

## CV

### Affiliation and contact address:

Associate Professor, Department of Computer Science, School of Computing,  
National University of Singapore (NUS)

Mailing address: Computing 1, 13 Computing Drive, Singapore 117590

fax: 65-6779-1610; tel: 65-6516-2863 (office) 65-96255863 (mobile)

[stan@comp.nus.edu.sg](mailto:stan@comp.nus.edu.sg); <http://www.comp.nus.edu.sg/~stan>

Home: 18 Yew Siang Rd #01-04, Flynn Park, Singapore 117755; 65-64764003

**Citizenship:** Polish and Canadian; Singapore PR

### Education:

**Ph.D.** (1979), Dept. of Mathematics (Computer Science Division), Warsaw University, *Thesis*: “A Generic Program Optimizer for a Compiler-Compiler”

**M. Sc.** (1972), Dept. of Mathematics, Warsaw University, *Thesis*: “Product K-Machines”

**Teaching areas:** Software Engineering (year 3), Software Engineering Project (year 3), Object-Oriented Methods (year 4), Software Architectures (level 4), Software Re-engineering (graduate level), Software Reuse (graduate level), Compilers (year 4), Data Structures (year 3), Formal Language Theory (year 4)

### Professional career at glance:

**1992-now** Associate Professor, Department of Computer Science, School of Computing, National University of Singapore (tenure in 97)

**2000-06** Adjunct Associate Professor, Department of Electrical & Computer Engineering, University of Waterloo, Waterloo, Ontario, Canada

**1999 Sabbatical leave: January-June:** Visiting Scientist at Fraunhofer Institute for Experimental Software Engineering, Kaiserslautern, Germany; short visits to centers of Prof. Mehdi Jazayeri, Technical University of Vienna and Prof. Carlo Ghezzi, Politecnico de Milano; **July-October:** Visiting Associate Professor at Software Engineering Group, Department of Electrical and Computer Engineering, University of Waterloo, Canada

**1990-92** Research Manager, CSA Research Pte. Ltd. in Singapore and Adjunct Senior Lecturer, National University of Singapore

CSA Research, a CASE tool development company, engaged me to plan the next generation company product with reverse engineering and re-engineering capabilities. I also participated in management of a CASE tool development project.

**1984-89** Assistant Professor, Department of Computer Science and Systems, McMaster University, Hamilton, Ontario, Canada

I did research on software engineering environments; designed a generation system for language-based programming environments; designed an incremental attribute evaluation mechanism propagating attributes across a forest of inter-related syntax trees. I did consulting on compiler technology and C++ for industries in Toronto.

**1982-84** Lecturer I at the University of Maiduguri, Nigeria

**1972-82** Research Fellow, Institute of Computers - an industrial research institute in Warsaw

I worked on a compiler-compiler project; designed a generic program optimizer for a class of n-tuple compiler intermediate languages; implemented a number of cross-compilers.

## Research

Reuse is based on the premise that much similarity exists in software systems. This is particularly true for systems in a given application domain. Software Product Line (SPL) approach is based on reuse of core assets – architecture, code components, documentation and other software artifacts – during software development and evolution. Reuse attempts to identify and automate routine, repetitive development tasks. While the progress towards practical solutions is slow, reuse remains the most promising approach in the horizon to improving software quality and productivity.

My goal is to have impact on software practice. I lead research on software reuse at Software Engineering Lab. NUS. We have been working on methods and tools to support SPLs since 2000. Initially, our work focused on variability management, to help domain engineers design generic, adaptable core assets, and to automate reuse-based product development. We developed XVCL (XML-based Variant Configuration Language, <http://xvcl.comp.nus.edu.sg>), a uniform technique to manage variability in SPL core assets such as code, architecture, documentation (e.g., written in WORD), models, or test cases. In 2000-02, I was a Principal Investigator in the Singapore-Ontario Joint Research project: *Reuse Framework for Reliable Mission-Critical Software Systems*. The project was funded by A\*STAR (Agency for Science, Technology and Research), and the Canadian Ministry of Energy, Science and Technology. Project partners included NUS, University of Waterloo, SES Systems Pte Ltd and Netron, Inc, Toronto. This project led to XVCL.

We have applied XVCL to SPLs in range of application domains (command & control systems, business applications, games for mobile phones, Web portals), using variety of programming technologies (Java, ASP, .NET, JEE). We have developed XVCL Workbench (prototype) comprising tools for creating, analyzing, debugging, reusing, and evolving SPL core assets.

In complementary line of work, we developed techniques for re-engineering legacy code into SPL. Our approach is based on detecting similar program structures (called structural clones) in existing product variants. Structural clones recurring in product variants help us identify program areas with much reuse opportunities - attractive candidates for core assets [34][37][42][43]. We developed a technique and tool called Clone Miner [2] capable of detecting certain classes of structural clones, and Clone Analyzer [35] that allows a human expert to participate in post-detection analysis of software clones. Our experimental work demonstrated usefulness of structural clones in software re-engineering for reuse, program understanding and design recovery.

My team aims at a Software Product Line (SPL) method and toolset (SPL Workbench) providing comprehensive support for the whole SPL lifecycle, namely:

- 1) Re-engineering of legacy code (existing product variants) into SPL core assets for reuse.
  - a) Analysis of similarities and differences in existing product variants
  - b) Design of common Product Line Architecture shared by products
  - c) Applying variability management techniques to create generic, adaptable and reusable core assets
- 2) Reuse-based development of custom products from SPL core assets.
- 3) Evolution of SPL core assets and custom products deployed to customers, while keeping evolving assets in sync with evolving products.

In long run, we aim at practical, scalable SPL method and toolset that will successfully compete with other SPL tools available on the market. While in our lab we can only develop SPL Workbench prototypes, we want these prototypes to be a base for development of a production-quality SPL Workbench (in cooperation with and sponsorship from industry).

We have been working with industry partners to make our SPL methods directly relevant to industrial practice. I believe our concept of how to support SPL is more comprehensive, effective and automated than commercial SPL tools and SPL solutions proposed by others.

The cornerstones of my strategy are the following:

- 1) Continue research in SPL to get insights into core problems and continuously improve our approach.
- 2) Collaborate with research centers and industries to complement our competencies, and to apply our SPL methods in wide range of situations.
- 3) Apply our SPL methods in lab studies and industrial projects to validate our methods.

- 4) Source government funding for SPL basic research. Source funding for SPL Workbench from industry or liaise with an industrial partner (or venture capitalists) who will bring SPL Workbench to the production-quality status and promote it to the market.

#### ***On-going projects***

- 1) Re-engineering into SPL: similarity analysis of legacy code (clone, semantic differentiation of clones), feature analysis, and XVCL-Re-engineering Facility.
- 2) Documenting SPL: Documenting SPL with UML models with variants, Managing SPL documentation in WORD.
- 3) MDD and generators for SPL: flexible generator approach with MS DSL Toolkit and XVCL.
- 4) Model transformations to automate transition from analysis to design models in MDA.
- 5) Testing in SPL: Building libraries of generic test cases, adaptable to specific products, SPL members.
- 6) SPL Workbench.

Themes 1) - 5) offer research challenges, are suitable for lab studies and PhD thesis. Documenting SPLs, and applying MDD and generators for SPL is of great practical importance, and offers exciting opportunities for research.

Theme 6): By now, techniques underlying Workbench have matured, and required tool capabilities are well understood. However, implementation of the production-quality SPL Workbench poses engineering challenges, in terms of usability and reliability that are difficult to address in university projects, with research grant funding. While Basic Workbench is close to completion and can be useful tool for developers on its own, other tools require much effort to bring them to the production-quality status. Our plan is to either release XVCL Workbench to open source or complete the work in collaboration with and sponsorship from industry.

***Industry collaborations:*** We have been working with industry partners, namely ST Electronics (Info-Software Systems) Pte Ltd, Paul Basset (Cutter Consortium), and Fudan Wingsoft Ltd. Collaboration with ST Electronics led to the first application of XVCL in industry setting [44]. Paul Bassett is an inventor of frame concepts, and former Research Director of Netron Inc., a company that developed Frame Technology™, a predecessor of XVCL. We have 10 year history of fruitful collaboration with ST Electronics and Paul Bassett, documented in published papers. Fudan Wingsoft Ltd is a university-affiliated software company China, a developer of financial software for universities. Our collaboration centers on Fudan Wingsoft's financial software SPL [32]. We study conventional variability management techniques Wingsoft uses to manage variability in core assets, and apply XVCL to achieve similar goals, evaluating the benefits and trade-offs.

Since 2000, the total grant value to support our SPL research has been S\$687,000.

#### **Supervision of students**

Currently supervised: 4 PhD students, 3 Undergraduate Research students

Past supervision: 3 PhD students, 15 Master students, 6 Research Assistant, 40 Undergraduate Research students, 10 overseas interns

#### **Consultancy**

Software design, expert witness in disputes about potential plagiarism of design and code.

Expert consultant on software design: Evaluation and second opinion on software design of a system under development, or existing system; propose and evaluate alternative designs; finding potential faults.

Expert witness in software plagiarism: Cases of dispute about potential plagiarism in court cases involving design plagiarism, code plagiarism, IP. Providing opinion regarding the reasons for project failures.

## Publications

### Books:

Jarzabek, S. [\*Effective Software Maintenance and Evolution: Reused-based Approach\*](#), Auerbach, CRC Press Taylor and Francis, May 2007 (480 pages)

Rajapakse, D.C. and Jarzabek, S. [\*Using Similarity Patterns in Developing Web Applications: An Approach to Enhance Reuse and Maintainability\*](#), VDM Verlag, 2010 (156 pages)

Basit, H. and Jarzabek, S. [\*Towards Structural Clones: Analysis and semi-automated detection of design-level similarities in software\*](#), VDM Verlag, 2010 (172 pages)

### Articles in refereed journals:

1. Rajapakse, D. and Jarzabek, S. ["Towards generic representation of web applications: solutions and trade-offs,"](#) *Software, Practice & Experience*, Volume 39 Issue 5, April 2009, pp. 501 – 530, Published Online: 27 Nov 2008
2. Basit, H. A., Jarzabek, S. ["Data Mining Approach for Detecting Higher-level Clones in Software,"](#) *IEEE Trans. on Soft. Eng.*, July/August 2009 (vol. 35 no. 4) pp. 497-514; Published online January 2009
3. Jarzabek, S. and Li, S. "Unifying clones with a generative programming technique: a case study," *Journal of Software Maintenance and Evolution: Research and Practice*, John Wiley & Sons, Volume 18, Issue 4, July/August 2006, pp. 267-292, extended version of ESEC-FSE'03 paper that received ACM Distinguished Paper Award
4. Sun, J., Dong, J.S, and Jarzabek, S. "CAD System Family Architecture and Verification: An Integrated Formal Approach," *IEE Proceedings Software*, IEE and British Computer Society, Vol. 153, No. 3, July 2006, p. 87-136
5. Jarzabek, S, Yang, B. and Sam, S. "Addressing Quality Attributes in Domain Analysis for Product Lines," *IEE Proceedings Software*, IEE and British Computer Society, Vol. 153, No. 2, April 2006, pp. 61-73
6. Jarzabek, S., Zhang, H., Ru, S., Lam, V.T., and Sun, Z. "Analysis of meta-programs: a case study," *Journal of Software Engineering and Knowledge Engineering*, Vol. 16, No. 1, Feb. 2006, pp. 77-101, extended version of **best papers** from *Proc. 16th Int. Conference on Software Engineering and Knowledge Engineering (SEKE'04)*, Banff, Canada, June 2004
7. Zhang, H. and Jarzabek, S. "A Bayesian Network Approach to rational architectural design," *Int. Journal of Software Engineering and Knowledge Engineering*, Vol. 15, No. 4, August 2005, pp. 695-719
8. Zhang, H. and Jarzabek, S. ["A Mechanism for Handling Variants in Software Product Lines,"](#) special issue on Software Variability Management of Elsevier's journal *Science of Computer Programming*, Volume 53, Issue 3, Dec. 2004, pp. 381-407
9. Stan Jarzabek, Wai Chun Ong and Hongyu Zhang ["Handling Variant Requirements in Domain Modeling,"](#) *Journal of Software and Systems*, Vol. 68, Issue 3, 15 Dec. 2003, pp.171-182; extended version of **best papers** on Software Engineering from conf. SEKE'01
10. Jarzabek, S. and R. Seviora ["Engineering components for ease of customization and evolution,"](#) *IEE Proceedings - Software*, Vol. 147, No. 6, December 2000, pp. 237-248, a special issue on Component-based Software Engineering
11. Jarzabek, S. and G. Wang ["Model-based Design of Reverse Engineering Tools"](#), *Journal of Software Maintenance: Research and Practice*, No. 10, 1998, John Wiley & Sons, pp. 353-380
12. Jarzabek, S. ["Design of Flexible Static Program Analyzers with PQL,"](#) *IEEE Transactions on Software Engineering*, March 1998, pp. 197-215
13. Chee, C.L., Jarzabek, S. and Paul, R. ["F-metric: a WWW-based framework for intelligent formulation and analysis of metric queries,"](#) *Journal of Systems and Software*, No. 43, 1998, Elsevier Science Inc., pp. 119-132
14. Jarzabek, S. and Huang, R. ["The case for User-Centered CASE Tools,"](#) *Communications of ACM*, August 1998, pp. 93-99
15. Jarzabek, S. and T.W. Ling "Model-based Support for Business Re-engineering," *Journal of Information and Software Technology*, vol. 38, No. 5, May 1996, pp. 355-374
16. Jarzabek, S "Lifecycle approach to strategic re-engineering of software," *Journal of Software Maintenance: Research and Practice*, vol. 6, no. 6, December 1994, John Wiley & Sons, 287-317
17. Jarzabek, S., CL Tan and Tham, K. An Object-oriented Model for Recovered Designs in Software Reengineering. *Information Technology Journal*, vol. 6, no. 2, December 1994, 80-94
18. Jarzabek, S. "Domain Model-Driven Software Reengineering and Maintenance," *Journal of Systems and Software*, January 1993, 20:37-51
19. Jarzabek, S. "Research Trends in Software Development Environments," *Information Technology Journal*, April 1991, vol. 4, no. 1, 9-14
20. Jarzabek, S. "Specifying and Generating Multi-Language Software Development Environments," *Software Engineering Journal*, IEE and British Computer Society, March 1990, 125-137
21. Jarzabek, S. "The Role of Specifications and Abstractions in the Design of a Software Environment Generation System," *International Journal on Policy and Information*, vol. 13, no. 2, December 1989, 145-164

22. Jarzabek, S. and Krawczyk, T. "LL-Regular Grammars," *Information Processing Letters*, No. 2, vol. 4, 1975

**Letters to editors:**

23. Jarzabek, S. "Will MDD Fulfill It's Promises?" *IEEE Software*, Jan/Feb 2004, pp.5-6

**Articles in refereed international conference proceedings:**

24. Jarzabek, S and Trung, H. D. "Flexible Generators for Software Reuse and Evolution," *Int. Conf. on Software Eng., ICSE'2011*, New Ideas and Emerging Results Track, Honolulu, USA, May 2011, pp. 920-923
25. Jarzabek, S, Pettersson, U. and Zhang, H. "University-Industry Collaboration Journey towards Product Lines," *Int. Conf. on Software Reuse, ICSR'2011*, S. Korea, June 2011, pp. 223-237
26. Zhu, J., Peng, X., Jarzabek, S., Xing, Z., Xue, Y., and Zhao, W. "Improving Product Line Architecture Design and Customization by Raising the Level of Variability Modeling," *Int. Conf. on Software Reuse, ICSR'2011*, S. Korea, June 2011, pp. 151-166
27. Xue, Y, Jarzabek, S., Ye, P., Peng, X., and Zhao, W. "Scalability of Variability Management: An Example of Industrial Practice and Some Improvements," 23<sup>rd</sup> *Int. Conf. Soft. Eng and Knowledge Eng, SEKE*, July 2011, USA, pp. 705-710
28. Basit, H., Ali, U. and Jarzabek, S. "Viewing Simple Clones from a Structural Clones' Perspective," *Int. Workshop on Software Clones, IWSC'2011*, ICSE Workshop, Honolulu, USA, May 2011, pp. 1-6
29. Xue, Y., Xing, Z. and Jarzabek, S. "[Understanding Feature Evolution in a Family of Product Variants](#)," 17<sup>th</sup> *Working Conf. on Reverse Engineering*, Boston, MA, October 2010, pp. 109-118
30. Zhang, H. and Jarzabek, S. "Hybrid Approach to Feature-Oriented Programming," *Software Product Lines Conference, SPLC'10: Going Beyond*, Lecture Notes in Computer Science, Springer, Aug. 2010, Volume 6287/2010, pp. 440-445
31. Jarzabek, S. Xue, Y., Zhang, H. and Lee, Y. "Avoiding Some Common Preprocessing Pitfalls with Feature Queries," *Proc. 16<sup>th</sup> Asia-Pacific Software Engineering Conference, APSEC'09*, Penang, Dec. 2009, pp.283-290
32. Ye, P., Peng, X, Xue, Y. and Jarzabek, S. "A Case Study of Variation Mechanism in an Industrial Product Line," 11<sup>th</sup> *Int. Conf. on Software Reuse, ICSR09*, Falls Church, VA, USA, Sept. 27-30, 2009, Springer, pp. 126-136
33. Stan Jarzabek, Hongyu Zhang, Youpeng Lee, Yinxing Xue, Naveed Shaikh "Increasing Usability of Preprocessing for Feature Management in Product Lines with Query-based Visualization," *Int. Conf. Software Engineering, ICSE'09*, Vancouver, Canada, May 2009, pp. 111-114 (a track on New Ideas and Emerging Results)
34. Basit, H. and Jarzabek, S. "A Case for Structural Clones," 3<sup>rd</sup> *Int. Workshop on Software Clones, IWSC'2009*, Kaiserslautern, Germany, Tuesday, March 24, 2009, pp. 18-22
35. Yali Zhang, Hamid Abdul Basit, Stan Jarzabek, Dang Anh, and Melvin Low "[Query-based Filtering and Graphical View Generation for Clone Analysis](#)," *Proc. 24<sup>th</sup> IEEE Int. Conf. on Software Maintenance, ICSM'08*, Beijing, September 2008, pp. 376-385
36. Lok Fang Fang Stella, Stan Jarzabek and Bimlesh Wadhwa "A Comparative Study of Maintainability of Web Applications on J2EE, .NET and Ruby on Rails," 10<sup>th</sup> *IEEE Int. Symp. on Web Site Evolution*, Beijing, October 2008
37. Basit, H., Puglisi, S., Smyth, W., Turpin, A. and Jarzabek, S. "[Efficient Token Based Clone Detection with Flexible Tokenization](#)," *ESEC-FSE'07, European Software Engineering Conference and ACM SIGSOFT Symposium on the Foundations of Software Engineering*, ACM Press, September 2007, Dubrovnik, pp. 513-516
38. Grudzien, A., Traczyk, T. and Jarzabek, S. "[Application of Generative Programming to Evolution of Object-Relational Mapping Layer](#)," *Proc. 2<sup>nd</sup> AIS SIGSAND European Symposium on System Analysis and Design*, Gdansk, June 5, 2007, pp. 64-71, ISBN 978-83-7326-447-2
39. Rajapakse, D.C. and Jarzabek, S. "[Using Server Pages to Unify Clones in Web Applications: A Trade-off Analysis](#)," *Int. Conf. Software Engineering, ICSE'07*, Minneapolis, USA, May 2007, pp. 116-125
40. Peng, D., Jarzabek, S., Rajapakse, D. and Zhang, H. "Reuse of Database Access Layer Components in JEE Product Lines: Limitations and a Possible Solution (Case Study)," *Proc. 9<sup>th</sup> Int. Conf. on Soft. Eng & Knowledge Eng, SEKE'07*, Boston, July 2007, pp. 308-313
41. Jarzabek, S. "[Genericity - a Missing in Action](#)" *Key to Software Simplification and Reuse*," 13<sup>th</sup> *Asia-Pacific Software Engineering Conference, APSEC'06*, IEEE Comp. Soc., 6-8 December 2006, Bangalore, India, pp. 293-300
42. Basit, H.A., Rajapakse, D.C., and Jarzabek, S. "[Beyond Templates: a Study of Clones in the STL and Some General Implications](#)," *Int. Conf. Software Engineering, ICSE'05*, St. Louis, USA, May 2005, pp. 451-459
43. Basit, A.H. and Jarzabek, S. "[Detecting Higher-level Similarity Patterns in Programs](#)," *ESEC-FSE'05, European Software Engineering Conference and ACM SIGSOFT Symposium on the Foundations of Software Engineering*, ACM Press, September 2005, Lisbon, pp. 156-165
44. Pettersson, U., and Jarzabek, S. "[Industrial Experience with Building a Web Portal Product Line using a Lightweight, Reactive Approach](#)," *ESEC-FSE'05, European Software Engineering Conference and ACM SIGSOFT Symposium on the Foundations of Software Engineering*, ACM Press, September 2005, Lisbon, pp. 326-335

45. Zhang, W. and Jarzabek, S. "[Reuse without Compromising Performance: Experience from RPG Software Product Line for Mobile Devices](#)," *9<sup>th</sup> Int. Software Product Line Conference, SPLC'05*, September 2005, Rennes, France, pp. 57-69
46. Yang, J. and Jarzabek, S. "[Applying a Generative Technique for Enhanced Reuse on J2EE Platform](#)," *4<sup>th</sup> Int. Conf. on Generative Programming and Component Engineering, GPCE'05*, Sep 29 - Oct 1, 2005, Tallinn, Estonia, pp. 237-255
47. Rajapakse, D.C and Jarzabek, S. "[A Need-Oriented Assessment of Technological Trends in Web Engineering](#)," *Int. Conf. on Web Engineering, ICWE'05*, July 2005, Sydney, pp. 30-35
48. Rajapakse, D.C and Jarzabek, S. "[An Investigation of Cloning in Web Portals](#)," *Int. Conf. on Web Engineering, ICWE'05*, July 2005, Sydney, pp. 252-262 (also poster at *WWW'05*)
49. Rajapakse, D.C, Basit, A.H. and Jarzabek, S. "[An Empirical Study on Limits of Clone Unification Using Generics](#)" for *17<sup>th</sup> Int. Conference on Software Engineering and Knowledge Engineering, SEKE'05*, July 2005, Taipei, Taiwan, pp. 109-114
50. Jarzabek, S. and Eng, P.K. "[Teaching an Advanced Design, Team-oriented Software Project Course](#)," *18<sup>th</sup> Int. Conference on Software Engineering Education and Training (CSEE&T)*, IEEE CS, April 2005, Ottawa, pp. 223-230
51. Jarzabek, S., Ru, S., Zhang, H. and Sun, Z. "Analysis of meta-programs: a case study," *Proc. 16th Int. Conference on Software Engineering and Knowledge Engineering (SEKE'04)*, Banff, Canada, June 2004, pp. 68-73; **selected as one of the best papers** for a special issue of *Journal of Software and Systems*
52. Loughran, N., Rashid, A., Zhang, W. and Stan Jarzabek "Supporting Product Line Evolution with Framed Aspects," *3<sup>rd</sup> AOSD Workshop on Aspects, Components, and Patterns for Infrastructure Software, ACPAIS'04*, March 22-26, 2004, Lancaster UK
53. Jarzabek, S. and Li, S. "[Eliminating Redundancies with a "Composition with Adaptation" Meta-programming Technique](#)," *Proc. ESEC-FSE'03, European Software Engineering Conference and ACM SIGSOFT Symposium on the Foundations of Software Engineering*, ACM Press, September 2003, Helsinki, pp. 237-246; **the paper received ACM SIGSOFT distinguished paper award**
54. Zhang, H. and Jarzabek, S. "An XVCL approach to handling variants: A KWIC product line example," *10<sup>th</sup> Asia-Pacific Software Engineering Conference, APSEC'03*, IEEE Comp. Soc., 10-12 December 2003, Chiangmai, Thailand
55. Zhang, W., Jarzabek, S., Loughran, N and Rashid, A. "Reengineering a PC-based System into the Mobile Device Product Line," *Proc. 4th Int. Workshop on Principles of Software Evolution IWPSE'03*, IEEE Comp. Soc., September 2003, Helsinki, Finland, pp. 149-160
56. Zhang, H. and Jarzabek, S "[An XVCL-based Approach to Software Product Line Development](#)," *Conf. on Software Engineering and Knowledge Engineering, SEKE'03*, San Francisco, July 2003, pp. 267-275
57. Zhang, H., Jarzabek, S. and Yang, B. "Quality Prediction and Assessment for Product Lines," *Conf. on Advanced Information Systems Engineering CAiSE'03*, Austria, June 2003, Springer-Verlag LNCS 2681, pp. 681-695
58. Soe, M.S., Zhang, H. and Jarzabek, S. "XVCL: A Tutorial," *Proc. of 14<sup>th</sup> Int. Conf. on Software Engineering and Knowledge Engineering, SEKE'02*, ACM Press, July 2002, Italy, pp. 341-349
59. Jarzabek, S. and Zhang, H. "Software Decomposition and Instrumentation for Enhanced Flexibility and Reusability," *Proc. of IASTED Int. Symp. on Software Engineering, Databases and Applications*, ACTA Press, Feb. 2002, Innsbruck, pp. 91-96
60. Zhang, H., Jarzabek, S. and Myat Swe, S. "XVCL Approach to Separating Concerns in Product Line Assets," *Proc. of 3<sup>rd</sup> International Conference on Generative and Component-Based Software Engineering*, LNCS, Springer Verlag, September 2001, Erfurt, Germany, pp. 36-47
61. Jarzabek, S. "[Flexible components with frame technology: a case study](#)," *27<sup>th</sup> EUROMICRO Conference on Component-based Software Engineering*, IEEE Comp. Soc., September 2001, Warsaw, Poland, pp. 146-153
62. Jarzabek, S. and Zhang, H. "[XML-based Method and Tool for Handling Variant Requirements in Domain Models](#)", *Proc. of 5<sup>th</sup> IEEE International Symposium on Requirements Engineering, RE'01*, IEEE Comp. Soc., August 2001, Toronto, Canada, pp. 166-173
63. Stan Jarzabek, Wai Chun Ong and Hongyu Zhang "Handling Variant Requirements in Domain Modeling," *Proc. of 13<sup>th</sup> International Conference on Software Engineering and Knowledge Engineering, SEKE'01*, Knowledge System Institute, June 2001, Buenos Aires, Argentina, pp. pp. 61-68; **selected as one of the best papers** for a special issue of *Journal of Software and Systems*
64. Stan Jarzabek and Hongyu Zhang, "Enhancing Component Reuse with Control Flow Abstraction Analysis," *Proc. of 13<sup>th</sup> International Conference on Software Engineering and Knowledge Engineering, SEKE'01*, Knowledge System Institute, June 2001, Buenos Aires, Argentina, pp. 171-178
65. Wong, T.W., Jarzabek, S., Myat Swe, S., Shen, R. and Zhang, H.Y. "[XML Implementation of Frame Processor](#)," *Proc. ACM Symposium on Software Reusability, SSR'01*, ACM Press, Toronto, Canada, May 2001, pp. 164-172
66. Zhang, H., Jarzabek, S. and Myat Swe, S., "x-Frame Approach for Handling Variants within Concerns," *Workshop on Advanced Separation of Concerns at 23<sup>rd</sup> International Conference on Software Engineering, ICSE'01*, Toronto, Canada, 2001, pp. 146-151

67. Jarzabek, S. and Knauber, P. "[Synergy between Component-based and Generative Approaches](#)," *Proc. ESEC/FSE'99 Joint 7<sup>th</sup> European Software Engineering Conference and 7<sup>th</sup> ACM SIGSOFT Symposium on the Foundations of Software Engineering*, ACM Press, Toulouse, France, September 1999, Lecture Notes in Computer Science No. 1687, Oscar Nierstrasz and Michel Lemoine (Eds.) Springer Verlag, pp. 429-445
68. Cheong, Y.C. and Jarzabek, S. "[Frame-based Method for Customizing Generic Software Architectures](#)," *Symposium on Software Reusability, SSR'99*, ACM Press, Los Angeles, May 1999, pp. 103-112
69. Teh H.Y., Jarzabek, S. and Tiako, P. "WWW-based Communication Tool for Distributed Team-based Software Development," *Proc. Conf. Systemics, Cybernetics and Informatics and the International Conference on Information Systems Analysis and Synthesis, SCI'99/ISAS'99*, Florida, August 1999
70. Jarzabek, S. "Component Criteria for Software System Families," *Proc. 11<sup>th</sup> CAiSE'99*, Heidelberg, June 1999, Lecture Notes in Computer Science No. 1626, *Advanced Information Systems Engineering*, Springer Verlag
71. Jarzabek, S. and Hitz, M. "Business-oriented and Component-based Software Development and Evolution," *International Workshop on Large-Scale Software Composition*, August 28, 1998, Vienna, Austria
72. Lau, K. W. and Jarzabek, S. "A Generic Discretionary Access Control System for Reuse Frameworks", *COMPSAC'98*, IEEE Comp. Soc., August 19-21, 1998, Vienna, Austria, pp. 356-361
73. Jarzabek, S. and Woon, I. "Interplay between an Enterprise Information Architecture and Domain Analysis," *Proc. Third World Conference on Integrated Design and Process Technology*, July 6, 1998, Berlin, pp. 154-161
74. Cheong, Y.C. and Jarzabek, S. "Modeling Variant User Requirements in Domain Engineering for Reuse," *Proc. 8<sup>th</sup> European-Japanese Conference on Information Modeling and Knowledge Bases*, Vammala, Finland, May 1998, pp. 231-250; also published in *Information Modelling and Knowledge Bases*, Eds. Hannu Jaakkola, Hannu Kangassalo and Eiji Kawaguchi, IOS Press, Netherlands, ISSN: 0922-6389, pp. 220-234
75. Cheong Y. C., Ananda, A. L. and Jarzabek, S. "Handling Variant Requirements in Software Architectures for Product Families," *Proc. 2<sup>nd</sup> International Workshop on Software Architectures for Product Families*, 26 February, 1998, Las Palmas, Gran Canaria, Spain, Lecture Notes in Computer Science No. 1429, Frank van der Linden (ed.), *Development and evolution of software architectures for product families*, Springer Verlag
76. Jarzabek, S. "Modeling Multiple Domains for Software Reuse," *Proc. Symposium on Software Reusability, SSR'97*, Boston, May 1997, ACM Press, pp. 65-79
77. Jarzabek, S and Woon, I. "Towards precise Description of Reverse Engineering Heuristics," *Proc. EUROMICRO Working Conference on Software Maintenance and Reengineering*, IEEE Comp. Soc., March 1997, Berlin, pp. 3-9
78. Jarzabek, S. "A Reuse Framework for Multi-Domain Software Development," *3rd Asia-Pacific Software Engineering Conference, APSEC'96*, IEEE Comp. Soc., Seoul, Korea, December 1996, pp.28-38
79. Huang, R. and Jarzabek, S. "PCS: A CASE Tool for Distributed Group Software Development," *Proc. International Federation for Information Processing Congress, IFIP'96, Advanced IT Tools*, Canberra, September 1996, UK: Chapman & Hall, pp. 402-410
80. Chee, C.L., Jarzabek, S. and Ramamoorthy, C.V. "An Intelligent Process for Formulating and Answering Project Queries," *Proc. 6th Int. Conference on Software Engineering and Knowledge Engineering, SEKE'96*, Nevada, USA, June 1996, pp. 309-316
81. Jarzabek, S. and Ling, T.W. "A conceptual model for business re-engineering methods and tools," *Proc. 14th Int. Conference on Object-Oriented and Entity-Relationship Modeling, OO-ER'95*, Queensland, Australia, Dec. 12-15, 1995, in *Lecture Notes in Computer Science*, Springer-Verlag, Germany, Dec. 1995, pp. 260-269
82. Jarzabek, S. "PQL: A language for specifying abstract program views," *Proc. 5th European Software Engineering Conference, ESEC'95*, Barcelona, September 1995, Lecture Notes in Computer Science, No. 989, Springer Verlag, pp. 324-342
83. Jarzabek, S and Tan, P.K. "Design of a Generic Reverse Engineering Assistant Tool," *Proc. 2nd Working Conference on Reverse Engineering, WCRE*, Toronto, Canada, July 14-16, 1995, IEEE Computer Society Press, Los Alamitos, USA, pp. 61-70
84. Jarzabek, S. and Ling, T.W. "Model-based Design of Tools for Business Understanding and Re-engineering," Appendix: *Proc. 2nd Working Conference on Reverse Engineering, WCRE*, Toronto, Canada, July 14-16, 1995, IEEE Computer Society Press, Los Alamitos, USA, pp. 324-333
85. Jarzabek, S. and Ling, T.W. "Model-based Design of Tools for Business Understanding and Re-engineering," *Proc. 7th Int. Workshop on Computer Aided Software Engineering, CASE'95*, Toronto, Canada, July 10-14, 1995, IEEE Computer Society Press, Los Alamitos, USA, pp. 328-337
86. Jarzabek, S. "Specifying Program Transformations with PQTL," *Proc. ICSE-17 Workshop on Program Transformations for Software Evolution*, 24 April 1995, Seattle, USA, ed. William Griswold, TRCS95-418, University of California, San Diego, pp. 35-46
87. Jarzabek, S. "From Reuse Library Experiences to Application Generation Architectures," *Proc. Symposium on Software Reusability, SSR'95*, Seattle, USA, April 28-30, 1995, ACM Press, pp. 114-122
88. Jarzabek, S., Shen, H. and Chan, H.C. A hybrid Program Knowledge Base system for Static Program Analyzers. *Proc. First Asia Pacific Software Engineering Conference, APSEC'94*, Tokyo, December 1994, IEEE Computer Society Press, Los Alamitos, USA, pp. 400-409

89. Jarzabek, S. "Systematic Design of Static Program Analyzers," *Proc. 18th Annual Int. Computer Software & Applications Conf. COMPSAC'94*, Taipei, November 9-11, 1994, IEEE Computer Society Press, Los Alamitos, USA, pp. 281-286
90. Jarzabek, S. and Lim, W.M. "Modeling in Strategic Reengineering," *Proc. 6th Int. Conference on Software Engineering and Knowledge Engineering*, Riga, Latvia, June 1994, published by Knowledge System Institute, USA, pp. 249-256
91. Jarzabek, S. and Tang, S.T. "Conceptual Modeling of Families of Software Systems," *Proc. 4th European-Japanese Seminar on Information Modeling and Knowledge Bases*, Stockholm, Sweden, May 31-June 3, 1994; chapter 19 in book *Information Modeling and Knowledge Bases VI*, Edts. H. Kangassalo, H. Jaakkola, S. Ohsuga and B. Wangler, IOS Press Amsterdam, 1995, pp. 299-312
92. Jarzabek, S. and Tan, C. L. "Modeling Multiple Views of Common Features in Software Reengineering for Reuse," *Proc. 6th Int. Conference on CAiSE'94*, Utrecht, Holland, June 1994; published in *Lecture Notes in Computer Science*, No. 811, *Advanced Information Systems Engineering*, Springer-Verlag, June 1994, pp. 269-282
93. Jarzabek, S., Tan, C.L. and Tham, K. "An Object-oriented Model for Recovered Designs in Software Reengineering," *Proc. of the InfoScience'93*, Seoul, Korea, October 1993, pp. 217-224
94. Jarzabek, S. "Software Reengineering for Reusability," *Proc. 17th Annual Int. Computer Software and Applications Conference COMPSAC93*, IEEE Computer Society, Phoenix, USA, November 1993, pp. 100-106
95. Jarzabek, S. "Strategic Reengineering of Software: Lifecycle Approach," *Proc. 6th Int. Workshop on CASE, CASE'93*, IEEE Computer Society, Singapore, July 1993, pp. 211-220
96. Tan, H.B., Ling, T.W., Jarzabek, S. and Ho, Y.S. "The Data Derivation Model: A Program Specification Technique that Improves Reusability," *Proc. 1993 ACM Symposium on Applied Computing*, (ACM), Indiana, Feb. 1993, pp. 95-102
97. Tan, W.G. and Jarzabek, S. "Current practices and Future Needs of Software Maintenance in Singapore," *Proc. SCS Silver Jubilee Conference on Software Engineering: New technologies & Business Payoffs*, Singapore, October 1992, pp. 121-135
98. Jarzabek, S. "Domain Model-Driven Software Reengineering," In *Workshop Notes of AAAI-92 AI & Automated Program Understanding*, July 1992, San Jose, California, pp. 72-75
99. Jarzabek, S. and Tham, K. "Towards Automating Software Maintenance," *Proc. 3rd International Conference CAiSE'91*, Trondheim, May 1991; published in *Lecture Notes in Computer Science*, No. 498, *Advanced Information System Engineering*, May 1991, Springer-Verlag, pp. 336-355
100. Jarzabek, S. "Software Maintenance with CASE," *Proc. CASE: The Next Generation*, Sydney, Digital Consulting, Inc, April 1991, pp. 1-24
101. Jarzabek, S. "From Object-Oriented Analysis and Design to Implementation," *Proc. International Conference on Object-Oriented Programming*, organized by Systems Education Centre, Singapore, February 1991, pp. 1-21
102. Jarzabek, S. "Towards Integration of CASE Back-End Tools," *Proc. CASE'90 Fourth International Workshop on CASE*, (IEEE Computer Society), December 1990, Irvine, pp. 14-15
103. Jarzabek, S. and Tan, C.L. "A Reusability Framework for Software Reengineering," *Proc. Joint Conference on Software Engineering, JCSE'93*, Fukuoka, Japan, November 1993, pp. 381-388
104. Jarzabek, S. "A Method for Specifying Form-Oriented User Interfaces," *Proc. International Computer Conference*, Taipei, Dec. 1988, vol. I, pp. 100-105
105. Jarzabek, S. "The Role of Specifications and Abstractions in the Design of a Software Environment Generation System," *Proc. International Computer Conference*, Taipei, Dec. 1988, vol. I, pp. 671-676
106. Jarzabek, S. "A Method for Specifying and Prototyping User Interfaces Based on the Structure Editor Technology," *Proc. Second International Conference on Human-Computer Interaction*, Hawaii, August 1987, p. 146 (abstract)
107. Jarzabek, S. "Software Environments for Development, Maintenance, and Reuse of Software Descriptions," *Proc. CIPS CONGRESS'87*, Winnipeg, Manitoba, May 1987, 201-208
108. Jarzabek, S. "Language-Independence of Project Information Bases," *Proc. Second Kansas Conference on Knowledge-Based Software Development*, Manhattan, Kansas, October 1986
109. Jarzabek, S. "Generation of Software Production Environments," *Proc. ACM Computer Science Conference*, (ACM), February 1986, Cincinnati (Research in Progress Abstract)
110. Jarzabek, S. "Introduction to the Compiler Production System," *Proc. of the INFOGRYF'80*, 1980, Kolobrzeg, Poland
111. Jarzabek, S. "Automatic Generation of Program Optimizers," *Proc. of the INFORMATICA'79*, 1979, Bled

#### **Short papers in refereed international conference proceedings**

112. Jarzabek, S. "Pragmatic strategies for variability management in software product lines," *Int. Conf. on Software Reuse, ICSR'2011*, S. Korea, June 2011, pp. 244-245
113. Jarzabek, S. "Pragmatic strategies for variability management in product lines in small- to medium-size companies," *Proc. 14th International Software Product Line Conference*, Jeju, S. Korea, Sept. 2010, pp. 503-504

114. Jarzabek, S. "Pragmatic strategies for variability management in product lines in small- to medium-size companies," *Proc. 13<sup>th</sup> International Software Product Line Conference*, San Francisco, Aug. 2009, p. 327
115. Jarzabek, S. "[Variability Management for Product Lines with XVCL](#)," *11<sup>th</sup> Int. Software Product Line Conf., SPLC'07*, Kyoto, Sept. 2007, pp. 13-14
116. Jarzabek, S. and Pettersson, U. "Research Journey Towards Industrial Application of Reuse Technique," *Int. Conf. Software Engineering, ICSE'06*, Shanghai, May 2006, pp. 608-611
117. Jarzabek, S. and Pettersson, U. "[Cost-Effective Engineering of Web Applications—Pragmatic Reuse: Building Web Application Product Lines](#)," *Int. Conf. Software Engineering, ICSE'06*, Shanghai, May 2006, pp. 1053-1054 (description of the tutorial presented at ICSE)
118. Sinson, R., Jarzabek, S., Ow, S.H., Rivepiboon, Nguyen, N.H "Software Practices in Five ASEAN Countries: An Exploratory Study," *Int. Conf. Software Engineering, ICSE'06*, Shanghai, May 2006, pp. 628-631
119. Jarzabek, S. and Pettersson, U. "Project-Driven University-Industry Collaboration: Modes of Collaboration, Outcomes, Benefits, Success Factors," *3rd Int. Summit on Software Engineering Education, SSEE'06*, Shanghai, May 2006, pp. 9-12
120. Jarzabek, S., Basset, P., Zhang, H. and Zhang, W. "[XVCL: XML-based Variant Configuration Language](#)," *Proc. Int. Conf. on Software Engineering, ICSE'03*, IEEE Comp. Soc., May 2003, Portland, pp. 810-811, also, formal presentation and open session demo of the XVCL system

### **Book chapters:**

1. Jarzabek, S. "Software Reuse Beyond Components with XVCL," *2<sup>nd</sup> Summer School on Generative and Transformational Techniques in Software Engineering, GTTSE'07*, Braga, Portugal, July 2007, LNCS 5235, Springer-Verlag Berlin Heidelberg, 2008, pp. 47-77
2. Zhang, W., Jarzabek, S., Zhang H., Loughran, N. and Rashid, A. "Software evolution with XVCL," chapter in *Software Evolution with UML and XML*, Idea Group Inc.
3. Cheong, Y.C. and Jarzabek, S. "Modeling Variant User Requirements in Domain Engineering for Reuse," in *Information Modelling and Knowledge Bases*, Eds. Hannu Jaakkola, Hannu Kangassalo and Eiji Kawaguchi, IOS Press, Netherlands, ISSN: 0922-6389, pp. 220-234
4. Jarzabek, S. and Tang, S.T. "Conceptual Modeling of Families of Software Systems," in book *Information Modeling and Knowledge Bases VI*, Edts. H. Kangassalo, H. Jaakkola, S. Ohsuga and B. Wangler, IOS Press Amsterdam, 1995, pp. 299-312
5. Jarzabek, S. "EPDS: An Educational Program Development System," in *The Information Edge: The Future for Educational Computing*, Rasmussen, Bruce (Editor), Brisbane, July 1985
6. Some of the conference papers have been published as chapters in Lecture Notes in Computer Science, Springer Verlag; they are listed in the Section "Conference papers".

### **Conference Organization and PC Member**

CSEE&T, Conference on Software Engineering Education and Training, May 22-24, 2011, co-located with ICSE'2011: Tutorial Chair, PC member

SPLC, Int. Software Product Line Conference: 2007-2011: PC member, Demo and Poster Chair, Doctoral Symposium panel member, Panelist (session on Quality of SPL assets); Co-Editor SPLC, 2010 proceedings

IWSC, Int. Workshop on Software Clones: 2009-2011, co-organizer and PC member

ICSR, Int. Conference on Software Reuse: 2005-2011, PC member, Tutorial Chair

GPCE Generative Programming and Component Engineering ([GPCE'06](#)): General Chair October 22-26, 2006, Portland, Oregon, co-located with OOPSLA; PC member

Co-organized Workshop on Implementation of Software Product Lines and Reusable Components, in conjunction with the 8<sup>th</sup> International Conference on Software Reuse (ICSR 8), July 5 to 9, 2004 in Madrid, Spain

I have been a PC member at WWW, Int. Conf. on Software Maintenance (ICSM), Int. Conference on Software Reuse (ICSR), SCAM, PEPM, Software product Line Conf (SPLC), QSI, VAMOS, Int. Workshop on System/SW Architectures, Int. Conf. on Software Engineering Advances, Int. Workshop On Evaluation Of Novel Approaches To Software Engineering, Int. Workshop on Web Site Evolution, Working Conference on Reverse Engineering

Guest Editor of a special issue on Teaching Software Project Courses in Forum for Advancing Software Engineering Education (FASE), an internet journal.

### **Invited talks and tutorials given at international conferences**

1. ½ day tutorial, Pragmatic Strategies for Variability Management in Software Product Lines in Small- to Medium-Size Companies, ICSR'2011, SPLC 2010, APSEC 2009, SPLC 2009, ICSR'2008

## Stan Jarzabek

2. ½ day tutorial: *Problems We Can Solve with Power-Generics*, at GPCE/OOPSLA Nashville, Oct. 2008
3. Full-day tutorial: *Variability Management for Product Lines with a Generative Technique*, at ICSR'08, Beijing, May 2008
4. ½ day tutorial: *Variability Management for Product Lines with a Generative Technique*, Software Product Line Conference, SPLC'07, Sept. 2007, Kyoto
5. Invited lectures at 2nd Summer School on Generative and Transformational Techniques in Software Engineering, GTTSE'07, Braga, Portugal, July 2007
6. ½ day tutorial: *Cost-Effective Engineering of Web Applications—Pragmatic Reuse: Building Web Applications*, at *Int. World Wide Web Conference, WWW'07*, Banff, May 2007
7. Full-day tutorial: *Cost-Effective Engineering of Web Applications—Pragmatic Reuse: Building Web Application Product Lines*, at *Int. Conf. Software Engineering, ICSE'06*, Shanghai, May 2006, pp. 1053-1054; speakers: Jarzabek, S. and Pettersson, U.
8. ½ tutorial on “Pragmatic approach to reuse-based Engineering of Web Applications” at [ESEC-FSE'05](#)
9. ½ day tutorial: *Cost-Effective Engineering of Web Applications*, at Int. Conf. on Web Engineering
10. ½ day tutorials on XVCL at ICSR July 04 and ICSM Sept 04.
11. “Using Clone Analysis to Improve Program Quality: CCFinder Experience,” keynote at the Software Engineering Workshop on Code Clone Detection Technique and its Applications, Cosponsored by Software Engineering Studio Project in Osaka University, IT forum OACIS, EASE project, Tokyo, March 15, 2005
12. “A Pragmatic Method for Enhanced Reusability and Evolution,” seminar at Toshiba, Tokyo, March 16, 2005
13. I have given seminars and tutorials at universities and for software professionals. During my sabbatical leave, I gave 14 invited seminars at universities in Europe and Canada.

### Public courses

- 2-day course: *Cost-Effective Engineering of Web Applications Pragmatic Reuse: Building Web Application Product Lines*, 15-16 February, 2006 in Kuala Lumpur and 20-21 February, 2006 in Singapore  
five 2-day courses on reuse and reengineering given in 1992-1996 in Singapore

### Reviews for Journals and Other Reviews

I review 30-50 papers per year for international conferences and journals and review research grant proposals for Jacquard program, Netherlands, and SIIRD Singapore-Israel R&D agency.

### Awards

1. ACM SIGSOFT distinguished paper award: Jarzabek, S. and Li, S. “[Eliminating Redundancies with a “Composition with Adaptation” Meta-programming Technique](#),” *Proc. ESEC-FSE'03, European Software Engineering Conference and ACM SIGSOFT Symposium on the Foundations of Software Engineering*, ACM Press, September 2003, Helsinki, pp. 237-246
2. Best papers on Software Engineering from *13<sup>th</sup> International Conference on Software Engineering and Knowledge Engineering, SEKE'01*, Knowledge System Institute, June 2001, Buenos Aires, Argentina: Stan Jarzabek, Wai Chun Ong and Hongyu Zhang “[Handling Variant Requirements in Domain Modeling](#),” *Journal of Software and Systems*, Vol. 68, Issue 3, 15 Dec. 2003, pp.171-182
3. Best papers from *16th Int. Conference on Software Engineering and Knowledge Engineering (SEKE'04)*: Jarzabek, S., Ru, S., Zhang, H. and Sun, Z. “Analysis of meta-programs: a case study,” *Proc. 16th Int. Conference on Software Engineering and Knowledge Engineering (SEKE'04)*, Banff, Canada, June 2004, pp. 68-73; to appear in a special issue of *Journal of Software and Systems*

### Research Grants

- |           |                                                                                                                                                                                                                                                                                                                                             |
|-----------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 2010-12   | NUS Research Grant, Flexible Generators: Applying Generators in Software Reuse and Evolution, S\$31,744                                                                                                                                                                                                                                     |
| 2009-11   | NUS Research Grant, New Tools for XVCL Workbench and Exploring Non-Software Application Domains for XVCL, S\$60,000                                                                                                                                                                                                                         |
| 2008-11   | NUS Research Grant, Towards a Model for Comparing and Evaluating Generative Techniques, S\$69,500                                                                                                                                                                                                                                           |
| 2005-09   | NUS Research Grant, <i>Meta-level parameterization and generation for enhanced genericity</i> , S\$200,000                                                                                                                                                                                                                                  |
| 2005-06   | NUS Research Grant, <i>Assessment and Dynamic Analysis of Meta-Programs</i> , S\$59,000                                                                                                                                                                                                                                                     |
| 2004-05   | NUS Research Grant, <i>Meta-programming: Elimination of Redundancies</i> , S\$61,000                                                                                                                                                                                                                                                        |
| 1999-2002 | Singapore-Ontario Joint Research Grant, NSTB, <i>Software Reuse Framework For Reliable Mission-Critical Systems</i> , S\$204,000 (Singapore component); competition was open to all the disciplines (engineering and science), with 1 project among 4 being selected for funding; our project was the only IT project that received funding |
| 1999-2003 | NUS Research Grant, <i>Engineering Variant Requirements in Component-based Software Product Lines</i> , S\$131,700                                                                                                                                                                                                                          |

## Stan Jarzabek

1995-99	NUS Research Grant, <i>Integrated Business-Software Evolution</i> , S\$58,500
1995-99	NUS Research Grant, <i>Multi-Domain Reuse Framework</i> , S\$106,500
1995-97	National Science and Technology Board, a grant for employment of a Postdoctoral Fellow
1992-95	NUS Research Grant, <i>Domain-driven Software Re-engineering</i> , S\$45,000
1986-89	National Science and Engineering Research Council, <i>Programming environments</i> , CND\$30,000
1984-85	Science & Engineering Research Board McMaster University, CND\$6,350

### **Teaching**

I like teaching courses that require active participation of students. I teach a course on product line approach to reuse [CS6201 Software Reuse](#) (PhD level), based on latest results published in the literature and my own research. I also teach level three software engineering project course.

I believe the role of universities is to teach students fundamental concepts, not over-emphasizing specific technologies which change so fast. Understanding fundamentals creates a reference point and shapes judgment that will help graduates adapt to changes throughout their careers. I specialize in teaching design in-the-large in the frame of team-based software project courses.

Design is one of the critical success factors in large software projects. Design is a basis for planning projects and achieving functional and quality software requirements. Poor design leads to rising costs of software evolution. What are the effective ways to teach advanced design concepts in university courses?

In my experience, design principles cannot be taught in abstraction. Design principles are best learned when students immediately apply them in practice and experience their benefits. Design principles matter only in projects of substantial size. Enforcing design in assignments is artificial and not effective: problems students work on are too small, most often programs can be “hacked”, so students can’t experience the real need and value of applying design principles.

It follows that project courses provide better basis to teach design than assignment-based courses. Two types of project courses are popular: real world projects done in industry attachment and projects supervised by various faculty members, focused on application domains of their expertise.

Such project courses enhance students’ skills in areas of requirement elicitation, dealing with fuzzy and incomplete requirements. Students get exposure to new application domains, industry practices, team work and software technologies. There is a great value of such project courses for students.

Still, the above project courses may face some difficulties to teach design know-how. The instructors and supervisors may not be able to scrutinize design deliverables (architecture, interfaces, and design decisions) at the level of details that is needed to effectively coach students.

In 1990s, industry often hinted at poor development skills of our graduates. I developed [CS3215 Software Engineering Project](#) course to address this challenge. The idea behind the course is that design is best taught by example: Most of the students must see good design and experience its benefits before they can come up with good design on their own. I identified model problem that is particularly useful in teaching design-in-the-large (just like stacks and queues serve the purpose of teaching abstract data types and design in-the-small). Working in teams of six, students do modular decomposition, architecture design and learn to specify component interfaces. This part is closely monitored by supervisors during weekly consultation sessions, to give students early feedback and opportunity to refine their design. Students follow up with implementation (10-15KLOC) in iterative development. We emphasize design for change, reliability and quality documentation. At the end, students present their projects, and we test programs using our regression testing tool comprising 400 test cases.

I developed courseware for the above project course in 2000 and have been teaching it ever since. We have been continuously refining the teaching method and tool infrastructure for the project course. Despite heavy load (project course is equivalent to two normal courses) students like the course. I observe huge difference in student’s ability to tackle the project work at the beginning and at the end of the course.

#### **Courses taught at NUS:**

[CS3215 Software Engineering Project](#) , year 3  
Object-Oriented Methods, year 3

Software Engineering, year 3  
Software Architecture, year 4  
Software Re-engineering, graduate course  
[TCS6201 Software Reuse](#), graduate course

***Courses taught at McMaster:***

Introduction to Computer Programming for Science, year 1  
Computer Architecture, year 1  
Data Structures, year 2  
Compiler Design, year 4  
Formal Language Theory, year 4  
Software Engineering, graduate course

**Service**

Member of International Advisory Committee for Cooperation between South-East Asia and European Union  
Curriculum Committee member (rep for software engineering)  
Member of Plagiarism Committee  
A judge in Singapore Science and Engineering Fair  
Coordinator of the Software Engineering Research Lab  
Member of IT Management Committee  
University representative to the Asian Universities Network  
Representative to Science Library  
Member of the Incubator Management Committee  
School Coordinator of Honours Year Projects  
School Editorial Committee Member  
Supervisor of the School's Technical Information Center  
School Representative to the Science Library Committee  
Member of the Curriculum Committee  
Academic Advisor, Software Engineering Area of Focus  
University Host to Exchange Students  
Member of the Social and Recreation Committee  
Founding member of Chopin Society Singapore

***at McMaster:***

Appointment Committee Member  
Technical Report Editor

--- The End ---