Real-time Java. Alex Potanin, James Noble, Tian Zhao, Jan Vitek. In Proceedings of the 3rd Workshop on Java Technologies for Real-time and Embedded Systems, San Diego, CA, USA, October 2005.
199. Querying in Java. Darren Willis, David J. Pearce, James Noble. Presented at the OOPSLA Workshop on Multiparadigm Programming (MPOOL2005), San Diego, October 2005.
200. An Analysis of Persuasive Technology Tool Strategies. Khaled, R., Noble, J., and Biddle, R. In Proceedings of the International Workshop on the Internationalisation of Products and Systems (IWIPS) 2005, Amsterdam, Netherlands.
201. Featherweight Generic Ownership. Alex Potanin, James Noble, Dave Clarke, Robert Biddle. In proceedings of the workshop on Formal Techniques for Java-like Programs (FTfJP) in European Conference for Object-Oriented Programming, Glasgow, Scotland, July 2005.
202. Defaulting Generic Java to Ownership. Alex Potanin, James Noble, Dave Clarke, Robert Biddle. In proceedings of the workshop on Formal Techniques for Java-like Programs (FTfJP) in European Conference for Object-Oriented Programming, Oslo, Norway, June 2004.

203. Featherweight Generic Confinement. Alex Potanin, James Noble, Dave Clarke, and Robert Biddle. In proceedings of the International Workshop on Foundations of Object-Oriented Languages (FOOL), affiliated with Symposium on Principles of Programming Languages, Venice, Italy, January 2004.
204. Towards a Model of Encapsulation. James Noble, Robert Biddle, Ewan Tempero. Alex Potanin, David Clarke. Presented at the ECOOP 2003 International Workshop on Aliasing and Confinement (IWACO). 2003.
205. Larry Constantine, Robert Biddle, and James Noble. Usage-centered design and software engineering: Models for integration. In IFIP Working Group 2.7/13.4, editor, ICSE 2003 Workshop on Bridging the Gap Between Software Engineering and Human-Computer Interaction, Portland, Oregon, 2003
206. Angela Martin, Brenda Chawner, Robert Biddle, and James Noble. Requirements with XP and the open source customer. In Brian Fitzgerald and David Parnas, editors, Proceedings of the Workshop on Free and Open Source Software, Genoa, Italy, 2003
207. Checking Ownership and Confinement Properties. Alex Potanin and James Noble. Presented at the ECOOP 2002 Workshop on Formal Techniques for Java-like Programs.
208. ECOOP'99 Intercontinental Workshop on Aliasing in Object-Oriented Systems. James Noble and Jan Vitek and Paulo Sergio Almeida. In Object-Oriented Technology: ECOOP'99 Workshop Reader. Ana Moreira and Serge Demeyer (eds). LNCS 1743. Springer-Verlag 1999.
209. The Objects of Aliasing. James Noble. ECOOP'99 Workshop on Aliasing in Object-Oriented Systems. In Object-Oriented Technology: ECOOP'99 Workshop Reader. Ana Moreira and Serge Demeyer (eds). LNCS 1743. Springer-Verlag 1999.
210. Three Features for Component Frameworks. James Noble. ECOOP'99 Workshop on Component Oriented Programming. In Object-Oriented Technology: ECOOP'99 Workshop Reader. Ana Moreira and Serge Demeyer (eds). LNCS 1743. Springer-Verlag 1999.
211. Overcoming Representation Exposure. David Clarke, James Noble, John Potter. ECOOP'99 Workshop on Aliasing in Object-Oriented Systems. In Object-Oriented Technology: ECOOP'99 Workshop Reader. Ana Moreira and Serge Demeyer (eds). LNCS 1743. Springer-Verlag 1999.
212. Toward Reusable Synchronisation for Object-Oriented Languages. David Holmes, James Noble, John Potter. In Object-Oriented Technology: ECOOP'98 Workshop Reader. Serge Demeyer and Jan Bosch (Eds.) LNCS 1543, Springer-Verlag, 1998.

Patent Application

213. Industrial automation display arrangement and method. Helmut Windl, Helmut Fritz, Larry Constantine, James Noble. United States Patent Application 20020070972, 2002.

Workshop Papers and Reports

214. John Rankin, Craig Anslow, James Noble, Brenda Chawner, Donald Gordon. Wikipublisher: a print-on-demand Wiki. International Symposium on Wikis (WIKISYM) 2009
215. Introduction to Computer Science & Software Engineering at Victoria University of Wellington. Presentation to EU COST Action IC0701 Meeting, Lisbon, Portugal, June 2009

216. Abstraction vs Verification: Classes vs Patterns: Presentation to EU COST Action IC0701 Meeting, Lisbon, Portugal, June 2009
217. The Art of Funding your Research. Presentation to the BuildIT Emerging Researchers' Talent Camp, Auckland, 2009.
218. Experience on the Human Side of Agile. Angela Martin, Robert Biddle, James Noble. XP2008.
219. Teaching Object-Oriented Modelling with Alloy. James Noble, David J. Pearce, Lindsay Groves. OOPSLA Educators' Symposium, Nashville, 2008.
220. Introducing Software Modelling with Alloy at VUW. James Noble, David J. Pearce, Lindsay Groves. SIENZ — Software Engineering New Zealand 2007.
221. X3D Web Software Visualisation in Action. Craig Anslow, James Noble, Stuart Marshall, Robert Biddle. OOPSLA Companion 2007. p734
222. Software Visualization Tools for Component Reuse. Craig Anslow, Stuart Marshall, James Noble, and Robert Biddle. Second Workshop on Method Engineering for Object-Oriented and Component-Based Development at OOPSLA 2004. Magdy Serour, eds. Centre for Object Technology Applications and Research (COTAR), Sydney, 2004.
223. Angela Martin, Brenda Chawner, Robert Biddle, and James Noble. Requirements with XP and the open source customer. In Brian Fitzgerald and David Parnas, editors, Proceedings of the Workshop on Free and Open Source Software, Genoa, Italy, 2003
224. Checking Ownership and Confinement Properties. Alex Potanin and James Noble. Presented at the ECOOP 2002 Workshop on Formal Techniques for Java-like Programs.
225. Lightweight Web-Based Tools for Usage-Centered and Object-Oriented Design. James Noble and Robert Biddle. Presented at forUSE 2002, Portsmouth, New Hampshire, August 2002.
226. The Language of Programming. Robert Biddle and James Noble. Presented at the Feyerabend Workshop at ECOOP 2002.
227. Visualising Objects-Oriented Systems. James Noble. Presented at Schloss Dagstuhl Seminar 01211, Schloss Dagstuhl Report No. 307, 2001
228. Dynamic Flexible Alias Protection in a Prototype-Based Language. James Noble, David Clarke, John Potter. Thinking with Prototypes — OOPSLA'98 Workshop on Prototype-based Object-Oriented Programming.
229. GUI Iterator and GUI Decorator. James Noble. OOPSLA'97 Workshop on Non-software Examples of Design Patterns. 1997.
230. Prototype Based User Interfaces. James Noble. ECOOP'97 Workshop on Prototype-Based Programming. Jyvaskyla, Finland. 1997.
231. Aspects of Synchronisation. David Holmes, James Noble, and John Potter. ECOOP'97 Workshop on Aspect Oriented Programming, Jyvaskyla, Finland. 1997.
232. On Prototype Based User Interfaces. James Noble. COTAR'96 Workshop. Melbourne, 1996.
233. ECOOP'96 Workshop on Prototype Based Object Oriented Programming. James Noble and Antero Taivalsaari. ACM SIGPLAN Notices, December 1996.
234. Encapsulators in Self. James Noble. ECOOP'96 Workshop on Prototype Based Object Oriented Programming. Linz, Austria. 1996
235. Scribble: A Prototype Lightweight Design Tool. James Noble. COTAR'95 Workshop. Sydney, 1995.

236. Some Patterns of Tarraingím. James Noble. OOPSLA'95 Workshop on Frameworks for Software Toolsets. Austin, Texas. 1995.
237. Moses and Aaron. James Noble and Karin Kolbe. OOPSLA'95 Workshop on CASE Frameworks. Austin, Texas. 1995.

Technical Reports and Miscellaneous Publications

238. Towards a Semiotics of Object- and Aspect-Oriented Design. James Noble, Robert Biddle, Ewan Tempero. Abstract accepted for presentation at the European Conference on Computing and Philosophy (ECAP 2006), Trondheim Norway.
239. Barr, P., Noble, J., and Biddle, R. Oppositional Play in Action, poster presented at Children's Learning in a Digital World Conference, 2005
240. Khaled, R., Luxton, A, Noble, J., Ferres, L., Judy, B., and Biddle, R. Visualisation for learning OOP, using AOP and eclipse. Poster at the 19th annual ACM SIGPLAN conference on Object-Oriented Programming Systems, Languages, and Applications, 2004.
241. Oh! Gee! Java! Ownership types for (almost) free. James Noble, Robert Biddle. 2003.
242. Object Ownership to Order. David Clarke, Ryan Shelswell, John Potter, James Noble, 1998.
243. Adaptive Gridlines for Aligning Objects. James Noble, MRI Technical Report, 1997.
244. Abstract Constraints. James Noble. MRI Technical report, 1996.
245. Writers' Workshop Patlets. James Noble. MRI Technical report, 1996.
246. Emergent Architecture. James Noble. MRI Technical Report, 1996.
247. Java to Eiffel Translator — Project Bruce Initial Report. James Noble. MRI Technical Report, 1996.
248. Shaping Object Oriented Programs. Barry Jay and James Noble. UTS Technical Report, 1996.
249. Book Review: Object-Oriented Development: Building CASE Tools with C++ by David Brumbaugh. R. James Noble. SIGPLAN Notices
250. Book Review: Parallel Logic Programming by Akikazu Takeuchi. R. James Noble. SIGPLAN Notices, 30(4), April 1995.