Miguel Velez

Publications

Books

[42] S. Apel, D. Batory, C. Kästner, and G. Saake. *Feature-Oriented Software Product Lines: Concepts and Implementation*. Berlin/Heidelberg, Germany: Springer-Verlag, 2013.

[41] A. Downey. *Think Bayes*. " O'Reilly Media, Inc.", 2013.

[40] T. Lindholm, F. Yellin, G. Bracha, and A. Buckley. *The Java Virtual Machine Specification, Java SE 7 Edition*. 1st. Addison-Wesley Professional, 2013.

[39] T. Parr. *The Definitive ANTLR 4 Reference*. 2nd. Pragmatic Bookshelf, 2013.

[38] A. Brown and G. Wilson. *The Architecture of Open Source Applications, Volume II*. The Architecture of Open Source Applications v. 2. Kristian Hermansen, 2012.

[37] A. V. Levitin. *Introduction to the Design and Analysis of Algorithms (3rd Edition)*. Boston, MA, USA: Addison-Wesley, 2012.

[36] J. Mongan, N. Suojanen, and E. Giguere. *Programming Interviews Exposed*. Wiley Publishing, Inc., 2012.

[35] K. P. Murphy. *Machine Learning: A Probabilistic Perspective*. MIT Press, 2012.

[34] A. Brown and G. Wilson. *The Architecture of Open Source Applications, Volume I*. The Architecture of Open Source Applications v. 1. CreativeCommons, 2011.

[33] M. Odersky, L. Spoon, and B. Venners. *Programming in Scala: A Comprehensive Step-by-Step Guide, 2nd Edition*. 2nd. USA: Artima Incorporation, 2011.

[32] I. H. Witten, E. Frank, and M. A. Hall. *Data Mining: Practical Machine Learning Tools and Techniques*. 3rd. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2011.

[31] F. P. Brooks. *The Design of Design: Essays from a Computer Scientist*. 1st. Addison-Wesley Professional, 2010.

[30] F. Nielson, H. R. Nielson, and C. Hankin. *Principles of Program Analysis*. Springer Publishing Company, Incorporated, 2010.

[29] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms, Third Edition*. 3rd. MIT Press, 2009.

[28] T. Parr. *Language Implementation Patterns: Create Your Own Domain-Specific and General Programming Languages*. 1st. Pragmatic Bookshelf, 2009.

[27] R. Yin. *Case Study Research: Design and Methods*. 4th. Applied Social Research Methods. SAGE Publications, 2009.

[26] M. Herlihy and N. Shavit. *The Art of Multiprocessor Programming*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2008.

[25] T. Parr. *The Definitive ANTLR Reference: Building Domain-Specific Languages*. Pragmatic Bookshelf, 2007.

[24] F. Shull, J. Singer, and D. I. Sjøberg. *Guide to Advanced Empirical Software Engineering*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2007.

[23] A. V. Aho, M. S. Lam, R. Sethi, and J. D. Ullman. *Compilers: Principles, Techniques, and Tools (2nd Edition)*. Boston, MA, USA: Addison-Wesley, 2006.

[22] C. M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2006.

[21] P. Feiler, K. Sullivan, K. Wallnau, R. Gabriel, J. Goodenough, R. Linger, T. Longstaff, R. Kazman, M. Klein, L. Northrop, and D. Schmidt. *Ultra-Large-Scale Systems: The Software Challenge of the Future*. Software Engineering Institute, Carnegie Mellon University, 2006.

[20] D. C. Montgomery. *Design and Analysis of Experiments*. John Wiley & Sons, 2006.

[19] S. Krishnamurthi. *Programming Languages: Application and Interpretation*. e-book, 2003.

[18] R. Laddad. *AspectJ in Action: Practical Aspect-Oriented Programming*. Greenwich, CT, USA: Manning Publications, 2003.

[17] P. Clements, B. Felix, L. Bass, D. Garlan, J. Ivers, R. Little, P. Merson, R. Nord, and J. Stafford. *Documenting Software Architectures: Views and Beyond*. Pearson Education, 2002.

[16] P. Clements and L. Northrop. *Software Product Lines*. Addison-Wesley, 2002.

[15] D. J. C. MacKay. *Information Theory, Inference & Learning Algorithms*. New York, NY, USA: Cambridge University Press, 2002.

[14] A. Shalloway and J. R. Trott. *Design Patterns Explained: A New Perspective on Object-Oriented Design*. Boston, MA, USA: Addison-Wesley, 2002.

[13] C. Szyperski. *Component Software: Beyond Object-Oriented Programming*. 2nd. Boston, MA, USA: Addison-Wesley, 2002.

[12] K. Czarnecki and U. Eisenecker. *Generative Programming: Methods, Tools, and Applications*. New York, NY, USA: ACM Press/Addison-Wesley, 2000.

[11] D. E. Knuth. *The Art of Computer Programming, Volume 3: (2Nd Ed.) Sorting and Searching*. Redwood City, CA, USA: Addison-Wesley, 1998.

[10] D. E. Knuth. *The Art of Computer Programming, Volume 1 (3rd Ed.): Fundamental Algorithms*. Redwood City, CA, USA: Addison-Wesley, 1997.

[9] D. E. Knuth. *The Art of Computer Programming, Volume 2 (3rd Ed.): Seminumerical Algorithms*. Boston, MA, USA: Addison-Wesley, 1997.

[8] T. M. Mitchell. *Machine Learning*. 1st. New York, NY, USA: McGraw-Hill, Inc., 1997.

[7] H. Abelson and G. J. Sussman. *Structure and Interpretation of Computer Programs*. 2nd. Cambridge, MA, USA: MIT Press, 1996.

[6] M. Shaw and D. Garlan. *Software Architecture: Perspectives on an Emerging Discipline*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1996.

[5] H. A. Simon. *The Sciences of the Artificial (3rd Ed.)* Cambridge, MA, USA: MIT Press, 1996.

[4] F. P. Brooks Jr. *The Mythical Man-Month*. anniversary. Boston, MA, USA: Addison-Wesley, 1995.

[3] E. Gamma, R. Helm, R. Johnson, and J. Vlissides. *Design Patterns: Elements of Reusable Object-oriented Software*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1995.

[2] D. P. Siewiorek, C. G. Bell, and A. Newell. *Computer structures: principles and examples*. McGraw-Hill computer science series. McGraw-Hill, 1987. Chap. 2, Levels and Abstraction.

[1] P. Naur and B. Randell. *Software Engineering: Report of a Conference Sponsored by the NATO Science Committee, Garmisch, Germany, 7-11 Oct. 1968, Brussels, Scientific Affairs Division, NATO*. NATO, 1969.

Refereed Journal Articles

[48] N. Polikarpova, J. Yang, S. Itzhaky, and A. Solar-Lezama. "Type-Driven Repair for Information Flow Security". In *CoRR* abs/1607.03445 (2016).

[47] B. A. Myers, A. J. Ko, T. D. LaToza, and Y. Yoon. "Programmers Are Users Too: Human-Centered Methods for Improving Programming Tools". In *Computer* 49.7 (July 2016), pp. 44–52.

[46] B. A. Myers and J. Stylos. "Improving API Usability". In *Commun. ACM* 59.6 (May 2016), pp. 62–69.

[45] J. Bell and G. Kaiser. "Phosphor: Illuminating Dynamic Data Flow in Commodity Jvms". In *SIGPLAN Notices* 49.10 (Oct. 2014), pp. 83–101.

[44] J. M. Barnes, D. Garlan, and B. Schmerl. "Evolution Styles: Foundations and Models for Software Architecture Evolution". In *Softw. Syst. Model. (SoSyM)* 13.2 (May 2014), pp. 649–678.

[43] F. Hutter, L. Xu, H. H. Hoos, and K. Leyton-Brown. "Algorithm Runtime Prediction: Methods & Evaluation". In *Artif. Intell.* 206 (Jan. 2014), pp. 79–111.

[42] N. Siegmund, M. Rosenmüller, M. Kuhlemann, C. Kästner, S. Apel, and G. Saake. "SPL Conqueror: Toward Optimization of Non-functional Properties in Software Product Lines". In *Software Quality Journal* 20.3-4 (Sept. 2012), pp. 487–517.

[41] H. H. Hoos. "Programming by Optimization". In *Commun. ACM* 55.2 (Feb. 2012), pp. 70–80.

[40] T. Ball, V. Levin, and S. K. Rajamani. "A Decade of Software Model Checking with SLAM". In *Commun. ACM* 54.7 (July 2011), pp. 68–76.

[39] G. Bell and D. P. Siewiorek. "The Book Computer Structures: Thoughts After 40 Years". In *IEEE Ann. Hist. Comput.* 33.2 (Apr. 2011), pp. 89–95.

[38] A. Bessey, K. Block, B. Chelf, A. Chou, B. Fulton, S. Hallem, C. Henri-Gros, A. Kamsky, S. McPeak, and D. Engler. "A Few Billion Lines of Code Later: Using Static Analysis to Find Bugs in the Real World". In *Commun. ACM* 53.2 (Feb. 2010), pp. 66–75.

[37] N. Ayewah, D. Hovemeyer, J. D. Morgenthaler, J. Penix, and W. Pugh. "Using Static Analysis to Find Bugs". In *IEEE Softw.* 25.5 (Sept. 2008), pp. 22–29.

[36] A. Nhlabatsi, R. Laney, and B. Nuseibeh. "Feature Interaction: The Security Threat from within Software Systems". In *Progress in Informatics* 5 (Mar. 2008), pp. 75–89.

[35] W. Visser. "Designing as construction of representations: A dynamic viewpoint in cognitive design research". In *Human–Computer Interaction* 21.1 (Dec. 2006), pp. 103–152.

[34] B. Schmerl, J. Aldrich, D. Garlan, R. Kazman, and H. Yan. "Discovering Architectures from Running Systems". In *IEEE Trans. Softw. Eng. (TSE)* 32.7 (July 2006), pp. 454–466.

[33] D. P. Siewiorek and P. Narasimhan. “Fault-tolerant architectures for space and avionics applications”. In *NASA Ames Research* (2005).

[32] N. Ducheneaut. “Socialization in an Open Source Software Community: A Socio-Technical Analysis”. In *Comput. Supported Coop. Work* 14.4 (Aug. 2005), pp. 323–368.

[31] J. F. Maranzano, S. A. Rozsypal, G. H. Zimmerman, G. W. Warnken, P. E. Wirth, and D. M. Weiss. “Architecture Reviews: Practice and Experience”. In *IEEE Softw.* 22.2 (Mar. 2005), pp. 34–43.

[30] D. Garlan, S.-W. Cheng, A.-C. Huang, B. Schmerl, and P. Steenkiste. “Rainbow: Architecture-Based Self-Adaptation with Reusable Infrastructure”. In *Computer* 37.10 (Oct. 2004), pp. 46–54.

[29] D. Batory, J. N. Sarvela, and A. Rauschmayer. “Scaling step-wise refinement”. In *IEEE Trans. Softw. Eng. (TSE)* 30.6 (June 2004), pp. 355–371.

[28] D. P. Siewiorek, R. Chillarege, and Z. T. Kalbarczyk. “Reflections on Industry Trends and Experimental Research in Dependability”. In *IEEE Trans. Dependable Secur. Comput.* 1.2 (Apr. 2004), pp. 109–127.

[27] G. Tassey. “The economic impacts of inadequate infrastructure for software testing”. In *National Institute of Standards and Technology, RTI Project* 7007.011 (2002).

[26] A. Mockus, R. T. Fielding, and J. D. Herbsleb. “Two Case Studies of Open Source Software Development: Apache and Mozilla”. In *ACM Trans. Softw. Eng. Methodol. (TOSEM)* 11.3 (July 2002), pp. 309–346.

[25] A. Zeller. “Yesterday, My Program Worked. Today, It Does Not. Why?” In *SIGSOFT Softw. Eng. Notes* 24.6 (Oct. 1999), pp. 253–267.

[24] G. C. Murphy and D. Notkin. “Reengineering with Reflexion Models: A Case Study”. In *Computer* 30.8 (Aug. 1997), pp. 29–36.

[23] J. Wang and C. J. Wu. “A hidden projection property of Plackett-Burman and related designs”. In *Statistica Sinica* (1995), pp. 235–250.

[22] D. Garlan, R. Allen, and J. Ockerbloom. “Architectural Mismatch: Why Reuse Is So Hard”. In *IEEE Softw.* 12.6 (Nov. 1995), pp. 17–26.

[21] R. E. Kraut and L. A. Streeter. “Coordination in Software Development”. In *Commun. ACM* 38.3 (Mar. 1995), pp. 69–81.

[20] V. R. B.-G. Caldiera and H. D. Rombach. “Goal question metric paradigm”. In *Encyclopedia of Software Engineering* 1 (1994), pp. 528–532.

[19] G. D. Gopen and J. A. Swan. “The science of scientific writing”. In *American Scientist* 78.6 (1990), pp. 550–558.

[18] A. Hall. “Seven Myths of Formal Methods”. In *IEEE Softw.* 7.5 (Sept. 1990), pp. 11–19.

[17] B. Curtis, H. Krasner, and N. Iscoe. “A Field Study of the Software Design Process for Large Systems”. In *Commun. ACM* 31.11 (Nov. 1988), pp. 1268–1287.

[16] R. Johnson and B. Foote. “Designing Reusable Classes”. In *Journal of Object-Oriented Programming SIGS* 1.5 (June 1988), pp. 22–35.

[15] B. W. Boehm. “A Spiral Model of Software Development and Enhancement”. In *Computer* 21.5 (May 1988), pp. 61–72.

[14] W. S. Humphrey. “Characterizing the Software Process: A Maturity Framework”. In *IEEE Softw.* 5.2 (Mar. 1988), pp. 73–79.

[13] D. Harel. “Statecharts: A Visual Formalism for Complex Systems”. In *Sci. Comput. Program.* 8.3 (June 1987), pp. 231–274.

[12] F. P. Brooks Jr. “No Silver Bullet Essence and Accidents of Software Engineering”. In *Computer* 20.4 (Apr. 1987), pp. 10–19.

[11] E. J. Weyuker. “Axiomatizing Software Test Data Adequacy”. In *IEEE Trans. Softw. Eng. (TSE)* 12.12 (Dec. 1986), pp. 1128–1138.

[10] E. M. Clarke, E. A. Emerson, and A. P. Sistla. “Automatic Verification of Finite-state Concurrent Systems Using Temporal Logic Specifications”. In *ACM Trans. Program. Lang. Syst.* 8.2 (Apr. 1986), pp. 244–263.

[9] M. Shaw. “The impact of abstraction concerns on modern programming languages”. In *Proc. of the IEEE* 68.9 (Apr. 1980), pp. 1119–1130.

[8] R. A. De Millo, R. J. Lipton, and A. J. Perlis. “Social Processes and Proofs of Theorems and Programs”. In *Commun. ACM* 22.5 (May 1979), pp. 271–280.

[7] J. C. King. “Symbolic Execution and Program Testing”. In *Commun. ACM* 19.7 (July 1976), pp. 385–394.

[6] W. A. Wulf, R. L. London, and M. Shaw. “An Introduction to the Construction and Verification of Alphard Programs”. In *IEEE Trans. Softw. Eng. (TSE)* 2.4 (July 1976), pp. 253–265.

[5] F. DeRemer and H. Kron. “Programming-in-the-Large Versus Programming-in-the-Small”. In *IEEE Trans. Softw. Eng. (TSE)* SE-2.2 (June 1976), pp. 80–86.

[4] H. W. J. Rittel and M. M. Webber. “Dilemmas in a general theory of planning”. In *Policy Sciences* 4.2 (June 1973), pp. 155–169.

[3] C. A. R. Hoare. “Proof of correctness of data representations”. In *Acta Informatica* 1.4 (Dec. 1972), pp. 271–281.

[2] D. L. Parnas. “On the Criteria to Be Used in Decomposing Systems into Modules”. In *Commun. ACM* 15.12 (Dec. 1972), pp. 1053–1058.

[1] M. E. Conway. “How do committees invent?” In *Datamation* 14.4 (1968), pp. 28–31.

Refereed Conference Publications

[87] V. Nair, T. Menzies, N. Siegmund, and S. Apel. "Using Bad Learners to find Good Configurations". In *Proc. Conference*. Washington, DC, USA: IEEE Computer Society, July 2017.

[86] P. Jamshidi, M. Velez, C. Kästner, N. Siegmund, and P. Kawthekar. "Transfer Learning for Improving Model Predictions in Highly Configurable Software". In *Proc. Int'l Symp. Software Engineering for Adaptive and Self-Managing Systems (SEAMS)*. Buenos Aires, Argentina: IEEE Computer Society, May 2017. Accepted for publication. (23% acceptance rate).

[85] E. Kang, A. Milicevic, and D. Jackson. "Multi-representational Security Analysis". In *Proc. Int'l Symp. Foundations of Software Engineering (FSE)*. Seattle, WA, USA: ACM, Nov. 2016, pp. 181–192.

[84] T. Nguyen, T. Koc, J. Cheng, J. S. Foster, and A. A. Porter. "iGen Dynamic Interaction Inference for Configurable Software". In *Proc. Int'l Symp. Foundations of Software Engineering (FSE)*. Seattle, WA, USA: IEEE Computer Society, Nov. 2016.

[83] T. Xu, X. Jin, P. Huang, Y. Zhou, S. Lu, L. Jin, and S. Pasupathy. "Early Detection of Configuration Errors to Reduce Failure Damage". In *Proc. Conf.Operating Systems Design and Implementation (OSDI)*. Savannah, GA, USA: USENIX Association, Nov. 2016, pp. 619–634.

[82] L. Braz, R. Gheyi, M. Mongiovi, M. Ribeiro, F. Medeiros, and L. Teixeira. "A Change-centric Approach to Compile Configurable Systems with #Ifdefs". In *Proc. Int'l Conf. Generative Programming and Component Engineering (GPCE)*. Amsterdam, Netherlands: ACM, Oct. 2016, pp. 109–119.

[81] S. P. De Rosso and D. Jackson. "Purposes, Concepts, Misfits, and a Redesign of Git". In *Proc. Int'l Conf. Object-Oriented Programming, Systems, Languages and Applications (OOPSLA)*. Amsterdam, Netherlands: ACM, Oct. 2016, pp. 292–310.

[80] M. Al-Hajjaji, S. Krieter, T. Thüm, M. Lochau, and G. Saake. "IncLing: Efficient Product-line Testing Using Incremental Pairwise Sampling". In *Proc. Int'l Conf. Generative Programming and Component Engineering (GPCE)*. Amsterdam, Netherlands: ACM, Oct. 2016, pp. 144–155.

[79] V. Rothberg, C. Dietrich, A. Ziegler, and D. Lohmann. "Towards Scalable Configuration Testing in Variable Software". In *Proc. Int'l Conf. Generative Programming and Component Engineering (GPCE)*. Amsterdam, Netherlands: ACM, Oct. 2016, pp. 156–167.

[78] J. Meinicke, C.-P. Wong, C. Kästner, T. Thüm, and G. Saake. "On Essential Configuration Complexity: Measuring Interactions in Highly-configurable Systems". In *Proc. Int'l Conf. Automated Software Engineering (ASE)*. Singapore, Singapore: ACM, Sept. 2016, pp. 483–494.

[77] L. Wei, Y. Liu, and S.-C. Cheung. "Taming Android Fragmentation: Characterizing and Detecting Compatibility Issues for Android Apps". In *Proc. Int'l Conf. Automated Software Engineering (ASE)*. Singapore, Singapore: ACM, Sept. 2016, pp. 226–237.

[76] M. Velez, J. Sawin, A. Ingerson, and D. Chiu. “Improving Bitmap Execution Performance Using Column-Based Metadata”. In *Int’l Conf. Future Internet of Things and Cloud (FiCloud)*. Vienna, Austria: IEEE Computer Society, Aug. 2016, pp. 371–378. (30% acceptance rate).

[75] P. Kawthekar and C. Kästner. “Sensitivity Analysis For Building Evolving & Adaptive Robotic Software”. In *Proc. Workshop Autonomous Mobile Service Robots (WSR)*. New York, NY, USA, July 2016.

[74] M. Dhok, M. K. Ramanathan, and N. Sinha. “Type-aware Concolic Testing of JavaScript Programs”. In *Proc. Int’l Conf. Software Engineering (ICSE)*. Austin, TX, USA: ACM, May 2016, pp. 168–179.

[73] P. Thongtanunam, S. McIntosh, A. E. Hassan, and H. Iida. “Revisiting Code Ownership and Its Relationship with Software Quality in the Scope of Modern Code Review”. In *Proc. Int’l Conf. Software Engineering (ICSE)*. Austin, TX, USA: ACM, May 2016, pp. 1039–1050.

[72] B. Vasilescu, K. Blincoe, Q. Xuan, C. Casalnuovo, D. Damian, P. Devanbu, and V. Filkov. “The Sky is Not the Limit: Multitasking Across GitHub Projects”. In *Proc. Int’l Conf. Software Engineering (ICSE)*. Austin, TX, USA: ACM, May 2016, pp. 994–1005.

[71] A. S. Buyukkayhan, K. Onarlioglu, W. Robertson, and E. Kirda. “CrossFire: An Analysis of Firefox Extension-Reuse Vulnerabilities”. In *Network and Distributed System Security Symposium (NDSS)*. San Diego, CA, USA: Internet Society, Feb. 2016.

[70] A. C. Bahnsen, D. Aouada, A. Stojanovic, and B. Ottersten. “Detecting Credit Card Fraud Using Periodic Features”. In *Int’l Conf. Machine Learning and Applications (ICMLA)*. Miami, FL, USA: IEEE Press, Dec. 2015, pp. 208–213.

[69] F. Lv, H. Zhang, J.-g. Lou, S. Wang, D. Zhang, and J. Zhao. “CodeHow: Effective Code Search Based on API Understanding and Extended Boolean Model”. In *Proc. Int’l Conf. Automated Software Engineering (ASE)*. Washington, DC, USA: IEEE Computer Society, Nov. 2015, pp. 260–270.

[68] A. Sarkar, J. Guo, N. Siegmund, S. Apel, and K. Czarnecki. “Cost-Efficient Sampling for Performance Prediction of Configurable Systems”. In *Proc. Int’l Conf. Automated Software Engineering (ASE)*. Washington, DC, USA: IEEE Computer Society, Nov. 2015, pp. 342–352.

[67] M. Eichberg, B. Hermann, M. Mezini, and L. Glanz. “Hidden Truths in Dead Software Paths”. In *Proc. Europ. Software Engineering Conf. Foundations of Software Engineering (ESEC/FSE)*. Bergamo, Italy: ACM, Aug. 2015, pp. 474–484.

[66] N. Siegmund, A. Grebhahn, S. Apel, and C. Kästner. “Performance-influence Models for Highly Configurable Systems”. In *Proc. Europ. Software Engineering Conf. Foundations of Software Engineering (ESEC/FSE)*. Bergamo, Italy: ACM, Aug. 2015, pp. 284–294.

[65] T. Xu, L. Jin, X. Fan, Y. Zhou, S. Pasupathy, and R. Talwadker. “Hey, You Have Given Me Too Many Knobs!: Understanding and Dealing with Over-designed Configuration in System Software”. In *Proc. Int’l Symp. Foundations of Software Engineering (FSE)*. Bergamo, Italy: ACM, Aug. 2015, pp. 307–319.

[64] C. Henard, M. Papadakis, M. Harman, and Y. Le Traon. "Combining Multi-objective Search and Constraint Solving for Configuring Large Software Product Lines". In *Proc. Int'l Conf. Software Engineering (ICSE)*. Florence, Italy: IEEE Press, May 2015, pp. 517–528.

[63] S. Elbaum, G. Rothermel, and J. Penix. "Techniques for Improving Regression Testing in Continuous Integration Development Environments". In *Proc. Int'l Symp. Foundations of Software Engineering (FSE)*. Hong Kong, China: ACM, Nov. 2014, pp. 235–245.

[62] F. Anon, V. Navarathinarasah, M. Hoang, and C.-H. Lung. "Building a Framework for Internet of Things and Cloud Computing". In *Proc. Int'l Conf. Internet of Things (iThings)*. Taipei, Taiwan: IEEE Computer Society, Sept. 2014, pp. 132–139.

[61] M. Lillack, C. Kästner, and E. Bodden. "Tracking Load-time Configuration Options". In *Proc. Int'l Conf. Automated Software Engineering (ASE)*. Vasteras, Sweden: ACM, Sept. 2014, pp. 445–456.

[60] S. Arzt, S. Rasthofer, C. Fritz, E. Bodden, A. Bartel, J. Klein, Y. Le Traon, D. Octeau, and P. McDaniel. "FlowDroid: Precise Context, Flow, Field, Object-sensitive and Lifecycle-aware Taint Analysis for Android Apps". In *Proc. Conf. Programming Language Design and Implementation (PLDI)*. Edinburgh, UK: ACM, June 2014, pp. 259–269.

[59] D. Garlan. "Software Architecture: A Travelogue". In *Proc. Future of Software Engineering (FOSE)*. Hyderabad, India: ACM, June 2014, pp. 29–39.

[58] J. Guo, K. Czarnecki, S. Apel, N. Siegmund, and A. Wąsowski. "Variability-aware performance prediction: A statistical learning approach". In *Proc. Int'l Conf. Automated Software Engineering (ASE)*. IEEE Computer Society. Silicon Valley, CA, USA: ACM, Nov. 2013, pp. 301–311.

[57] T. Xu, J. Zhang, P. Huang, J. Zheng, T. Sheng, D. Yuan, Y. Zhou, and S. Pasupathy. "Do Not Blame Users for Misconfigurations". In *Proc. Symp. Operating Systems Principles*. Farminton, PA, USA: ACM, Nov. 2013, pp. 244–259.

[56] N. Siegmund, N. von Rhein, and S. Apel. "Family-Based Performance Measurement". In *Proc. Int'l Conf. Generative Programming and Component Engineering (GPCE)*. Indianapolis, IN, USA: ACM, Oct. 2013, pp. 95–104.

[55] M. Böhme, B. C. d. S. Oliveira, and A. Roychoudhury. "Regression Tests to Expose Change Interaction Errors". In *Proc. Int'l Symp. Foundations of Software Engineering (FSE)*. Saint Petersburg, Russia: ACM, Aug. 2013, pp. 334–344.

[54] C. H. P. Kim, D. Marinov, S. Khurshid, D. Batory, S. Souto, P. Barros, and M. d'Amorim. "SPLat: Lightweight Dynamic Analysis for Reducing Combinatorics in Testing Configurable Systems". In *Proc. Europ. Software Engineering Conf. Foundations of Software Engineering (ESEC/FSE)*. Saint Petersburg, Russia: ACM, Aug. 2013, pp. 257–267.

[53] S. Hao, D. Li, W. G. J. Halfond, and R. Govindan. "Estimating Mobile Application Energy Consumption Using Program Analysis". In *Proc. Int'l Conf. Software Engineering (ICSE)*. San Francisco, CA, USA: IEEE Press, May 2013, pp. 92–101.

[52] A. Niknafs and D. M. Berry. "The impact of domain knowledge on the effectiveness of requirements idea generation during requirements elicitation". In *Proc. Int'l Requirements Engineering Conf. (RE)*. Chicago, IL, USA: IEEE Computer Society, Sept. 2012, pp. 181–190.

[51] D. Alrajeh, J. Kramer, A. v. Lamsweerde, A. Russo, and S. Uchitel. "Generating Obstacle Conditions for Requirements Completeness". In *Proc. Int'l Conf. Software Engineering (ICSE)*. Zurich, Switzerland: IEEE Press, June 2012, pp. 705–715.

[50] M. Greiler, A. v. Deursen, and M.-A. Storey. "Test Confessions: A Study of Testing Practices for Plug-in Systems". In *Proc. Int'l Conf. Software Engineering (ICSE)*. Zurich, Switzerland: IEEE Press, June 2012, pp. 244–254.

[49] A. Hindle, E. T. Barr, Z. Su, M. Gabel, and P. Devanbu. "On the Naturalness of Software". In *Proc. Int'l Conf. Software Engineering (ICSE)*. Zurich, Switzerland: IEEE Press, June 2012, pp. 837–847.

[48] C. Le Goues, M. Dewey-Vogt, S. Forrest, and W. Weimer. "A Systematic Study of Automated Program Repair: Fixing 55 out of 105 Bugs for $8 Each". In *Proc. Int'l Conf. Software Engineering (ICSE)*. Zurich, Switzerland: IEEE Press, June 2012, pp. 3–13.

[47] D. Garlan, V. Dwivedi, I. Ruchkin, and B. Schmerl. "Foundations and Tools for End-user Architecting". In *Proc. Conf. Large-Scale Complex IT Systems: Development, Operation and Management*. Oxford, UK: Springer-Verlag, Mar. 2012, pp. 157–182.

[46] L. Dabbish, C. Stuart, J. Tsay, and J. Herbsleb. "Social Coding in GitHub: Transparency and Collaboration in an Open Software Repository". In *Proc. Conf. Computer Supported Cooperative Work (CSCW)*. Seattle, WA, USA: ACM, Feb. 2012, pp. 1277–1286.

[45] J. Sunshine, K. Naden, S. Stork, J. Aldrich, and É. Tanter. "First-class State Change in Plaid". In *Proc. Int'l Conf. Object-Oriented Programming, Systems, Languages and Applications (OOPSLA)*. Portland, OR, USA: ACM, Oct. 2011, pp. 713–732.

[44] Z. Yin, D. Yuan, Y. Zhou, S. Pasupathy, and L. Bairavasundaram. "How Do Fixes Become Bugs?" In *Proc. Europ. Software Engineering Conf. Foundations of Software Engineering (ESEC/FSE)*. Szeged, Hungary: ACM, Sept. 2011, pp. 26–36.

[43] A. Zeller, T. Zimmermann, and C. Bird. "Failure is a Four-letter Word: A Parody in Empirical Research". In *Int'l Conf. Predictive Models in Software Engineering*. Banff, Canada: ACM, Sept. 2011, 5:1–5:7.

[42] C. Parnin and A. Orso. "Are Automated Debugging Techniques Actually Helping Programmers?" In *Proc. Int'l Symp. Software Testing and Analysis (ISSTA)*. Toronto, Canada: ACM, July 2011, pp. 199–209.

[41] A. Rabkin and R. Katz. "Static Extraction of Program Configuration Options". In *Proc. Int'l Conf. Software Engineering (ICSE)*. Waikiki, Honolulu, HI, USA: ACM, May 2011, pp. 131–140.

[40] A. Bhave, B. Krogh, D. Garlan, and B. Schmerl. "View Consistency in Architectures for Cyber-Physical Systems". In *Int'l Conf. Cyber-Physical Systems (ICCPS)*. Chicago, IL, USA: IEEE Computer Society Press, Apr. 2011, pp. 151–160.

[39] H. Hoffmann, S. Sidiroglou, M. Carbin, S. Misailovic, A. Agarwal, and M. Rinard. "Dynamic Knobs for Responsive Power-aware Computing". In *Proc. Int'l Conf. Architectural Support for Programming Languages and Operating Systems (ASPLOS)*. Newport Beach, CA, USA: ACM, Mar. 2011, pp. 199–212.

[38] C. H. P. Kim, D. S. Batory, and S. Khurshid. "Reducing Combinatorics in Testing Product Lines". In *Proc. Int'l Conf. Aspect-Oriented Software Development (AOSD)*. Porto de Galinhas, Brazil: ACM, Mar. 2011, pp. 57–68.

[37] F. Hutter, H. H. Hoos, and K. Leyton-Brown. "Sequential Model-based Optimization for General Algorithm Configuration". In *Proc. Int'l Conf. Learning and Intelligent Optimization*. Rome, Italy: Springer-Verlag, Jan. 2011, pp. 507–523.

[36] W. Enck, P. Gilbert, B.-G. Chun, L. P. Cox, J. Jung, P. McDaniel, and A. N. Sheth. "TaintDroid: An Information-flow Tracking System for Realtime Privacy Monitoring on Smartphones". In *Proc. Conf. Operating Systems Design and Implementation (OSDI)*. Vancouver, BC, Canada: USENIX Association, 2010, pp. 393–407.

[35] N. Maiden, S. Jones, K. Karlsen, R. Neill, K. Zachos, and A. Milne. "Requirements Engineering As Creative Problem Solving: A Research Agenda for Idea Finding". In *Proc. Int'l Requirements Engineering Conf. (RE)*. Sydney, Australia: IEEE Computer Society, Sept. 2010, pp. 57–66.

[34] P. Sawyer, N. Bencomo, J. Whittle, E. Letier, and A. Finkelstein. "Requirements-Aware Systems: A Research Agenda for RE for Self-adaptive Systems". In *Proc. Int'l Requirements Engineering Conf. (RE)*. Sydney, Australia: IEEE Computer Society, Sept. 2010, pp. 95–103.

[33] E. J. Schwartz, T. Avgerinos, and D. Brumley. "All You Ever Wanted to Know About Dynamic Taint Analysis and Forward Symbolic Execution (but Might Have Been Afraid to Ask)". In *Proc. Symp. Security and Privacy (SP)*. Oakland, CA, USA: IEEE Computer Society, May 2010, pp. 317–331.

[32] D. F. Sutherland and W. L. Scherlis. "Composable Thread Coloring". In *Proc. Symp. Principles and Practice of Parallel Programming*. Bangalore, India: ACM, Jan. 2010, pp. 233–244.

[31] J. Aranda and G. Venolia. "The Secret Life of Bugs: Going Past the Errors and Omissions in Software Repositories". In *Proc. Int'l Conf. Software Engineering (ICSE)*. Vancouver, Canada: IEEE Computer Society, May 2009, pp. 298–308.

[30] U. Dekel and J. D. Herbsleb. "Improving API Documentation Usability with Knowledge Pushing". In *Proc. Int'l Conf. Software Engineering (ICSE)*. Vancouver, Canada: IEEE Computer Society Press, May 2009, pp. 320–330.

[29] H. Müller, M. Pezzè, and M. Shaw. "Visibility of Control in Adaptive Systems". In *Proc. Int'l Workshop Ultra-Large-Scale Software-Intensive Systems (ULSSIS)*. Leipzig, Germany: ACM, May 2008, pp. 23–26.

[28] N. Nagappan, B. Murphy, and V. Basili. "The Influence of Organizational Structure on Software Quality: An Empirical Case Study". In *Proc. Int'l Conf. Software Engineering (ICSE)*. Leipzig, Germany: ACM, May 2008, pp. 521–530.

[27] J. Aranda, S. Easterbrook, and G. Wilson. "Requirements in the wild: How small companies do it". In *Proc. Int'l Requirements Engineering Conf. (RE)*. New Delhi, India: IEEE Computer Society, Oct. 2007, pp. 39–48.

[26] F. Steimann. "The Paradoxical Success of Aspect-oriented Programming". In *Proc. Int'l Conf. Object-Oriented Programming, Systems, Languages and Applications (OOPSLA)*. Portland, OR, USA: ACM, Oct. 2006, pp. 481–497.

[25] D. Garlan and B. Schmerl. "Architecture-driven Modelling and Analysis". In *Proc. Safety Critical Systems and Software (SCS)*. Melbourne, Australia: Australian Computer Society, Inc., Aug. 2006, pp. 3–17.

[24] D. Batory. "Feature Models, Grammars, and Propositional Formulas". In. Rennes, France: Springer-Verlag, Sept. 2005, pp. 7–20.

[23] P. Godefroid, N. Klarlund, and K. Sen. "DART: Directed Automated Random Testing". In *Proc. Conf. Programming Language Design and Implementation (PLDI)*. Chicago, IL, USA: ACM, June 2005, pp. 213–223.

[22] D. Batory, J. N. Sarvela, and A. Rauschmayer. "Scaling Step-wise Refinement". In *Proc. Int'l Conf. Software Engineering (ICSE)*. Portland, OR, USA: IEEE Computer Society, May 2003, pp. 187–197.

[21] M. Shaw. "Writing Good Software Engineering Research Papers: Minitutorial". In *Proc. Int'l Conf. Software Engineering (ICSE)*. Portland, Oregon: IEEE Computer Society, May 2003, pp. 726–736.

[20] T. A. Henzinger, R. Jhala, R. Majumdar, and G. Sutre. "Lazy Abstraction". In *Proc. Symp. Principles of Programming Languages (POPL)*. Portland, OR, USA: ACM, Jan. 2002, pp. 58–70.

[19] D. Fox. "KLD-Sampling: Adaptive Particle Filters". In *Advances in Neural Information Processing Systems 14*. MIT Press, 2001.

[18] A. Van Lamsweerde. "Goal-Oriented Requirements Engineering: A Guided Tour". In *Proc. Int'l Requirements Engineering Conf. (RE)*. Toronto, Canada: IEEE Computer Society, Aug. 2001, pp. 249–262.

[17] T. Ball and S. K. Rajamani. "Automatically Validating Temporal Safety Properties of Interfaces". In *Proc. Int'l Workshop Model Checking of Software (SPIN)*. Toronto, Canada: Springer-Verlag New York, Inc., May 2001, pp. 103–122.

[16] B. R. Murphy and M. S. Lam. "Program Analysis with Partial Transfer Functions". In *Proc. Workshop Partial Evaluation and Semantics-based Program Manipulation*. Boston, MA, USA: ACM, Jan. 2000, pp. 94–103.

[15] M. D. Ernst, J. Cockrell, W. G. Griswold, and D. Notkin. "Dynamically Discovering Likely Program Invariants to Support Program Evolution". In *Proc. Int'l Conf. Software Engineering (ICSE)*. Los Angeles, CA, USA: ACM, May 1999, pp. 213–224.

[14] P. Tarr, H. Ossher, W. Harrison, and S. M. Sutton Jr. "N Degrees of Separation: Multi-dimensional Separation of Concerns". In *Proc. Int'l Conf. Software Engineering (ICSE)*. Los Angeles, CA, USA: ACM, May 1999, pp. 107–119.

[13] L. Cardelli. "Program Fragments, Linking, and Modularization". In *Proc. Symp. Principles of Programming Languages (POPL)*. Paris, France: ACM, Jan. 1997, pp. 266–277.

[12] C. Potts and W. C. Newstetter. "Naturalistic Inquiry and Requirements Engineering: Reconciling Their Theoretical Foundations". In *Proc. Int'l Requirements Engineering Conf. (RE)*. Annapolis, MD, USA: IEEE Computer Society, Jan. 1997, pp. 118–127.

[11] M. Jackson. "The World and the Machine". In *Proc. Int'l Conf. Software Engineering (ICSE)*. Seattle, WA, USA: ACM, Apr. 1995, pp. 283–292.

[10] O. Gotel and A. Finkelstein. "Contribution Structures". In *Proc. Int'l Requirements Engineering Conf. (RE)*. York, U.K.: IEEE Computer Society, Mar. 1995, pp. 100–107.

[9] T. Biggerstaff. "The Library Scaling Problem and the Limits of Concrete Component Reuse". In *Proc. Int'l Conf. Software Reuse (ICSR)*. Rio de Janeiro, Brazil: IEEE Computer Society, 1994, pp. 102–109.

[8] R. Allen and D. Garlan. "Formalizing Architectural Connection". In *Proc. Int'l Conf. Software Engineering (ICSE)*. Sorrento, Italy: IEEE Computer Society Press, May 1994, pp. 71–80.

[7] L. Osterweil. "Software Processes Are Software Too". In *Proc. Int'l Conf. Software Engineering (ICSE)*. Monterey, CA, USA: IEEE Computer Society Press, Mar. 1987, pp. 2–13.

[6] W. W. Royce. "Managing the Development of Large Software Systems: Concepts and Techniques". In *Proc. Int'l Conf. Software Engineering (ICSE)*. Monterey, CA, USA: IEEE Computer Society Press, Mar. 1987, pp. 328–338.

[5] S. T. Redwine Jr. and W. E. Riddle. "Software Technology Maturation". In *Proc. Int'l Conf. Software Engineering (ICSE)*. London, England: IEEE Computer Society Press, Aug. 1985, pp. 189–200.

[4] D. L. Parnas, P. C. Clements, and D. M. Weiss. "The Modular Structure of Complex Systems". In *Proc. Int'l Conf. Software Engineering (ICSE)*. Orlando, FL, USA: IEEE Press, Mar. 1984, pp. 408–417.

[3] M. Weiser. "Program Slicing". In *Proc. Int'l Conf. Software Engineering (ICSE)*. San Diego, CA, USA: IEEE Press, Mar. 1981, pp. 439–449.

[2] D. L. Parnas. "Designing Software for Ease of Extension and Contraction". In *Proc. Int'l Conf. Software Engineering (ICSE)*. Atlanta, GA, USA: IEEE Press, May 1978, pp. 264–277.

[1] G. A. Kildall. “A Unified Approach to Global Program Optimization”. In *Proc. Symp. Principles of Programming Languages (POPL)*. Boston, MA, USA: ACM, Oct. 1973, pp. 194–206.

Technical Reports

[4] J. Newsome and D. Song. *Dynamic Taint Analysis for Automatic Detection, Analysis, and Signature Generation of Exploits on Commodity Software*. CMU-CS-04-140. Carnegie Mellon University, 2005.

[3] R. E. Filman and D. P. Friedman. *Aspect-Oriented Programming is Quantification and Obliviousness*. Technical Report. NASA, 2000.

[2] R. Kazman, M. Klein, and P. Clements. *ATAM: Method for Architecture Evaluation*. Technical Report CMU/SEI-2000-TR-004. Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2000.

[1] D. Garlan and M. Shaw. *An Introduction to Software Architecture*. Technical Report CMU-CS-94-166. Carnegie Mellon University, Jan. 1994.

Part of Books

[1] M. Shaw. "The Role of Design Spaces". In M. Petre and A. van der Hoek. *Software Designers in Action: A Human-Centric Look at Design Work*. CRC Press, 2013. Chap. 3.

Thesis

[2] N. Siegmund. “Measuring and Predicting Non-Functional Properties of Customizable Programs”. PhD thesis. University of Magdeburg, 2012.

[1] C. Kästner. “Virtual Separation of Concerns: Toward Preprocessors 2.0”. Logos Verlag Berlin, isbn 978-3-8325-2527-9. PhD thesis. Magdeburg, Germany: University of Magdeburg, May 2010.

Miscellaneous

[9] TurtleBot. *What is TurtleBot?*

[8] M. Anders and M. I. Schwartzbach. *Static Program Analysis*. 2017.

[7] ROS.org. *amcl*. Ed. by FlorianSteinhardt. Aug. 18, 2016.

[6] M. Velez and J. Sawin. *Improving the Efficiency of CHA through Parallelization*. Poster. Inquiry at St. Thomas. St. Paul, MN, USA, May 2016.

[5] M. Velez and J. Sawin. *Faster WAH Compression Querying through the Use of Metadata*. Poster. Consortium for Computing Sciences in Colleges Midwest Region. 1^{st} place Discovery Track. Evansville, IN, USA, Oct. 2015.

[4] M. Velez and A. Solar-Lezama. *Simpler Implementation of Sketches through Enhanced Expressiveness*. Poster. MIT Summer Research Poster Session. Cambridge, MA, USA, Aug. 2015.

[3] M. Velez. *Current and Future Relationships Between Robots and Humans*. Summa Cum Laude Paper. Apr. 2015.

[2] M. Velez, P. Gittins, and J. Sawin. *Extending SMILES to Encode Reaction Mechanisms*. Poster. Inquiry at St. Thomas. St. Paul, MN, USA, May 2014.

[1] M. I. Schwartzbach. *Lecture Notes on Static Analysis*. 2008.

Manual

[1] E. Bruneton. *ASM 4.0 A Java bytecode engineering library*. 2nd. ASM. Sept. 2011.