

# A Bibliography of Aspect-Oriented Software Development

## Version 2.0

Robert E. Filman

November 13, 2005

A bibliography of the literature related to Aspect-Oriented Software Development.

My thanks to Mario Südholt, Karl Lieberherr, Sacher Dominik, Guenter Kniesel, Bart de Win, Lee Carver, Curtis Clifton, Mira Mezini, Olaf Conradi, Minhuan Huang and Bruno De Fraine for their contributions.

This is meant to be an evolving document. Please send additions, corrections, comments and disagreements to the author at [filman@computer.org](mailto:filman@computer.org). There's more detail in the bibtex version of this file (<http://home.comcast.net/~refilman/text/oif/aosd-bibliography.bib>).

## References

- [1] Rickard A. Åberg, Julia L. Lawall, Mario Südholt, and Gilles Muller. Evolving an OS kernel using temporal logic and aspect-oriented programming. In Coady et al. [326].
- [2] Franz Achermann. Language support for feature mixing. In Tarr et al. [1363].
- [3] Franz Achermann. *Forms, Agents and Channels - Defining Composition Abstraction with Style*. PhD thesis, University of Berne, January 2002.
- [4] Franz Achermann, Markus Lumpe, Jean-Guy Schneider, and Oscar Nierstrasz. PICCOLA—A small composition language. In H. Bowman and J. Derrick, editors, *Formal methods for distributed processing: a survey of object-oriented approaches*, pages 403–426. Cambridge University Press, Cambridge, UK, 2001.
- [5] Franz Achermann and Oscar Nierstrasz. Explicit Namespaces. In Jürg Gutknecht and Wolfgang Weck, editors, *Modular Programming Languages*, volume 1897 of *LNCS*, pages 77–89. Springer-Verlag, September 2000.
- [6] Franz Achermann and Oscar Nierstrasz. Applications = Components + Scripts—A Tour of Piccola. In Mehmet Akşit, editor, *Software Architectures and Component Technology*, pages 261–292. Kluwer, 2001.
- [7] Bram Adams, Kris De Schutter, and Andy Zaidman. AOP for legacy environments, a case study. In Gybels et al. [615].
- [8] Bram Adams and Tom Tourwé. Aspect orientation for c: Express yourself. In Bergmans et al. [156].
- [9] W. Aitken, B. Dickens, P. Kwiatkowski, O. de Moor, D. Richter, and C. Simonyi. Transformation in intentional programming. In *5th Int'l Conf. Software Reuse*, pages 114–123. IEEE, 1998.
- [10] M. Akşit. The analysis and design of distributed systems. In *Addendum to the Proc. on Object-Oriented Programming Systems, Languages and Applications*, page 223, 1992.
- [11] M. Akşit. Composition and separation of concerns in the object-oriented model. *ACM Computing Surveys*, 28A(4), 1996.
- [12] M. Akşit, editor. *Software Architectures and Component Technology: The State of the Art in Research and Practice*. Kluwer Academic Publishers, 2001.
- [13] M. Akşit and L. Bergmans. Composing multiple-client–multiple-server synchronizations. In *Proc. IEEE Joint Workshop on Parallel and Distributed Systems*, pages 269–282, April 1997.

- [14] M. Akşit and L. Bergmans. Examples of reusing synchronization code in aspect-oriented programming using composition-filters. In *Proc. 5th. Maghrebian Conf. Software Engineering and Artificial Intelligence (MCSEAI'98)*, pages 257–272, December 1998.
- [15] M. Akşit and L. Bergmans. Guidelines for identifying obstacles when composing distributed systems from components. In Akşit [12], pages 29–56.
- [16] M. Akşit, L. Bergmans, and S. Vural. An object-oriented language-database integration model: The composition-filters approach. In O. Lehrmann Madsen, editor, *Proc. 7th European Conf. Object-Oriented Programming*, pages 372–395. Springer-Verlag Lecture Notes in Computer Science, 1992.
- [17] M. Akşit and J. Bosch. Issues in object-oriented real time language design. In W. Halang and A. Stoyenko, editors, *Real Time Computing*, pages 510–511. Springer Verlag Nato ASI Series, 1992.
- [18] M. Akşit, J. Bosch, W. v. d. Sterren, and L. Bergmans. Real-time specification inheritance anomalies and real-time filters. In Tokoro and Pareschi [1393], pages 386–407.
- [19] M. Akşit and B. Tekinerdoğan. Component composability issues in object-oriented programming. *XOOTIC Magazine*, 5(2):15–20, December 1997.
- [20] M. Akşit and B. Tekinerdoğan. Reusing and composing components: Problems and solutions. In *Proc. Smalltalk und Java in Industrie und Ausbildung 97 (STJA 97)*, September 1997.
- [21] M. Akşit and B. Tekinerdoğan. Aspect-oriented programming using composition filters. In S. Demeyer and J. Bosch, editors, *Object-Oriented Technology, ECOOP'98 Workshop Reader*, page 435. Springer Verlag, July 1998.
- [22] M. Akşit and B. Tekinerdoğan. Solving the modeling problems of object-oriented languages by composing multiple aspects using composition filters. In Lopes et al. [914].
- [23] M. Akşit, B. Tekinerdoğan, and L. Bergmans. Achieving adaptability through separation and composition of concerns. In M. Muhlhauser, editor, *Special Issues in Object-Oriented Programming*, pages 12–23. dpunkt verlag, 1996.
- [24] M. Akşit, K. Wakita, J. Bosch, L. Bergmans, and A. Yonezawa. Abstracting object-interactions using composition-filters. In R. Guerraoui, O. Nierstrasz, and M. Riveill, editors, *Object-Based Distributed Processing*, pages 152–184. Springer-Verlag Lecture Notes in Computer Science, 1993.
- [25] Mehmet Akşit. Issues in aspect-oriented programming. In Lopes et al. [918].
- [26] Mehmet Akşit, editor. *Proc. 2nd Int' Conf. on Aspect-Oriented Software Development (AOSD-2003)*. ACM Press, March 2003.
- [27] Mehmet Akşit, Jean Bézivin, and Ella Roubtsova, editors. *1st International Workshop on Aspect-Based and Model-Based Separation of Concerns in Software Systems (ECMDA05)*, November 2005.
- [28] Mehmet Akşit and Zîéd Choukair, editors. *Proc. 2nd Int'l Workshop on Aspect Oriented Programming for Distributed Computing Systems (ICDCS-2002), Vol. 2*, July 2002.
- [29] Mehmet Akşit, Zîéd Choukair, and Tzilla Elrad. Experiences with auto-adaptive and reconfigurable systems. *Software, Practice and Experience*, xxx.
- [30] Mehmet Akşit, Jan Willem Dijkstra, and Anand Tripathi. Atomic delegation: Object-oriented transactions. *IEEE Software*, 8(2):84–92, 1991.
- [31] Mehmet Akşit and Mira Mezini, editors. *Net.Object Days 2002*, October 2002.
- [32] Mehmet Akşit, Bedir Tekinerdoğan, and Lodewijk Bergmans. The six concerns for separation of concerns. In Bergmans et al. [154].
- [33] Mehmet Akşit and Anand Tripathi. Data abstraction mechanisms in SINA/ST. In *3rd Conf. Object-oriented Programming, Systems, Languages, and Applications (OOP-SLA)*, pages 267–275. ACM, 1988.

- [34] Faisal Akkawi, Atef Bader, and Tzilla Elrad. Dynamic weaving for building reconfigurable software systems. In De Volder et al. [407].
- [35] Mehmet Aksit, Jean Bézivin, and Ella Roubtsova, editors. *Aspect-Based and Model-Based Separation of Concerns in Software Systems*, November 2005.
- [36] Omar Aldawud, Atef Bader, and Tzilla Elrad. Weaving with statecharts. In Aldawud et al. [37].
- [37] Omar Aldawud, Grady Booch, Siobhán Clarke, Tzilla Elrad, Bill Harrison, Mohamed Kandí, and Alfred Strohmeier, editors. *Workshop on Aspect-Oriented Modeling with UML (AOSD-2002)*, March 2002.
- [38] Omar Aldawud, Grady Booch, Jeff Gray, Jörg Kienzle, Dominik Stein, Mohamed Kandé, Faisal Akkawi, and Tzilla Elrad, editors. *The 5th Aspect-Oriented Modeling Workshop In Conjunction with UML 2004*, October 2004.
- [39] Omar Aldawud, Tzilla Elrad, and Atef Bader. Aspect-oriented modeling to automate the implementation and validation of concurrent software systems. In *Workshop on Specification, Implementation and Validation of Object-oriented Embedded Systems (SIVOES'2001) (ECOOP)*, 2001.
- [40] Omar Aldawud, Tzilla Elrad, and Atef Bader. A UML profile for aspect oriented modeling. In De Volder et al. [407].
- [41] Omar Aldawud, Tzilla Elrad, and Atef Bader. UML profile for aspect-oriented software development. In Aldawud et al. [43].
- [42] Omar Aldawud, Tzilla Elrad, Jeff Gray, Mohamed Kandé, Jörg Kienzle, and Dominik Stein, editors. *7th International Workshop on Aspect-Oriented Modeling*, October 2005.
- [43] Omar Aldawud, Mohamed Kandé, Grady Booch, Bill Harrison, and Dominik Stein, editors. *Third International Workshop on Aspect Oriented Modeling*, March 2003.
- [44] Omar Aldawud, Mohamed Kandé, Grady Booch, Bill Harrison, Dominik Stein, Jeff Gray, Siobhán Clarke, Aida Zakaria Santeon, Peri Tarr, and Faisal Akkawi, editors. *The 4th AOSD Modeling With UML Workshop*, October 2003.
- [45] Jonathan Aldrich. Challenge problems for separation of concerns. In Tarr et al. [1362].
- [46] Jonathan Aldrich. Open modules: A proposal for modular reasoning in aspect-oriented programming. In Clifton et al. [321], pages 7–18.
- [47] Jonathan Aldrich. Open modules: Reconciling extensibility and information hiding. In Bergmans et al. [155].
- [48] Roger T. Alexander and Anneliese A. Andrews, editors. *Workshop on Testing Aspect-Oriented Programs*, March 2005.
- [49] Roger T. Alexander and James M. Bieman. Challenges of aspect-oriented technology. In *Workshop on Software Quality, 24th Int'l Conf. Software Engineering*, May 2002.
- [50] Klaus Alfert. Requirements, features and aspects for software product lines. In Araújo et al. [79].
- [51] Chris Allan, Pavel Avgustinov, Aske Simon Christensen, Laurie Hendren, Sascha Kuzins, Ondřej Lhoták, Oege de Moor, Damien Sereni, Ganesh Sittampalam, and Julian Tibble. Adding trace matching with free variables to AspectJ. In *OOPSLA '05: Proceedings of the 20th annual ACM SIGPLAN conference on Object oriented programming systems languages and applications*, pages 345–364. ACM Press, 2005.
- [52] Chris Allan, Pavel Avgustinov, Aske Simon Christensen, Laurie J. Hendren, Sascha Kuzins, Jennifer Lhoták, Ondrej Lhoták, Oege de Moor, Damien Sereni, Ganesh Sittampalam, and Julian Tibble. *abc*: The AspectBench compiler for AspectJ. In Glück and Lowry [565], pages 10–16.
- [53] Sufyan Almajali and Tzilla Elrad. A dynamic aspect oriented C++ using MOP with minimal hook weaving approach. In Filman et al. [502], pages 1–8.
- [54] Sufyan Almajali and Tzilla Elrad. Coupling availability and efficiency for aspect oriented runtime weaving systems. In Filman et al. [515], pages 47–55.
- [55] Karine Altisen, Florence Maraninchi, and David Stauch. Exploring aspects in the context of reactive systems. In Clifton et al. [321], pages 45–51.

- [56] Rubén Altman, Alan Cyment, and Nicolás Kicillof. On the need for setpoints. In Gybels et al. [615].
- [57] Jordi Alvarez, Ignacio Gutierrez, and Miguel-Angel Sicilia. Personalization as a cross-cutting concern in web servers: A case study on Java servlet technology. In Lorenz and Coady [932].
- [58] Alexandre Alvaro, Daniel Lucrédio, Vinicius Cardoso Garcia, and Eduardo Santana de Almeida. Aspect IPM: Towards an incremental process model based on AOP for component-based development. In Garcia et al. [549].
- [59] Vander Alves and Paulo Borba. A design pattern for distributed applications. In Adriano Souza et al., editors, *XIV Brazilian Symposium on Software Engineering-Minicourses and Tutorials*, pages 191–219, October 2000.
- [60] Vander Alves and Paulo Borba. Distributed adapters pattern: A design pattern for object-oriented distributed applications. In *First Latin American Conference on Pattern Languages Programming, SugarLoaf-PLoP 2001*, October 2001. Published in UERJ Magazine: Special Issue on Software Patterns, June 2002, pages 132-142.
- [61] Vander Alves and Paulo Borba. An implementation method for distributed object-oriented applications. In *Proceedings of XV Brazilian Symposium on Software Engineering, SBES 2001*, pages 161–176, October 2001.
- [62] Vander Alves, Ayla Dantas, and Paulo Borba. AOP-driven variability in product lines of pervasive computing applications. In Matthias Riebisch, Jim Coplien, and Detlef Streitferdt, editors, *Modeling Variability for Object-Oriented Product Lines*, July 2003.
- [63] Vander Alves, Pedro Matos, Jr., and Paulo Borba. An incremental aspect-oriented product line method for J2ME game development. In Beuche et al. [164].
- [64] Pablo Amaya, Carlos González, and Juan Murillo. MDA and separation of aspects — an approach based on multiple views and subject oriented design. In Kandé et al. [737].
- [65] Pablo Amaya, Carlos Gonzalez, and Juan M. Murillo. Towards a subject-oriented model-driven framework. In Aksit et al. [35].
- [66] Davide Ancona, Giovanni Lagorio, and Elena Zucca. Jam—A smooth extension of Java with mixins. In Bertino [163], pages 154–178.
- [67] Ken Anderson. An example of using collaborator and adapters to reuse a synchronization pattern. In Tarr et al. [1362].
- [68] Carlos Andrade, Andre L. M. Santos, and Paulo H. M. Borba. AspectH: Uma extensão orientada a aspectos de Haskell. In Garcia et al. [549].
- [69] L. Andrade and J. L. Fiadeiro. An architectural approach to auto-adaptive systems. In Aksit and Choukair [28].
- [70] Luis Andrade, José Luiz Fiadeiro, João Gouveia, and Georgios Koutsoukos. Separating computation, coordination and configuration. *Journal of Software Maintenance and Evolution: Research and Practice*, 14(5):353–369, 2002.
- [71] James Andrews. Using process algebra as a foundation for programming by separation of concerns. In Tarr and Ossher [1366].
- [72] James H. Andrews. Process-algebraic foundations of aspect-oriented programming. In Yonezawa and Matsuoka [1494], pages 187–209.
- [73] James H. Andrews. Using process algebra as a foundation for programming by separation of concerns. In Bergmans et al. [154].
- [74] M. Antunes, H. Miranda, A. Rito Silva, L. Rodrigues, and J. Martins. Separating replication from distributed communication: Problems and solutions. In Choukair [280], pages 103–110.
- [75] Tomoyuki Aotani and Hidehiko Masuhara. Compiling conditional pointcuts for user-level semantic pointcuts. In Bergmans et al. [156].
- [76] Sven Apel, Thomas Leich, Marko Rosenmüller, and Gunter Saake. FeatureC++: On the symbiosis of feature-oriented and aspect-oriented programming. In Glück and Lowry [565], pages 125–140.

- [77] Sven Apel, Thomas Leich, and Gunter Saake. Aspect refinement in software product lines. In Araújo et al. [79].
- [78] J. Araújo and P. Coutinho. Identifying aspectual use cases using a viewpoint-oriented requirements method. In Araújo et al. [81].
- [79] João Araújo, Elisa Baniassad, Paul Clements, Krzysztof Czarnecki, Jeff Gray, Charles Krueger, Neil Loughran, Ana Moreira, Bedir Tekinerdogan, and Trevor Young, editors. *Aspects and Software Product Lines: An Early Aspects Workshop at SPLC-Europe 2005*, September 2005.
- [80] João Araújo, Ana Moreira, Isabel Brito, and Awais Rashid. Aspect-oriented requirements with UML. In Kandé et al. [735].
- [81] João Araújo, Awais Rashid, Bedir Tekinerdogan, Ana Moreira, and Paul Clements, editors. *Early Aspects 2003: Aspect-Oriented Requirements Engineering and Architecture Design*, March 2003.
- [82] João Araújo, Jon Whittle, and Dae-Kyoo Kim. Modeling and composing scenario-based requirements with aspects. In *Proc. 12th Int'l Requirements Engineering Conference*, pages 53–62. IEEE, September 2004.
- [83] F. A. Ariniégas. Introduction to Perceval: Aspect-oriented design using XML Schema and Groves. In *5th Int'l Conf. Parallel and Distributed Processing Techniques and Applications*, pages 26–29. CSREA Press, June 2000.
- [84] Ali Arsanjani, Brent Hailpern, Joanne Martin, and Peri Tarr. Web services: Promises and compromises. *Queue*, 1(1):48–58, 2003.
- [85] Aspect-Oriented Software Association. Aspect-oriented software development-tools and languages. <http://aosd.net/tools.htm>.
- [86] AspectC web page. <http://www.cs.ubc.ca/labs/spl/projects/aspectc.html>.
- [87] AspectJ project. <http://www.eclipse.org/aspectj/>.
- [88] AspectWerkz homepage. <http://aspectwerkz.codehaus.org/>.
- [89] Uwe Aßmann. A component model for invasive composition. In Lopes et al. [913].
- [90] Uwe Aßmann and Andreas Ludwig. Aspect weaving by graph rewriting. In U. W. Eisencker and K. Czarnecki, editors, *Generative Component-based Software Engineering (GCSE)*, October 1999.
- [91] Uwe Aßmann, Klaas van den Berg, Lynne Blair, Ramon Clout, Clemens Kerer, Ivan Kurtev, Elke Pulvermüller, Awais Rashid, and Andreas Speck, editors. *Modular Representation and Interpretation of Concerns in XML (AOSD-2002)*, March 2002.
- [92] Colin Atkinson and Thomas Kühne. Separation of concerns through stratified architectures. In Lopes et al. [913].
- [93] S. Ausmann and M. Haupt. Axon—Dynamic AOP through runtime inspection and monitoring. In Filman et al. [501].
- [94] Enis Avdičaušević, Mitja Lenič, Marjan Mernik, and Viljem Žumer. AspectCOOL: An experiment in design and implementation of aspect-oriented language. *SIGPLAN Not.*, 36(12):84–94, 2001.
- [95] Enis Avdičaušević, Marjan Mernik, Mitja Lenič, and Viljem Žumer. Experimental aspect-oriented language—AspectCOOL. In *Proceedings of the 2002 ACM Symposium on Applied Computing*, pages 943–947. ACM Press, 2002.
- [96] Pavel Avgustinov, Aske Simon Christensen, Laurie Hendren, Sascha Kuzins, Jennifer Lhotak, Ondrej Lhotak, Oege de Moor, Damien Sereni, Ganesh Sittampalam, and Julian Tibble. abc: An extensible AspectJ compiler. In Tarr [1361], pages 87–98.
- [97] Boris Bachmendo, Stefan Hanenberg, Stephan Herrmann, and Günter Kniesel, editors. *3rd Workshop on Aspect-Oriented Software Development (AOSD-GI) of the SIG Object-Oriented Software Development, German Informatics Society*, March 2003.
- [98] Boris Bachmendo and Rainer Unland. Aspect-based workflow evolution. In Rashid [1181].
- [99] Atef Bader, Husein Armouti, Shangping Ren, and Tzilla Elrad. Components weaving for software product features. In Beuche et al. [164].

- [100] Atef Bader and Tzilla Elrad. The adaptive arena: Language constructs and architectural abstractions for concurrent object-oriented systems. In *ICPADS '98*, 1998.
- [101] Atef Bader and Tzilla Elrad. Framework and design pattern for concurrent passive objects. In *Proc. IASTED/SE '98*, 1998.
- [102] Jason Baker and Wilson Hsieh. Runtime aspect weaving through metaprogramming. In Kiczales [781], pages 86–98.
- [103] J. Bakker, B. Tekinerdoğan, and M. Akşit. Characterization of early aspects approaches. In Clements et al. [320].
- [104] Shubhanan Bakre and Tzilla Elrad. Linguistic issues for developing aspect-rich high-performance adaptable systems. In Bergmans et al. [153].
- [105] David Balzarotti and Mattia Monga. Using program slicing to analyze aspect oriented composition. In Clifton et al. [321], pages 25–30.
- [106] Davide Balzarotti, Antonio Castaldo D’Ursi, Luca Cavallaro, and Mattia Monga. Slicing AspectJ woven code. In Leavens et al. [867].
- [107] E. Baniassad and S. Clarke. Finding aspects in requirements with Theme/Doc. In Tekinerdoğan et al. [1378], pages 15–22.
- [108] E. Baniassad and S. Clarke. Investigating the use of clues for scaling document-level concern graphs. In Moreira et al. [1017].
- [109] Elisa Baniassad, João Araújo, Paul Clements, and Paulo Merson, editors. *Early Aspects: OOPSLA 2005*, October 2005.
- [110] Elisa Baniassad, Gail Murphy, Christa Schwanninger, and Michael Kircher. Managing crosscutting concerns during software evolution tasks: An inquisitive study. In Kiczales [781], pages 120–126.
- [111] Elisa L. A. Baniassad, Gail C. Murphy, and Christa Schwanninger. Determining the “why” of concerns. In Bergmans et al. [154].
- [112] Elisa L. A. Baniassad, Gail C. Murphy, and Christa Schwanninger. Determining the “why” of concerns. In Tarr and Ossher [1366].
- [113] Elisa L. A. Baniassad, Gail C. Murphy, Christa Schwanninger, and Michael Kircher. Where are programmers faced with concerns? In Tarr et al. [1362].
- [114] D. Bardou. Roles, subjects and aspects: How do they relate? In Lopes et al. [914].
- [115] Daniel Bardou. Experimenting split-object in Java and AspectJ. In Bachmendo et al. [97].
- [116] Eduardo Barra, Gonzalo Génova, and Juan Llorens. An approach to aspect modeling with UML 2.0. In Aldawud et al. [38].
- [117] Luciano Porto Barreto, Rémi Douence, Gilles Muller, and Mario Südholt. Programming OS schedulers with domain-specific languages and aspects: New approaches for OS kernel engineering. In Coady et al. [327].
- [118] S. Barrett and P. Foster. Turning Java components into CORBA components with replication. In *OMG-DARPA-MCC Workshop on Compositional Software Architectures*, January 1998.
- [119] João Paulo Barros and Luís Gomes. Activities as behaviour aspects. In Kandé et al. [735].
- [120] João Paulo Barros and Luís Gomes. Towards the support for crosscutting concerns in activity diagrams: A graphical approach. In Aldawud et al. [44].
- [121] Ohad Barzilay, Yishai A. Feldman, Shmuel Tyszberowicz, and Amiram Yehudai. Call and execution semantics in AspectJ. In Clifton et al. [321], pages 19–24.
- [122] Mark Basch and Arturo Sanchez. Incorporating aspects into the UML. In Aldawud et al. [43].
- [123] L. Bass, M. Klein, and L. Northrop. Identifying aspects using architectural reasoning. In Tekinerdoğan et al. [1378], pages 50–56.
- [124] Paul G. Bassett. *Framing Software Reuse: Lessons From the Real World*. Prentice Hall PTR, Englewood Cliffs, New Jersey, 1996.
- [125] Adam Batenin and Eamonn O’Neill. Towards unanticipated composition of concerns in Hyperspaces. In Glandrup et al. [564].

- [126] Don Batory. Refinements and separation of concerns. In Tarr et al. [1363].
- [127] Don Batory, Clay Johnson, Bob MacDonald, and Dale von Heeder. Achieving extensibility through product-lines and domain-specific languages: A case study. *ACM Transactions on Software Engineering and Methodology (TOSEM)*, 11(2):191–214, 2002.
- [128] Don Batory, Jia Liu, and Jacob Neal Sarvela. Refinements and multi-dimensional separation of concerns. In *ESEC/FSE-11: Proceedings of the 9th European software engineering conference held jointly with 11th ACM SIGSOFT international symposium on Foundations of software engineering*, pages 48–57, New York, 2003. ACM Press.
- [129] Don Batory, Robert E. Lopez-Herrejon, and Jean-Philippe Martin. Generating product-lines of product-families. In *Proc. 17th IEEE Int'l Conf. on Automated Software Engineering*, pages 81–92, September 2002.
- [130] Don Batory, Vivek Singhal, Jeff Thomas, Sankar Dasari, Bart Geraci, and Marty Sirkin. The GenVoca model of software-system generators. *IEEE Software*, 11(5):89–94, 1994.
- [131] Benoit Baudry, Franck Fleurey, Robert France, and Raghu Reddy. Exploring the relationship between model composition and model transformation. In Aldawud et al. [42].
- [132] Joachim Bayer. Towards engineering product lines using concerns. In Tarr et al. [1363].
- [133] BCEL homepage.  
<http://jakarta.apache.org/bcel/>.
- [134] Christian Becker. Quality of service and O.O. oriented middleware multiple concerns and their separation. In Choukair [280], pages 117–126.
- [135] Christian Becker and Kurt Geihs. Quality of service—Aspects of distributed programs. In Lopes et al. [926].
- [136] U. Becker. D2AL: A design-based aspect language for distribution control. In Lopes et al. [914].
- [137] Georg Beier and Markus Kern. Aspects in UML models from a code generation perspective. In Kandé et al. [735].
- [138] N. Bencomo, G. Blair, and P. Sawyer. Tracing the distribution concern: Bridging the gap. In Moreira et al. [1017].
- [139] L. Berger, A. M. Dery, and M. Fornarino. Interactions between objects: An aspect of object-oriented languages. In Lopes et al. [914].
- [140] L. Berger, A. M. Dery, and M. Fornarino. Interactions between objects: An aspect of object-oriented languages. In Lopes et al. [926].
- [141] L. Bergmans. *Composing Concurrent Objects*. PhD thesis, University of Twente, 1994.
- [142] L. Bergmans. The composition filters object model. Technical report, Dept. of Computer Science, University of Twente, 1994.
- [143] L. Bergmans and M. Akşit. Reusability problems in object-oriented concurrent programs. In *Proc. ECOOP'92 Workshop Object-Based Concurrency and Reuse*, June 1992.
- [144] L. Bergmans and M. Akşit. Composing synchronisation and real-time constraints. *Journal of Parallel and Distributed Computing*, 36:32–52, 1996.
- [145] L. Bergmans and M. Akşit. Aspects and crosscutting in layered middleware systems. In *RM2000 Workshop in Reflective Middleware*, April 2000.
- [146] L. Bergmans and M. Akşit. Composing crosscutting concerns using composition filters. *Comm. ACM*, 44(10):51–57, October 2001.
- [147] L. Bergmans and M. Akşit. Composing multiple concerns using composition filters. Technical report, University of Twente, The Netherlands, 2001.
- [148] L. Bergmans, M. Akşit, and J. Bosch. Composition filters: Extended expressiveness for OOPLs. In *OOPSLA '92 Workshop Object-Oriented Programming Languages: The Next Generation*, 1992.

- [149] L. Bergmans, M. Akşit, and B. Tekinerdoğan. Aspect composition using composition filters. In Akşit [12], pages 357–382.
- [150] L. Bergmans and P. Cointe. Workshop report of the ECOOP’96 workshop on composability issues in object-orientation. In M. Mühlhäuser, editor, *Special Issues in Object-Oriented Programming*, pages 53–124. dpunkt verlag, Heidelberg, 1997.
- [151] Lodewijk Bergmans and Mehmet Akşit. Composing software from multiple concerns: A model and composition anomalies. In Tarr et al. [1363].
- [152] Lodewijk Bergmans and Mehmet Akşit. Principles and design rationale of composition filters. In Filman et al. [513], pages 63–95.
- [153] Lodewijk Bergmans, Johan Brichau, Peri Tarr, and Erik Ernst, editors. *SPLAT: Software engineering Properties of Languages for Aspect Technologies*, March 2003.
- [154] Lodewijk Bergmans, Maurice Glandrup, Johan Brichau, and Siobhán Clarke, editors. *Workshop on Advanced Separation of Concerns (ECOOP 2001)*, June 2001.
- [155] Lodewijk Bergmans, Kris Gybels, Peri Tarr, and Erik Ernst, editors. *SPLAT: Software engineering Properties of Languages for Aspect*, March 2004.
- [156] Lodewijk Bergmans, Kris Gybels, Peri Tarr, and Erik Ernst, editors. *Software Engineering Properties of Languages and Aspect Technologies*, March 2005.
- [157] Lodewijk Bergmans, Bedir Tekinerdoğan, Maurice Glandrup, and Mehmet Akşit. On composing separated concerns: Composability and composition anomalies. In Tarr et al. [1362].
- [158] Lodewijk M. J. Bergmans. Towards detection of semantic conflicts between crosscutting concerns. In Hannemann et al. [640].
- [159] Lodewijk M. J. Bergmans and Mehmet Akşit. Analyzing multi-dimensional programming in AOP and composition filters. In Ossher et al. [1091].
- [160] Lodewijk M. J. Bergmans and Mehmet Akşit. How to deal with encapsulation in aspect-orientation. In De Volder et al. [407].
- [161] Paul Bergstein. *Managing the Evolution of Object-Oriented Systems*. PhD thesis, Northeastern University, 1994.
- [162] Silvia Bertagnolli and Maria Lúcia Blanck Lisbôa. Checklists e léxico como artefatos para identificação de aspectos. In Garcia et al. [549].
- [163] E. Bertino, editor. *ECOOP 2000—Object-Oriented Programming: 14th European Conference, LNCS 1850*. Springer Verlag, June 2000.
- [164] Danilo Beuche, Krzysztof Czarnecki, Mira Mezini, Christa Schwanninger, and Markus Voelter, editors. *Managing Variabilities Consistently in Design and Code (MVCDC OOPSLA04)*, October 2004.
- [165] Antoine Beugnard. How to make aspect reusable, a proposition. In Lopes et al. [924].
- [166] Jean Bezivin. Aspect-oriented modeling: Oxymoron or pleonasm? In Akşit and Choukair [28].
- [167] Dave Binkley, Mariano Ceccato, Mark Harman, and Paolo Tonella. Automated point-cut extraction. In Tourwé et al. [1399].
- [168] Andrew P. Black and Mark P. Jones. Perspectives on software. In Tarr et al. [1362].
- [169] Andrew P. Black and Jonathan Walpole. Aspects of information flow. In Lopes et al. [913].
- [170] Gordon S. Blair, Lynne Blair, Awais Rashid, Ana Moreira, João Araújo, and Ruzanna Chitchyan. Engineering aspect-oriented systems. In Filman et al. [513], pages 379–406.
- [171] L. Blair and G. S. Blair. The impact of aspect-oriented programming on formal methods. In Lopes et al. [914].
- [172] L. Blair and G. S. Blair. The impact of aspect-oriented programming on formal methods. Technical Report MPG-98-08, Lancaster University, 1998.
- [173] L. Blair, G. S. Blair, and A. Andersen. Separating function behaviour and performance constraints: Aspect-oriented specification. Technical Report MPG-98-07, Lancaster University, 1998.

- [174] L. Blair, G. S. Blair, R. Andersen, and T. Jones. Formal support for dynamic QoS management in the development of open component-based distributed systems. *IEE Proceedings-Software*, 148(3):89–97, June 2001.
- [175] Lynne Blair and Gordon Blair. A tool suite to support aspect-oriented specification. In Lopes et al. [924].
- [176] Lynne Blair and Mattia Monga. Reasoning on AspectJ programmes. In Bachmendo et al. [97].
- [177] G. Blank and G. Vayngrib. Aspects of Enterprise Java Beans. In Lopes et al. [914].
- [178] M. Blay-Fornarino, A. M. Pinna-Dery, and M. Riveill. Configuring distributed applications. In Akşit and Choukair [28].
- [179] Christoph Bockisch, Michael Haupt, Mira Mezini, and Klaus Ostermann. Virtual machine support for dynamic join points. In Lieberherr [884], pages 83–92.
- [180] Christoph Bockisch, Mira Mezini, and Klaus Ostermann. Quantifying over dynamic properties of program execution. In Filman et al. [515], pages 71–75.
- [181] Eric Bodden. Concern specific languages and their implementation with abc. In Bergmans et al. [156].
- [182] Ron Bodkin. Commercialization of AOSD: The road ahead. In Bodkin et al. [185].
- [183] Ron Bodkin. Enterprise security aspects. In De Win et al. [413].
- [184] Ron Bodkin. Aop@work: Performance monitoring with aspectj, part 1. Technical report, IBM Developer Works, September 2005.
- [185] Ron Bodkin, Adrian Colyer, Juri Memmert, and Arno Schmidmeier, editors. *AOSD Workshop on Commercialization of AOSD Technology*, March 2003.
- [186] C. Bogdan. Facets of concerns. In Tekinerdoğan et al. [1378], pages 57–61.
- [187] K. Böllert. Aspect-oriented programming case study: System management application. In Lopes et al. [914].
- [188] Kai Böllert. On weaving aspects. In Lopes et al. [924].
- [189] Jonas Bonér. AspectWerkz 2: An extensible aspect container. *TheServerSide.COM*, November 2004.
- [190] Jonas Bonér. What are the key issues for commercial AOP use: how does AspectWerkz address them? In Lieberherr [884], pages 5–6.
- [191] Jonas Bonér, Alexandre Vasseur, and Joakim Dahlstedt. JRocket JVM support for AOP, part 1. Technical report, BEA dev2dev, August 2005.
- [192] Jonas Bonér, Alexandre Vasseur, and Joakim Dahlstedt. JRocket JVM support for AOP, part 2. Technical report, BEA dev2dev, August 2005.
- [193] J. Bosch. Composition through superimposition. In M. Mühlhäuser, editor, *Workshop report of the ECOOP'96 workshop on Composability Issues in Object-Oriented, Special Issues in Object-Oriented Programming*, pages 94–101. dpunkt, Heidelberg, 1997.
- [194] J. Bosch and M. Akşit. Composition-filters based real-time programming. In *OOP-SLA'92 Workshop on Evaluation of Object-Oriented Technology in Real-Time Systems: Past, Present & Future*, 1992.
- [195] Jan Bosch, editor. *Generative and Component-Based Software Engineering, Third International Conference (GCSE 2001), LNCS 2186*. Springer-Verlag Lecture Notes in Computer Science, September 2001.
- [196] Gustav Boström. A case study on estimating the software engineering properties of implementing database encryption as an aspect. In Bergmans et al. [155].
- [197] Gustav Bostrom. Database encryption as an aspect. In De Win et al. [413].
- [198] Gustav Boström, Martin Henkel, and Jaana Wäyrynen. Aspects in the agile toolbox. In Bergmans et al. [156].
- [199] Philippe Bouaziz and Lionel Seinturier. From software parameterization to software profiling. In Rashid [1181].

- [200] Nelis Boucké and Tom Holvoet. State-based join-points: Motivation and requirements. In Filman et al. [515], pages 1–4.
- [201] L. Bougé and N. Francez. A compositional approach to superimposition. In *15th Symp. Principles of Programming Languages (POPL)*, pages 240–249. ACM, 1988.
- [202] Noury Bouraqadi. Concern oriented programming using reflection. In Tarr et al. [1362].
- [203] Noury Bouraqadi and Thomas Ledoux. Supporting AOP using reflection. In Filman et al. [513], pages 261–282.
- [204] Noury M. N. Bouraqadi-Saâdani and Thomas Ledoux. How to weave? In Bergmans et al. [154].
- [205] Laurent Boussard. Towards a pragmatic composition model of CORBA services based on AspectJ. In Lopes et al. [913].
- [206] F. Boyer, S. Bouchenak, N. de Palma, and D. Hagimont. Aspects can be efficient: Experiences with replication and protection. Technical Report 4651, INRIA, 2001.
- [207] Gilad Bracha and William Cook. Mixin-based inheritance. In *Conf. Object-Oriented Programming: Systems, Languages, and Applications; European Conf. Object-Oriented Programming*, pages 303–311. ACM, 1990.
- [208] Alexandre M. Braga, Ricardo Darab, and Cecília M. F. Rubira. A meta-object protocol for secure composition of security mechanisms. In Tarr et al. [1362].
- [209] Stuart Bray, Marco Yuen, Yvonne Coady, and Marc E. Fiuczynski. Managing variability in systems: Oh what a tangled OS we weave. In Beuche et al. [164].
- [210] Silvia Breu. Towards hybrid aspect mining: Static extensions to dynamic aspect mining. In Tourwé et al. [1398].
- [211] Silvia Breu and Jens Krinke. Aspect mining using event traces. In *19th International Conference on Automated Software Engineering*, pages 310–315, Los Alamitos, California, September 2004. IEEE Computer Society.
- [212] Johan Brichau. Declarative composable aspects. In Tarr et al. [1362].
- [213] Johan Brichau, Wolfgang De Meuter, and Kris De Volder. Jumping aspects. In Lopes et al. [913].
- [214] Johan Brichau and Michael Haupt. Survey of aspect-oriented languages and execution models. Technical Report AOSD-Europe-VUB-01, AOSD-Europe, May 2005.
- [215] Johan Brichau, Kim Mens, and Kris De Volder. Building composable aspect-specific languages with logic metaprogramming. In *1st Conf. Generative Programming and Component Engineering*, volume 2487 of *lncs*, pages 110–127, Berlin, 2002. Springer-Verlag.
- [216] I. Brito and A. Moreira. Towards a composition process for aspect-oriented requirements. In Araújo et al. [81].
- [217] I. Brito and A. Moreira. Integrating the NFR framework in a RE model. In Tekinerdoğan et al. [1378], pages 27–32.
- [218] I. Brito, A. Moreira, and J. Araújo. A requirements model for quality attributes. In Rashid et al. [1191].
- [219] Alex Brodsky, Dima Brodsky, Ida Chan, Yvonne Coady, Stephan Gudmundson, Jody Pomkoski, and Joon Suan Ong. Coping with evolution: Aspects vs. aspirin. In De Volder et al. [407].
- [220] David Bruce and Nick Exon. Alternatives to aspect-oriented programming? In Rashid [1181].
- [221] Jean-Michel Bruel, João Araújo, Ana Moreira, and Albert Royer. Using aspects to develop built-in tests for components. In Aldawud et al. [44].
- [222] Eric Bruneton and Michel Riveill. Experiments with JavaPod, a platform designed for the adaptation of non-functional properties. In Yonezawa and Matsuoka [1494], pages 52–72.
- [223] Magiel Bruntink. Aspect mining using clone class metrics. In Tourwé et al. [1398].
- [224] Magiel Bruntink, Arie van Deursen, and Tom Tourwé. Isolating crosscutting concerns in system software. In Tourwé et al. [1398].

- [225] Avi Bryant, Andrew Catton, Kris De Volder, and Gail Murphy. Explicit programming. In Kiczales [781], pages 10–18.
- [226] Felix Bubl. Never mind the source code, but be aware of the context when dealing with cross cutting requirements. In Baniassad et al. [109].
- [227] Martin Büchi and Wolfgang Weck. Generic wrappers. In Bertino [163], pages 201–225.
- [228] R. J. A. Buhr. A possible design notation for aspect-oriented programming. In Lopes et al. [914].
- [229] A. Burggraaf. Solving modelling problems of CORBA using Composition Filters. Master’s thesis, Department of Computer Science, University of Twente, 1997.
- [230] Rainer Burgstaller, Thomas Fritz, and Egon Wuchner. Hot-deployment of aspects. In Filman et al. [502], pages 9–22.
- [231] G. Cabri, L. Leonardi, and F. Zambonelli. Separation of concerns in agent applications by roles. In Akşit and Choukair [28].
- [232] João Cachopo, António Menezes Leitão, and António Rito-Silva. The tyranny of the file decomposition. In Chu-Carroll et al. [286].
- [233] CAESARJ homepage. <http://caesarj.org>.
- [234] V. Cahill. An overview of the Coyote project. In Lopes et al. [914].
- [235] A. Cain, J. Schneider, D. Grant, and T. Chen. Runtime data analysis for Java programs. In Filman et al. [501].
- [236] Valter Camargo and Paulo Cesar Masiero. Implementação de variabilidades em frameworks orientados a aspecto. In Garcia et al. [549].
- [237] Juan Antonio Campos and Mónica Pinto. Desarrollo de aplicaciones colaborativas con CoopTEL: El juego del Pictionary. In Fuentes et al. [538].
- [238] Carlos Canal, Juan Manuel Murillo, and Pascal Poizat, editors. *First International Workshop on Coordination and Adaptation Techniques for Software Entities (WCAT’04) (ECOOP04)*, June 2004.
- [239] Jordi Alvarez Canal. Parametric aspects: A proposal. In Cazzola et al. [255], pages 91–100.
- [240] Sabine Canditt and Manfred Gunter. Aspect oriented logging in a real-world system. In Coady et al. [327].
- [241] Fei Cao, Barrett R. Bryant, Carol C. Burt, Rajeev R. Raje, Andrew M. Olson, and Mikhail Auguston. A component assembly approach based on aspect-oriented generative domain modeling. In Uwe Aßmann and Adrian Pop, editors, *Software Composition Workshop (ETAPS)*, pages 91–125, March 2004.
- [242] Adeline Capouillez, Pierre Crescenzo, and Philippe Lahire. Separation of concerns in OFL. In Bergmans et al. [154].
- [243] Richard Cardone, Adam Brown, Sean McDirmid, and Calvin Lin. Using mixins to build flexible widgets. In Kiczales [781], pages 76–85.
- [244] Richard Cardone and Calvin Lin. Using mixin technology to improve modularity. In Filman et al. [513], pages 219–241.
- [245] Lee Carver. Combining selector-guarded blocks. In Lopes et al. [913].
- [246] Lee Carver. A practical Hyperspace application: Lessons from the option-processing task. In Tarr et al. [1363].
- [247] Lee Carver. Using brackets to corral jumping aspects. In Tarr et al. [1362].
- [248] Lee Carver. Building a real-world application with aspect-oriented modules and Hyper/J. Master’s thesis, University of California, San Diego, June 2002.
- [249] Lee Carver. Composition behaviors for application construction. In Glandrup et al. [564].
- [250] Lee Carver. Simplified universe construction for Hyper/J composition. In Chu-Carroll et al. [286].
- [251] Lee Carver. Next steps for commercializing AOP. In Bodkin et al. [185].
- [252] Lee Carver and William G. Griswold. Sorting out concerns. In Ossher et al. [1091].

- [253] Jaelson Castro, Carlos Cuesta, Xavier Franch, Alessandro Garcia, Manuel Menezes Sequeira, Jennifer Pérez, and Miguel Pessoa Monteiro, editors. *Iberian workshop on Aspect Oriented Software Development (DSOA-2005)*, September 2005.
- [254] Jaelson Castro and Georgia Sousa. Supporting separation of concerns in requirements artifacts. In Garcia et al. [549].
- [255] Walter Cazzola, Shigeru Chiba, and Gunter Saake, editors. *ECOOP'04 Workshop on Reflection, AOP, and Meta-Data for Software Evolution (RAM-SE)*, June 2004.
- [256] Walter Cazzola, Sonia Pini, and Massimo Ancona. Evolving pointcut definition to get software evolution. In Cazzola et al. [255], pages 83–90.
- [257] Mariano Ceccato and Paolo Tonella. Measuring the effects of software aspectization. In Tourwé et al. [1398].
- [258] Vasian Cepa. Addressing ubiquitous software complexity with mobile containers. In Lorenz and Coady [932].
- [259] Vasian Cepa and Sven Kloppenburg. Representing explicit attributes in uml. In Aldawud et al. [42].
- [260] Matt Chapman. Aop@work: New ajdt releases ease aop development. Technical report, IBM Developer Works, August 2005.
- [261] Gary Chastek and John D. McGregor. Early aspects in software product line in product production. In Araújo et al. [79].
- [262] Rafael Chaves and Luiz Carlos Zancanella. Modelos executáveis baseados em aspectos. In Garcia et al. [549].
- [263] Christina Chavez, Alessandro Garcia, and Carlos Lucena. Some insights on the use of AspectJ and Hyper/J. In Rashid [1181].
- [264] Christina Chavez and Carlos Lucena. A metamodel for aspect-oriented modeling. In Aldawud et al. [37].
- [265] Christina Chavez and Carlos Lucena. Guidelines for aspect-oriented design. In Garcia et al. [549].
- [266] Marsha Chechik and Steve Easterbrook. Reasoning about compositions of concerns. In Tarr and Ossher [1366].
- [267] K. Chen and C.M. Huang. A practical aspect framework for enforcing fine-grained access control in web applications. In *Proc. of First Information Security Practice and Experience Conference (ISPEC 2005)*, volume 3439 of *LNCS*, pages 156–167. Springer-Verlag, 2005.
- [268] Kung Chen and Chih-Mao Huang. On designing access control aspects for web applications. In Bergmans et al. [156].
- [269] Li-Te Cheng, John Patterson, Steven Rohall, Susanne Hupfer, and Steven Ross. Weaving a social fabric into existing software. In Tarr [1361], pages 147–158.
- [270] Daniel Cheung-Foo-Woo, Mireille Blay-Fornarino, Jean-Yves Tigli, Anne-Marie Dery, and David Emsellem and Michel Riveill. Langage d’aspects pour la composition dynamique de composants embarqués. In Seinturier [1265].
- [271] S. Chiba, Y. Sato, and M. Tatsubori. Using HotSwap for implementing dynamic AOP systems. In Filman et al. [501].
- [272] Shigeru Chiba. What are the best join points? In De Volder et al. [407].
- [273] Shigeru Chiba and Kiyoshi Nakagawa. Josh: an open AspectJ-like language. In Lieberherr [884], pages 102–111.
- [274] R. Chitchyan, M. Pinto, and A. Rashid L. Fuentes. Relating ao requirements to ao architecture. In Baniassad et al. [109].
- [275] Ruzanna Chitchyan, Awais Rashid, Pete Sawyer, Alessandro Garcia, Mónica Pinto Alarcon, Jethro Bakker, Bedir Tekinerdogan, and Andrew Jackson Siobhán Clarke and. Survey of aspect-oriented analysis and design approaches. Technical Report AOSD-Europe-ULANC-9, AOSD-Europe, May 2005.
- [276] Ruzanna Chitchyan and Ian Sommerville. AOP and reflection for dynamic hyperslices. In Cazzola et al. [255], pages 29–36.
- [277] Ruzanna Chitchyan and Ian Sommerville. Comparing dynamic AO systems. In Filman et al. [502], pages 23–36.

- [278] Ruzanna Chitchyan, Ian Sommerville, and Awais Rashid. An analysis of design approaches for crosscutting concerns. In Glan-drup et al. [564].
- [279] Ruzanna Chitchyan, Ian Sommerville, and Awais Rashid. A model for dynamic hyper-spaces. In Bergmans et al. [153].
- [280] Ziéed Choukair, editor. *Proc. Int'l Workshop on Distributed Dynamic Multiservice Architectures (ICDCS-2001), Vol. 2*, April 2001.
- [281] Mark Chu-Carroll. Software configuration management as a mechanism for multidimensional separation of concerns. In Tarr et al. [1363].
- [282] Mark C. Chu-Carroll. Separation of concerns: An organizational approach. In Tarr et al. [1362].
- [283] Mark C. Chu-Carroll. Separation of concerns in software configuration management. In Tarr and Ossher [1366].
- [284] Mark C. Chu-Carroll. Separation of concerns in software configuration management. In Bergmans et al. [154].
- [285] Mark C. Chu-Carroll. Supporting aspects in program storage. In Chu-Carroll et al. [286].
- [286] Mark C. Chu-Carroll, Gail C. Murphy, Siobhan Clarke, Jacky Estublier, Anthony Finkelstein, Bill Harrison, and Elissa Newman, editors. *Workshop on Tools for Aspect-Oriented Software Development (OOPSLA 2002)*, November 2002.
- [287] Mark C. Chu-Carroll, James Wright, and Annie T. T. Ying. Visual separation of concerns through multidimensional program storage. In Aksit [26], pages 188–197.
- [288] William Chung, William Harrison, Vincent Kruskal, Harold Ossher, Jr. Stanley M. Sutton, Peri Tarr, Matthew Chapman, Andrew Clement, Helen Hawkins, and Sian January. The concern manipulation environment (demonstration). In *ICSE '05: Proceedings of the 27th international conference on Software engineering*, pages 666–667, New York, 2005. ACM Press.
- [289] William Chung, William Harrison, Vincent Kruskal, Harold Ossher, Stanley M. Sutton Jr., and Peri Tarr. Working with implicit concerns in the concern manipulation environment. In Tourwé et al. [1399].
- [290] Maria Agustina Cibran and Maja D'Hondt. Composable and reusable business rules using AspectJ. In Bergmans et al. [153].
- [291] María Agustina Cibrán, Maja D'Hondt, and Viviane Jonckers. Mapping high-level business rules to and through aspects. In Seinturier [1265].
- [292] Maria Agustina Cibran and Bart Verheecke. Dynamic business rules for web service composition. In Filman et al. [515], pages 13–18.
- [293] M. Cilia, M. Haupt, M. Mezini, and A. P. Buchmann. The convergence of AOP and active databases: Towards reactive middleware. In F. Pfenning and Y. Smaragdakis, editors, *Proceedings of 2nd International Conference on Generative Programming and Component Engineering (GPCE)*, volume 2830 of *LNCS*, pages 169–188. Springer-Verlag, 2003.
- [294] Tony Clark, Andy Evans, and Stuart Kent. Aspect-oriented metamodeling. *The Computer Journal*, 46(5):566–577, September 2003.
- [295] Siobhán Clarke. Composing design models: An extension to the UML. In *Proc. Third Int'l Conf. the Unified Modeling Language (UML), LNCS 1939*, pages 338–352. Springer-Verlag, October 2000.
- [296] Siobhán Clarke. Designing reusable patterns of cross-cutting behaviour with composition patterns. In Tarr et al. [1362].
- [297] Siobhán Clarke. Extending UML meta-model for design composition. In Tarr et al. [1363].
- [298] Siobhán Clarke. *Composition of Object-Oriented Software Design Models*. PhD thesis, Dublin City University, January 2001.
- [299] Siobhán Clarke. Extending standard UML with model composition semantics. *Science of Computer Programming*, 44(1):71–100, July 2002.

- [300] Siobhán Clarke, William Harrison, Harold Ossher, and Peri Tarr. Designing for evolution with subjects. In *Workshop on Software Change and Evolution, ICSE*, 1999.
- [301] Siobhán Clarke, William Harrison, Harold Ossher, and Peri Tarr. The dimensions of separating requirements concerns for the duration of the development lifecycle. In Ossher et al. [1091].
- [302] Siobhán Clarke, William Harrison, Harold Ossher, and Peri Tarr. Separating concerns throughout the development lifecycle. In Lopes et al. [924].
- [303] Siobhán Clarke, William Harrison, Harold Ossher, and Peri Tarr. Subject-oriented design: Support for evolution from the design stage. In *Workshop on Software and Organisation Co-Evolution*, 1999.
- [304] Siobhán Clarke, William Harrison, Harold Ossher, and Peri Tarr. Subject-oriented design: Towards improved alignment of requirements, design and code. In *Proc. Object-Oriented Programming, Systems, Languages and Applications (OOPSLA)*, pages 325–339, November 1999.
- [305] Siobhán Clarke, William Harrison, Harold Ossher, and Peri Tarr. Subject-oriented design: Towards improved alignment of requirements, design and code. Research Report RC 21300, IBM, April 1999.
- [306] Siobhán Clarke and John Murphy. Developing a tool to support composition of the components in a large-scale development. In *OOPSLA Workshop on Object-Oriented Behavioural Semantics*, 1997.
- [307] Siobhán Clarke and John Murphy. Developing a tool to support the application of aspect-oriented programming principles to the design phase. In Lopes et al. [926].
- [308] Siobhán Clarke and John Murphy. Verifying components under development at the design stage: A tool to support the composition of component design models. In *ICSE Workshop on Component-Based Software Engineering*, 1998.
- [309] Siobhán Clarke, John Murphy, and Mark Roantree. Composition of UML design models: A tool to support the resolution of conflicts. In *Proc. Object Oriented Information Systems (OOIS)*, September 1998.
- [310] Siobhán Clarke and Robert Walker. Towards a standard design language for AOSD. In Kiczales [781], pages 113–119.
- [311] Siobhán Clarke and Robert J. Walker. Composition patterns: An approach to designing reusable aspects. In *Proc. 23rd Int'l Conf. Software Engineering (ICSE)*, pages 5–14, May 2001.
- [312] Siobhán Clarke and Robert J. Walker. Mapping composition patterns to AspectJ and Hyper/J. In Bergmans et al. [154].
- [313] Siobhán Clarke and Robert J. Walker. Mapping composition patterns to AspectJ and Hyper/J. In Tarr and Ossher [1366].
- [314] Siobhán Clarke and Robert J. Walker. Separating crosscutting concerns across the lifecycle: From composition patterns to AspectJ and Hyper/J. Technical Report TCD-CS-2001-15, Trinity College, Dublin, May 2001.
- [315] Siobhán Clarke and Robert J. Walker. Generic aspect-oriented design with Theme/UML. In Filman et al. [513], pages 425–458.
- [316] Thomas Cleenewerck, Kris Gybels, and Adriaan Peeters. Aspects in a prototype-based environment. In Filman et al. [502], pages 37–43.
- [317] Andy Clement, Adrian Colyer, and Mik Kersten. Aspect-oriented programming with AJDT. In Hannemann et al. [640].
- [318] Pedro J. Clemente and Juan Hernández. Aspect component based software engineering. In Coady et al. [326].
- [319] Pedro J. Clemente, Juan Hernández, José Luis Herrero, Juan Manuel Murillo, and Fernando Sánchez. Aspect-orientation in the software lifecycle: Fact and fiction. In Filman et al. [513], pages 407–423.
- [320] Paul Clements, Awais Rashid, Elisa Baniassad, Bedir Tekinerdogan, Ana Moreira, and João Araújo, editors. *Early Aspects: Aspect-Oriented Requirements Engineering and Architecture Design*, March 2005.

- [321] Curtis Clifton, Ralf Lämmel, and Gary T. Leavens, editors. *FOAL: Foundations Of Aspect-Oriented Languages*, March 2004.
- [322] Curtis Clifton and Gary T. Leavens. Observers and assistants: A proposal for modular aspect-oriented reasoning. In Cytron and Leavens [379], pages 33–44.
- [323] Curtis Clifton and Gary T. Leavens. Obliviousness, modular reasoning, and the behavioral subtyping analogy. In Bergmans et al. [153].
- [324] Curtis Clifton and Gary T. Leavens. MiniMAO: Investigating the semantics of proceed. In Leavens et al. [867].
- [325] Yvonne Coady, Alex Brodsky, Dima Brodsky, Jody Pomkoski, Stephan Gudmundson, Joon Suan Ong, and Gregor Kiczales. Can AOP support extensibility in client-server architectures? In Bergmans et al. [154].
- [326] Yvonne Coady, Eric Eide, and David H. Lorenz, editors. *The Second AOSD Workshop on Aspects, Components, and Patterns for Infrastructure Software (ACP4IS)*, March 2003.
- [327] Yvonne Coady, Eric Eide, David H. Lorenz, Mira Mezini, Klaus Ostermann, and Roman Pichler, editors. *First AOSD Workshop on Aspects, Components, and Patterns for Infrastructure Software (AOSD-2002)*, March 2002.
- [328] Yvonne Coady, Eric Eide, David H. Lorenz, and Olaf Spinyzck, editors. *Proceedings of the Fourth AOSD Workshop on Aspects, Components, and Patterns for Infrastructure Software*, March 2005.
- [329] Yvonne Coady and Gregor Kiczales. Back to the future: A retroactive study of aspect evolution in operating system code. In Akşit [26], pages 50–59.
- [330] Yvonne Coady, Gregor Kiczales, and Michael Feeley. Exploring an aspect-oriented approach to operating system code. In Tarr et al. [1362].
- [331] Yvonne Coady, Gregor Kiczales, Mike Feeley, Norm Hutchinson, and Joon Suan Ong. Structuring operating system aspects: Using AOP to improve OS structure modularity. *Comm. ACM*, 44(10):79–82, October 2001.
- [332] Yvonne Coady, Gregor Kiczales, Mike Feeley, Norm Hutchinson, and Joon Suan Ong. Structuring system aspects. In Tarr and Ossher [1366].
- [333] Yvonne Coady, Gregor Kiczales, Mike Feeley, Norm Hutchinson, and Joon Suan Ong. Structuring operating system aspects. In Filman et al. [513], pages 651–656.
- [334] Yvonne Coady, Gregor Kiczales, Mike Feeley, and Greg Smolyn. Using AspectC to improve the modularity of path-specific customization in operating system code. In *Proceedings of the 8th European Software Engineering Conference held jointly with 9th ACM SIGSOFT Symposium on Foundations of Software Engineering*, pages 88–98. ACM Press, 2001.
- [335] Wesley Coelho and Gail C. Murphy. Activeaspect: presenting crosscutting structure. In Robillard [1212], pages 1–4.
- [336] G. Cohen, J. Chase, and D. Kaminsky. Automatic program transformation with JOIE. In *1998 USENIX Annual Technical Symposium*, pages 167–178, 1998.
- [337] Geoff Cohen. Using bytecode transformation to integrate new features. In Filman et al. [513], pages 297–310.
- [338] Geoff A. Cohen. Recombing concerns: Experience with transformation. In Ossher et al. [1091].
- [339] Geoff A. Cohen. A taxonomy of transformation. In Tarr et al. [1362].
- [340] T. Colcombet and P. Fradet. Enforcing trace properties by program transformation. In *Proc. 27th ACM Symp. on Principles of Programming Languages*, pages 54–66, January 2000.
- [341] Leonardo Cole and Paulo Borba. Using programming laws to modularize concurrency in a replicated database application. In Garcia et al. [549].
- [342] Leonardo Cole and Paulo Borba. Deriving refactorings for AspectJ. In Tarr [1361], pages 123–134.
- [343] Leonardo Cole, Paulo Borba, and Alexandre Mota. Proving aspect-oriented programming laws. In Leavens et al. [867].

- [344] Adrian Colyer. Towards widespread adoption of AOSD. In Bodkin et al. [185].
- [345] Adrian Colyer. AspectJ. In Filman et al. [513], pages 123–143.
- [346] Adrian Colyer, Gordon Blair, and Awais Rashid. Managing complexity in middleware. In Coady et al. [326].
- [347] Adrian Colyer and Andrew Clement. Large-scale AOSD for middleware. In Lieberherr [884], pages 56–65.
- [348] Adrian Colyer, Andy Clement, George Harley, and Matthew Webster. *Eclipse AspectJ: Aspect-Oriented Programming with AspectJ and the Eclipse AspectJ Development Tools*. Pearson Education, 2005.
- [349] J. Conejero, J. Hernández, and J. Pedrero. Definición de un perfil UML para el aspecto de notificación en entornos distribuidos CORBA. In Fuentes et al. [538].
- [350] José Conejero, Juan Hernández, and Roberto Rodríguez. UML profile definition for dealing with the notification aspect in distributed environments. In Kandé et al. [737].
- [351] C. Constantinides and T. Skotiniotis. Reasoning about a classification of cross-cutting concerns in object-oriented systems. In Costanza et al. [371]. Technical report IAI-TR-2002-1.
- [352] Constantinos Constantinides, Atef Bader, and Tzilla Elrad. An aspect-oriented design framework for concurrent systems. In Lopes et al. [924].
- [353] Constantinos A. Constantinides. A case study on making the transition from functional to and fine-grained decomposition. In Hannemann et al. [640].
- [354] Constantinos A. Constantinides, Atef Bader, and Tzilla Elrad. A framework to address a two-dimensional composition of concerns. In Ossher et al. [1091].
- [355] Constantinos A. Constantinides, Atef Bader, and Tzilla Elrad. Separation of concerns in concurrent software systems. In Lopes et al. [913].
- [356] Constantinos A. Constantinides, Atef Bader, Tzilla H. Elrad, P. Netinant, and Mohamed E. Fayad. Designing an aspect-oriented framework in an object-oriented environment. *ACM Computing Surveys*, 32(1es):41, 2000.
- [357] Constantinos A. Constantinides and Tzilla Elrad. On the requirements for concurrent software architectures to support advanced separation of concerns. In Tarr et al. [1362].
- [358] Constantinos A. Constantinides and Tzilla Elrad. Towards a two-dimensional separation of concerns (poster session). In *Addendum to the 2000 proceedings of the conference on Object-oriented programming, systems, languages, and applications (Addendum)*, pages 63–64. ACM Press, 2000.
- [359] Constantinos A. Constantinides and Tzilla Elrad. Composing concerns with a framework approach. In Choukair [280], pages 133–140.
- [360] Constantinos A. Constantinides, Tzilla Elrad, and Mohamed Fayad. Extending the object model to provide explicit support for crosscutting concerns. *Software Practice and Experience*, 32(7):703–734, May 2002.
- [361] Constantinos A. Constantinides, Therapon Skotiniotis, and Tzilla Elrad. Providing dynamic adaptability in an aspect-oriented framework. In Bergmans et al. [154].
- [362] K. Cooper, L. Dai, Y. Deng, and J. Dong. Towards an aspect-oriented architectural framework. In Araújo et al. [81].
- [363] Kendra Cooper, Lirong Dai, and Yi Deng. Modeling performance as an aspect: A UML based approach. In Aldawud et al. [44].
- [364] David Coppit and Benjamin Cox. Software plans for separation of concerns. In Lorenz and Coady [932].
- [365] Mateus Costa, Rodolfo Ferreira Resende, Pedro Santos Neto, and Mírian Halfeld Ferrari. Utilização de aspectos no desenvolvimento de aplicações baseadas em serviços web. In Garcia et al. [549].
- [366] P. Costanza. Dynamically scoped functions for runtime modification. In Filman et al. [501].

- [367] Pascal Costanza. Separation of object identity concerns. In Lopes et al. [913].
- [368] Pascal Costanza. Vanishing aspects. In Tarr et al. [1362].
- [369] Pascal Costanza. A short overview of AspectL. In Gybels et al. [616].
- [370] Pascal Costanza, Günter Kniesel, and Michael Austermann. Independent extensibility for aspect-oriented systems. In Bergmans et al. [154].
- [371] Pascal Costanza, Günter Kniesel, Katharina Mehner, Elke Pulvermüller, and Andreas Speck, editors. *Second Workshop on Aspect-Oriented Software Development of the German Information Society*. Institut für Informatik III, Universität Bonn, February 2002. Technical report IAI-TR-2002-1.
- [372] T. Cottenier and T. Elrad. Layer of collaboration aspects for pervasive computing. In Lopes et al. [929].
- [373] Thomas Cottenier and Tzilla Elrad. Linguistic provisions for aspect/core semantic interactions. In Bergmans et al. [155].
- [374] Thomas Cottenier and Tzilla Elrad. Contextual pointcut expressions for dynamic service customization. In Filman et al. [515], pages 87–91.
- [375] Thomas Cottenier, Aswin Van Den Berg, and Tzilla Elrad. Modeling aspect-oriented compositions. In Aldawud et al. [42].
- [376] Carine Courbis and Anthony Finkelstein. Towards an aspect weaving BPEL engine. In Lorenz and Coady [932].
- [377] Carine Courbis and Anthony Finkelstein. Towards aspect weaving applications. In *ICSE '05: Proceedings of the 27th international conference on Software engineering*, pages 69–77, New York, 2005. ACM Press.
- [378] Pierre Crescenzo and Philippe Lahire. Une approche pour améliorer la réutilisabilité des modèles métiers. In Seinturier [1265].
- [379] Ron Cytron and Gary T. Leavens, editors. *FOAL 2002: Foundations of Aspect-Oriented Languages (AOSD-2002)*, March 2002.
- [380] Krzysztof Czarnecki and Ulrich W. Eisenecker. *Generative Programming: Methods, Tools, and Applications*. Addison-Wesley, Boston, 2000.
- [381] Krzysztof Czarnecki and Ulrich W. Eisenecker. Separating the configuration aspect to support architecture evolution. In Lopes et al. [913].
- [382] Krzysztof Czarnecki, Ulrich W. Eisenecker, and Patrick Steyaert. Beyond objects: Generative programming. In Lopes et al. [918].
- [383] Christian Dalager, Simon Jorsal, and Eske Sort. Aspect oriented programming in JBoss 4. Master's thesis, IT University of Copenhagen, February 2004.
- [384] Dangeti, Thirunavukkarasu, and Jeyabal. Runtime weaving of aspects using dynamic code instrumentation technique for building adaptive software systems. In Coady et al. [327].
- [385] Ayla Dantas and Paulo Borba. Adaptability aspects: An architectural pattern for structuring adaptive applications with aspects. In *Third Latin American Conference on Pattern Languages of Programming, SugarLoaf-PLoP'2003*, August 2003.
- [386] Ayla Dantas and Paulo Borba. Developing adaptive J2ME applications using AspectJ. In *Proceedings of VII Brazilian Symposium on Programming Languages, SBLP 2003*, pages 226–242, May 2003.
- [387] Ayla Dantas, Paulo Borba, and Vander Alves. Using aspects to structure small devices applications. In *First Workshop on Reuse in Constrained Environments (RICE'03), OOPSLA2003*, October 2003.
- [388] Ayla Dantas, Paulo Borba, Joseph Yoder, and Ralph Johnson. Using aspects to make adaptive object-models adaptable. In Cazola et al. [255], pages 9–20.
- [389] Daniel S. Dantas, David Walker, Geoffrey Washburn, and Stephanie Weirich. Polyaml: A polymorphic aspect-oriented functional programming language. In *International Conference on Functional Programming (ICFP)*, pages 306–319. ACM, September 2005.

- [390] Pierre-Charles David and Thomas Ledoux. Dynamic adaptation of non-functional concerns. In *1st Int'l Workshop on Unanticipated Software Evolution (USE) (ECOOP)*, June 2002.
- [391] Pierre-Charles David and Thomas Ledoux. Une approche par aspects pour le développement de composants fractal adaptatifs. In Seinturier [1265].
- [392] Pierre-Charles David, Thomas Ledoux, and Noury M. N. Bouraqadi-Saâdani. Two-step weaving with reflection using AspectJ. In De Volder et al. [407].
- [393] Fabrício de Alexandria Fernandes and Thais Batista. Dynamic aspect-oriented programming: An interpreted approach. In Filman et al. [502], pages 44–50.
- [394] Brian De Alwis, Stephan Gudmundson, Greg Smolyn, and Gregor Kiczales. Coding issues in AspectJ. In Tarr et al. [1362].
- [395] Silvia de Castro Bertagnolli and Maria Lúcia Blanck Lisbôa. The FRIDA model. In Hannemann et al. [640].
- [396] Bruno De Fraine, Wim Vanderperren, Davy Suvéé, and Johan Brichau. Jumping aspects revisited. In Filman et al. [515], pages 77–86.
- [397] Ricardo Soto De Giorgis and Nivaldo Rodríguez Agurto. New UML 2.0 based models to design WAP applications. In Aldawud et al. [38].
- [398] Wolfgang De Meuter. Monads as a theoretical foundation for AOP. In Lopes et al. [918].
- [399] A. L. de Moura, C. Ururahy, R. Cerque, and N. Rodriguez. Dynamic support for distributed auto-adaptive applications. In Akşit and Choukair [28].
- [400] Flavio de Paoli. Multidimensional separation of concerns. In Tarr et al. [1363].
- [401] A. M. P. de Resende, F. F. Silveira, and A. M. da Cunha. Early aspects: Some analysis, trends and perspectives. In Baniassad et al. [109].
- [402] K. De Volder, J. Brichau, K. Mens, and T. D'Hondt. Logic meta-programming, a framework for domain-specific aspect programming languages. <http://www.cs.ubc.ca/~kdvolder/binaries/cacm-aop-paper.pdf>.
- [403] Kris De Volder. Aspect-oriented logic meta programming. In Lopes et al. [914].
- [404] Kris De Volder. Inheritance with destructive mixins for better separation of concerns. In Tarr et al. [1362].
- [405] Kris De Volder. Code reuse, an essential concern in the design of aspect languages? In Bergmans et al. [154].
- [406] Kris De Volder and Theo D'Hondt. Aspect-oriented logic meta programming. In P. Cointe, editor, *Meta-Level Architectures and Reflection, 2nd Int'l Conf. Reflection*, volume 1616 of *LNCS*, pages 250–272. Springer Verlag, 1999.
- [407] Kris De Volder, Maurice Glandrup, Siobhán Clarke, and Robert Filman, editors. *Workshop on Advanced Separation of Concerns in Object-Oriented Systems (OOPSLA 2001)*, October 2001.
- [408] Kris De Volder, Tom Tourwé, and Johan Brichau. Logic meta programming as a tool for separation of concerns. In Lopes et al. [913].
- [409] B. De Win, J. van den Bergh, F. Matthijs, B. De Decker, and W. Joosen. A security architecture for electronic commerce applications. In S. Qing and J. Eloff, editors, *Information Security for Global Information Infrastructures*, pages 491–500. Kluwer Academic Publishers, 2000.
- [410] B. De Win, B. Vanhaute, and B. De Decker. Security through aspect-oriented programming. In B. De Decker, F. Piessens, J. Smits, and F. van Herreweghen, editors, *Advances in Network and Distributed Systems Security*, volume 206 of *IFIP Conf. Proc.*, pages 125–138. Kluwer Academic Publishers, 2001.
- [411] Bart De Win, Wouter Joosen, and Frank Piessens. AOSD & security: A practical assessment. In Bergmans et al. [153].
- [412] Bart De Win, Wouter Joosen, and Frank Piessens. Developing secure applications through aspect-oriented programming. In Filman et al. [513], pages 633–650.

- [413] Bart De Win, Viren Shah, Wouter Joosen, and Ron Bodkin, editors. *AOSDSEC: AOSD Technology for Application-Level Security*, March 2004.
- [414] Bart De Win, Bart Vanhaute, and Bart De Decker. How aspect-oriented programming can help to build secure software. *Informatica*, 26(2):141–149, 2001.
- [415] Bart De Win, Bart Vanhaute, and Bart De Decker. Towards an open weaving process. In De Volder et al. [407].
- [416] Douglas R. Dechow. Exploiting the possibilities of “weave-time” aspects in the creation of component-based ecological models. In Coady et al. [327].
- [417] Demeter Research Group. Online material on adaptive programming Demeter/Java, and APPCs. <http://www.ccs.neu.edu/research/demeter/>.
- [418] John Dempsey and Vinny Cahill. Aspects of system support for distributed computing. In Lopes et al. [918].
- [419] Giovanni Denaro and Mattia Monga. An experience on verification of aspect properties. In *Proceedings of the 4th international workshop on Principles of software evolution*, pages 186–189. ACM Press, 2002.
- [420] Simon Denier, Hervé Albin-Amiot, and Pierre Cointe. Expression and composition of design patterns with aspects. In Seinturier [1265].
- [421] Lieven Desmet, Frank Piessens, and Wouter Joosen. Infrastructural support for data dependencies in data-centered software systems. In Lorenz and Coady [932].
- [422] Morgan Deters and Ron K. Cytron. Introduction of program instrumentation using aspects. In De Volder et al. [407].
- [423] Morgan Deters, Nick Leidenfrost, and Ron K. Cytron. Translation of Java to real-time Java using aspects. In Rashid [1181].
- [424] Benet Devereux. Compositional reasoning about aspects using alternating-time logic. In Leavens and Clifton [866].
- [425] Maja D’Hondt and Theo D’Hondt. Is domain knowledge an aspect? In Lopes et al. [924].
- [426] Maja D’Hondt and Viviane Jonckers. Hybrid aspects for weaving object-oriented functionality and rule-based knowledge. In Lieberherr [884], pages 132–140.
- [427] Antonella Di Stefano, Giuseppe Pappalardo, Corrado Santoro, Emiliano Tramontana, and Ian Welch, editors. *Track on Programming for Separation of Concerns*, April 2006.
- [428] Klissiomara Dias and Paulo Borba. Padrões de projeto para estruturação de aplicações distribuídas Enterprise JavaBeans. In *Second Latin American Conference on Pattern Languages of Programming, SugarLoaf-PLoP’2002*, August 2002. Published in ICMC - University of São Paulo Magazine, December, 2002, pages 55-86.
- [429] Jorge L. Diaz-Herrera, Jasmin Chadha, and Neil Pittsley. Aspect-oriented UML modeling for developing embedded systems product lines. In Aldawud et al. [37].
- [430] A. Díaz Pace, M. Campo, and F. Trilnik. Smartweaver: Aspect-oriented development using the Smartbooks approach. In *Argentine Symp. Software Engineering (ASSE)*, pages 1–21, September 2002.
- [431] J. Andrés Díaz Pace and Marcelo R. Campo. Analyzing the role of aspects in software design. *Comm. ACM*, 44(10):66–73, October 2001.
- [432] J. Andrés Díaz Pace, Marcelo R. Campo, and Federico U. Trilnik. Smartweaver: A knowledge-driven approach for aspect composition. In Filman et al. [513], pages 605–626.
- [433] J. Andrés Díaz Pace, Mohamed E. Fayad, and Marcelo R. Campo. A language for simulation: Bringing separation to the front. In Lopes et al. [913].
- [434] J. Andrés Díaz Pace, F. Trilnik, and Marcelo R. Campo. How to handle interacting concerns? In Tarr et al. [1362].
- [435] Christopher Diggins. Aspect-oriented programming & C++. *Dr. Dobbs Journal*, August 2004.
- [436] Edsger W. Dijkstra. *A discipline of programming*. Prentice-Hall, Englewood Cliffs, New Jersey, 1976.

- [437] A. Dingwal-Smith and A. Finkelstein. From requirements to monitors by way of aspects. In Rashid et al. [1191].
- [438] Filippo Diotalevi. Contract enforcement with AOP: Apply design by contract to Java software development with AspectJ. Technical report, IBM, July 2004.
- [439] Peter Dolog, Valentino Vranić, and Mária Bieliková. Representing change by aspect. *SIGPLAN Not.*, 36(12):77–83, 2001.
- [440] Lutz Dominick. Aspect of lifecycle control in a C++ framework. In Lopes et al. [924].
- [441] Lutz Dominick. Instrumentation aspects require symmetric join points. In Lopes et al. [913].
- [442] Lutz Dominick and Klaus Ostermann. Supporting extension of components with new paradigms. In Tarr et al. [1362].
- [443] R. Douence, P. Fradet, and M. Südholt. Detection and resolution of aspect interactions. Technical Report RR-4435, INRIA, April 2002.
- [444] R. Douence and M. Südholt. A model and a tool for event-based aspect-oriented programming (EAOP). Technical Report 02/11/INFO, Ecole des Mines de Nantes, 2002.
- [445] Rémi Douence. A restricted definition of AOP. In Gybels et al. [616].
- [446] Rémi Douence, Pascal Fradet, and Mario Südholt. A framework for the detection and resolution of aspect interactions. In *1st Conf. Generative Programming and Component Engineering*, volume 2487 of *lncs*, pages 173–188, Berlin, 2002. Springer-Verlag.
- [447] Rémi Douence, Pascal Fradet, and Mario Südholt. Composition, reuse and interaction analysis of stateful aspects. In Lieberherr [884], pages 141–150.
- [448] Remi Douence, Pascal Fradet, and Mario Südholt. Trace-based aspects. In Filman et al. [513], pages 201–217.
- [449] Rémi Douence and Narendra Jussien. Non-intrusive constraint solver enhancements. In Coady et al. [327].
- [450] Rémi Douence, Olivier Motelet, and Mario Südholt. A formal definition of crosscuts. In Yonezawa and Matsuoka [1494], pages 170–186.
- [451] Rémi Douence, Olivier Motelet, and Mario Südholt. Sophisticated crosscuts for e-commerce. In Bergmans et al. [154].
- [452] Rémi Douence and Jacques Noyé. Towards a concurrent model of event-based aspect-oriented programming. In Gybels et al. [615].
- [453] Rémi Douence and Luc Teboul. A cross-cut language for control-flow. In Karsai and Visser [745], pages 95–114.
- [454] Rémi Douence, Fritz Thomas, Nicolas Lorient, Jean-Marc Menaud, Marc Séguira-Devillechaise, and Mario Südholt. An expressive aspect language for system applications with Arachne. In Tarr [1361], pages 27–38.
- [455] Jim Dowling and Vinny Cahill. The k-component architecture meta-model for self-adaptive software. In Yonezawa and Matsuoka [1494], pages 81–88.
- [456] Desmond D’Souza, Aamod Sane, and Alan Birchenough. First-class extensibility for UML—Packaging of profiles, stereotypes, patterns. In Ossher et al. [1091].
- [457] C. Duan, J. Cleland-Huang, R. Settimi, and X. Zou. Decision making support for early aspects identification based on probabilistic trace retrieval. In Clements et al. [320].
- [458] Frederic Duclos, Jacky Estublier, and Philippe Morat. Describing and using non functional aspects in component based applications. In Kiczales [781], pages 65–75.
- [459] Dominic Duggan. A mixin-based, semantics-based approach to reusing domain-specific programming languages. In Bertino [163], pages 179–200.
- [460] Pascal Durr, Tom Staijen, Lodewijk Bergmans, and Mehmet Aksit. Reasoning about semantic conflicts between aspects. In Gybels et al. [615].
- [461] Gary Duzan, Joseph Loyall, Richard Schantz, Richard Shapiro, and John Zinky. Building adaptive distributed applications

- with middleware and aspects. In Lieberherr [884], pages 66–73.
- [462] Peter Ebraert and Éric Tanter. A concern-based approach to dynamic software evolution. In Filman et al. [502], pages 51–55.
- [463] Peter Ebraert and Tom Tourwé. A reflective approach to dynamic software evolution. In Cazzola et al. [255], pages 37–44.
- [464] Eclipse homepage. <http://www.eclipse.org>.
- [465] Michael Eichberg. MDA and programming languages. In *Workshop on Generative Techniques in the context of Model Driven Architecture (OOPSLA '02)*, November 2002.
- [466] Michael Eichberg. The proxy inter-type declaration. In Lorenz and Coady [932].
- [467] Eric Eide, Alastair Reid, Matthew Flatt, and Jay Lepreau. Aspect weaving as component knitting: Separating concerns with knit. In Tarr and Ossher [1366].
- [468] Torbjorn Ekman. Separation of concerns in compiler construction using JastAdd II. In Lorenz and Coady [932].
- [469] I. El-Maddah and T. Maibaum. Tracing aspects in goal driven requirements of process control systems. In Tekinerdoğan et al. [1378], pages 33–41.
- [470] Tzilla Elrad, Mehmet Akşit, Gregor Kiczales, Karl Lieberherr, and Harold Ossher. Discussing aspects of AOP. *Comm. ACM*, 44(10):33–38, October 2001.
- [471] Tzilla Elrad, Omar Aldawud, and Atef Bader. Aspect-oriented modeling: Bridging the gap between implementation and design. In *ACM SIGPLAN/SIGSOFT Conference on Generative Programming and Component Engineering (GPCE'02)*, October 2002.
- [472] Tzilla Elrad, Omar Aldawud, and Atef Bader. Expressing aspects using UML behavioral and structural diagrams. In Filman et al. [513], pages 459–478.
- [473] Tzilla Elrad, Robert E. Filman, and Atef Bader. Aspect-oriented programming. *Comm. ACM*, 44(10):29–32, October 2001.
- [474] Michael Engel and Bernd Freisleben. Supporting autonomic computing functionality via dynamic operating system kernel aspects. In Tarr [1361], pages 51–62.
- [475] Michael Engel and Bernd Freisleben. Using a low-level virtual machine to improve dynamic aspect support in operating system kernels. In Coady et al. [328].
- [476] Erik Ernst. Separation of concerns and then what. In Lopes et al. [913].
- [477] Erik Ernst. Syntax based modularization: Invasive or not? In Tarr et al. [1362].
- [478] Erik Ernst. Loosely coupled class families. In Bergmans et al. [154].
- [479] Erik Ernst. Separation of concerns. In Bergmans et al. [153].
- [480] Erik Ernst. Simple, eh? In Bergmans et al. [155].
- [481] Erik Ernst. The hardship of software. In Bergmans et al. [156].
- [482] Erik Ernst and David H. Lorenz. Aspects and polymorphism in AspectJ. In Akşit [26], pages 150–157.
- [483] M. J. Escalona, A. M. Reina, J. Torres, and M. Mejías. NDT: a methodology to deal with the navigation aspect at the requirements phase. In Moreira et al. [1017].
- [484] Clément Escoffier and Didier Donsez. Fractnet - une implémentation du modèle á composant fractal pour .net. In Seinturier [1265].
- [485] Y. Eterovich, J. M. Murillo, and K. Palma. Managing components adaptation using aspect oriented techniques. In Canal et al. [238].
- [486] Ran Ettinger and Mathieu Verbaere. Untangling: a slice extraction refactoring. In Lieberherr [884], pages 93–101.
- [487] J. Fabry. Replication as an aspect. In Lopes et al. [914].
- [488] Johan Fabry. Transaction management in EJBs: Better separation of concerns with AOP. In Lorenz and Coady [932].
- [489] Johan Fabry, Johan Brichau, and Tom Mens. Moving code. In Bergmans et al. [154].

- [490] Johan Fabry and Thomas Cleenerck. Aspect-oriented domain-specific languages for advanced transaction management. In Seinturier [1265].
- [491] Houssam Fakih, Noury Bouraqadi, and Laurence Duchien. Aspects and software components: A case study of the FRACTAL component model. In Huang et al. [701].
- [492] Houssam Fakith, Noury Bouraqadi, and Laurence Duchien. Towards unifying aspects and components. In Lorenz and Coady [932].
- [493] Paolo Falcarin, Mario Baldi, and Daniele Mazzocchi. Software tampering detection using AOP and mobile code. In De Win et al. [413].
- [494] M. E. Fayad, R. S. Pradeep, and F. Seddiqui. Aspects in communications: Performance. In Aldawud et al. [44].
- [495] M. E. Fayad and Anita Ranganath. Modeling aspects using software stability and UML. In Aldawud et al. [44].
- [496] Fabricio Fernandes, Nelio Cacho, and Thais Batista. LuaMOP — A meta-object protocol for dynamic weaving. In Garcia et al. [549].
- [497] Ricardo Ferreira, Ricardo Raminhos, and Ana Moreira. Metadata driven aspect specification. In Aldawud et al. [42].
- [498] Guilherme Filho, Geovani Santangelo, João Gualberto Rizzo, and Edeyson Gomes. AJEFW: Framework orientado a aspectos para tratamento de exceções. In Garcia et al. [549].
- [499] R. E. Filman, S. Barrett, D. D. Lee, and T. Linden. Inserting ilities by controlling communications. *Comm. ACM*, 45(1):116–122, January 2002.
- [500] R. E. Filman and D. P. Friedman. Aspect-oriented programming is quantification and obliviousness. In Tarr et al. [1362].
- [501] Robert Filman, Michael Haupt, and Katharina Mehner, editors. *1st Workshop on Advancing the State-of-the-Art in Run-Time Inspection (ECOOP 2003)*, July 2003.
- [502] Robert Filman, Michael Haupt, Katharina Mehner, and Mira Mezini, editors. *DAW: Dynamic Aspects Workshop*, March 2004.
- [503] Robert E. Filman. Retrofitting objects. In *ACM Conf. Object Oriented Programming Systems, Languages, and Applications (OOPSLA-87)*, pages 342–353, October 1987.
- [504] Robert E. Filman. Achieving ilities. In Linden and Thompson [901].
- [505] Robert E. Filman. Injecting ilities. In Lopes et al. [926].
- [506] Robert E. Filman. Managing ilities. In *Component-Based Software Engineering Workshop (ICSE)*, pages 81–85, April 1998.
- [507] Robert E. Filman. Applying aspect-oriented programming to intelligent synthesis. In Lopes et al. [913].
- [508] Robert E. Filman. A software architecture for intelligent synthesis environments. In *Proc. 2001 IEEE Aerospace Conference*, pages 2879–2888, March 2001.
- [509] Robert E. Filman. What is aspect-oriented programming, revisited. In Bergmans et al. [154].
- [510] Robert E. Filman. Understanding AOP through the study of interpreters. In Leavens and Clifton [866].
- [511] Robert E. Filman. A bibliography of aspect-oriented programming, version 2.0. Technical report, November 2005.
- [512] Robert E. Filman, Stuart Barrett, Diana D. Lee, and Ted Linden. Inserting ilities by controlling communications. In Filman et al. [513], pages 283–295.
- [513] Robert E. Filman, Tzilla Elrad, Siobhán Clarke, and Mehmet Akşit, editors. *Aspect-Oriented Software Development*. Addison-Wesley, Boston, 2005.
- [514] Robert E. Filman and Daniel P. Friedman. Aspect-oriented programming is quantification and obliviousness. In Filman et al. [513], pages 21–35.
- [515] Robert E. Filman, Michael Haupt, and Robert Hirschfeld, editors. *Dynamic Aspects Workshop*, March 2005.

- [516] Robert E. Filman and Klaus Havelund. Realising aspects by transforming for events. In Kris De Volder, Kim Mens, Tom Mens, and Roel Wuyts, editors, *Proc. Workshop on Declarative Meta Programming to Support Software Development*, September 2002.
- [517] Robert E. Filman and Klaus Havelund. Source-code instrumentation and quantification of events. In Cytron and Leavens [379], pages 45–49.
- [518] Robert E. Filman and Klaus Havelund. The effect of AOP on software engineering, with particular attention to OIF and event quantification. In Bergmans et al. [153].
- [519] Robert E. Filman, David J. Kormeyer, and Diana D. Lee. A CORBA extension for intelligent software environments. *Advances in Engineering Software*, 31(8-9):727–732, 2000.
- [520] Robert E. Filman and Diana D. Lee. Redirecting by injector. In Choukair [280], pages 141–146.
- [521] A. C. W. Finkelstein, D. Gabbay, A. Hunter, J. Kramer, and B. Nuseibeh. Inconsistency handling in multiperspective specifications. *IEEE Trans. Soft. Eng.*, 20(8):569–578, 1994.
- [522] Daniela Florescu, Andreas Grünhagen, and Donald Kossman. XL: An XML programming language for web service specification and composition. In *Proc. The Eleventh Int'l World Wide Web Conference*, pages 65–76, May 2002.
- [523] Marcus Fontura. Dimension templates: Multi-dimensional separation of concerns in UML. In Ossher et al. [1091].
- [524] Ira R. Forman. Superimposition: A form of separation of concerns for distributed systems. In Tarr et al. [1362].
- [525] P. Fradet and M. Südholt. AOP: Towards a generic framework using program transformation and analysis. In Lopes et al. [914].
- [526] P. Fradet and M. Südholt. An aspect language for robust programming. In Lopes et al. [924].
- [527] Pascal Fradet and Stéphane Hong Tuan Ha. Systèmes de gestion de ressources et aspects de disponibilité. In Seinturier [1265].
- [528] Pascal Fradet and Ralf Lämmel. Special issue on foundations of aspect-oriented programming. *Science of Computer Programming*, xxxx.
- [529] R. France, I. Ray, G. Georg, and S. Ghosh. Aspect-oriented approach to early design modelling. *IEE Proceedings Software*, 151(4):173–185, August 2004.
- [530] Robert France, Geri Georg, and Indrakshi Ray. Supporting multidimensional separation of concerns. In Aldawud et al. [43].
- [531] Elke Franz and Christoph Pohl. Towards unified treatment of security and other non-functional properties. In De Win et al. [413].
- [532] Paul Freeman. A position on an aspect oriented approach to the observer pattern. In Bergmans et al. [153].
- [533] Stephen T. Frezza. The ubiquitous overlapping, dynamic concerns project vs. product. In Tarr and Ossher [1366].
- [534] Thomas Fritz, Marc Ségura, Mario Südholt, Egon Wuchner, and Jean-Marc Menaud. An application of dynamic AOP to medical image generation. In Filman et al. [515], pages 5–12.
- [535] *Foundations of Software Engineering (FSE)*. ACM, October 2004.
- [536] Lidia Fuentes, Daniel Jimenez, and Monica Pinto. Towards the development of ambient intelligence environments using aspect-oriented techniques. In Lorenz and Coady [932].
- [537] Lidia Fuentes, Daniel Jimenez, and Monica Pinto. Experiences refactoring ambient intelligence applications with aspects. In Tourwé et al. [1399].
- [538] Lidia Fuentes, Ana Moreira, and Juan Manuel Murillo, editors. *Iberian workshop on Aspect Oriented Software Development (DSOA04)*, November 2004.
- [539] Lidia Fuentes, Mónica Pinto, and Pablo Sánchez. Dynamic weaving in CAM/DAOP: An application architecture driven approach. In Filman et al. [515], pages 33–39.
- [540] A. Furfaro, L. Nigro, and F. Pupo. Aspect oriented programming using actors. In Akşit and Choukair [28].

- [541] Jason Furlong. Using aspect-oriented software development to build a scenario based access control reference monitor. Master's thesis, Royal Military College of Canada, 2005.
- [542] Critina Gacek and Michalis Anastasopoulos. Implementing product line variabilities. In *Proceedings of 2001 Symposium on Software Reusability : Putting Software Reuse in Context*, pages 109–117. ACM Press, 2001.
- [543] Patrice Gahide, Noury Bouraqadi, and Laurence Duchien. Promoting component reuse by integrating aspects and contracts in an architecture model. In Coady et al. [327].
- [544] Andreas Gal, Michael Franz, and Danilo Beuche. Learning from components: Fitting AOP for system software. In Coady et al. [326].
- [545] Andreas Gal, Wolfgang Schröder-Preikschat, and Olaf Spinczyk. AspectC++: Language proposal and prototype implementation. In De Volder et al. [407].
- [546] Andreas Gal and Olaf Spinczyk. Build management for AspectC++. In Chu-Carroll et al. [286].
- [547] S. Gao, Z. Dai, and H. Yu. Improving scenario-driven role engineering process with aspects. In Moreira et al. [1017].
- [548] Alessandro Garcia, Christina Chavez, Otavio Silva, Viviane Silva, and Carlos Lucena. Promoting advanced separation of concerns in intra-agent and inter-agent software engineering. In De Volder et al. [407].
- [549] Alessandro Garcia, Carlos Lucena, Christina von Flach G. Chavez, Eduardo Kessler Piveta, Luiz Carlos Zancanella, Paulo Borba, and Sérgio Soares, editors. *First Brazilian Workshop on Aspect-Oriented Software Development (WASP) (SBES)*, October 2004.
- [550] Alessandro Garcia, Cláudio Sant'Anna, Eduardo Figueiredo, Uirá Kulesza, Carlos Lucena, and Arndt von Staa. Modularizing design patterns with aspects: A quantitative study. In Tarr [1361], pages 3–14.
- [551] Alessandro F. Garcia and Carlos J. P. de Lucena. An aspect-based object-oriented model for multi-agent systems. In Tarr and Ossher [1366].
- [552] David Garlan and David Notkin. Formalizing design spaces: Implicit invocation mechanisms. In *4th Int'l Symp. of VDM Europe on Formal Software Development-Volume I*, volume 552 of *lncs*, pages 31–44, Berlin, 1991. Springer-Verlag.
- [553] G. Georg, I. Ray, and R. France. Using aspects to design a secure system. In *Proc. 8th IEEE Int'l Conf. on Engineering of Complex Computer Systems*. IEEE CS Press, December 2002.
- [554] Geri Georg, Robert France, and Indrakshi Ray. Composing aspect models. In Aldawud et al. [44].
- [555] Naghmeh Ghafari, Alexander Lau, Barry Pekilis, James Thai, and Rudolph Seviora. H&V consistency checking for software health monitoring. In Coady et al. [327].
- [556] Celina Gibbs and Yvonne Coady. Garbage collection in Jikes: Could dynamic aspects add value? In Filman et al. [502], pages 56–63.
- [557] Celina Gibbs and Yvonne Coady. OASIS: Organic aspects for system infrastructure software easing evolution and adaptation through natural decomposition. In Cazzola et al. [255], pages 45–52.
- [558] Celina Gibbs, Chunjian Robin Liu, and Yvonne Coady. And the band played on: Are aspects adrift in a sea of sinking code? In Tourwé et al. [1399].
- [559] Holger Giese. Towards ruling component-based distributed systems with role-based modeling and cross-cutting aspects. In Tarr and Ossher [1366].
- [560] Holger Giese and Alexander Vilbig. Towards aspect-oriented design and architecture. In Tarr et al. [1362].
- [561] Wasif Gilani and Olaf Spinczyk. A family of aspect dynamic weavers. In Filman et al. [502], pages 64–75.
- [562] M. Glandrup. Extending C++ using the concepts of composition filters. Master's thesis, University of Twente, 1995.

- [563] Maurice Glandrup and Arend Rensink. Formal foundations for reasoning about evolution. In De Volder et al. [407].
- [564] Maurice Glandrup, Peri Tarr, Siobhán Clarke, and Faisal Akkawi, editors. *Workshop on Aspect Oriented Design — Identifying, Separating and Verifying Concerns in the Design (AOSD-2002)*, March 2002.
- [565] Robert Glück and Michael R. Lowry, editors. *Generative Programming and Component Engineering, 4th International Conference (GPCE)*, volume 3676 of *Lecture Notes in Computer Science*. Springer, September 2005.
- [566] Steffen Göbel, Christoph Pohl, Simone Röttger, and Steffen Zschaler. The COMQUAD component model: enabling dynamic selection of implementations by weaving non-functional aspects. In Lieberherr [884], pages 74–82.
- [567] Aniruddha Gokhale and Jeff Gray. An integrated aspect-oriented model-driven development toolsuite for distributed real-time and embedded systems. In Kandé et al. [737].
- [568] Carlos Gonzaález, Juan Manuel Murillo, and Pablo Amaya. Un modelo de propiedades y dependencias para el análisis orientado a aspectos en MDA. In Fuentes et al. [538].
- [569] C. González, J.M Murillo, and Pablo A. Amaya. Aspect-oriented analysis: A MDA based approach. In Clements et al. [320].
- [570] Joseph D. Gradecki and Nicholas Lesiecki. *Mastering AspectJ: Aspect-Oriented Programming in Java*. John Wiley and Sons, 2003.
- [571] Kasper Graversen and Kasper Østerbye. Aspects of aspects—A framework for discussion. In Gybels et al. [616].
- [572] Kasper B. Graversen. Role collaborations. In Bergmans et al. [155].
- [573] Kasper B. Graversen and Johannes Beye. Conceptual programming using roles. In Glandrup et al. [564].
- [574] Kasper B. Graversen and Kasper Østerbye. Aspect modelling as role modelling. In Chu-Carroll et al. [286].
- [575] Kasper B. Graversen and Kasper Østerbye. Implementation of a role language for object-specific dynamic separation of concerns. In Bergmans et al. [153].
- [576] Jeff Gray. Using software component generators to construct a meta-weaver framework. In *Proceedings of the 23rd International Conference on Software Engineering*, pages 789–790. IEEE Computer Society, 2001.
- [577] Jeff Gray. Handling crosscutting constraints in domain-specific modeling. In Aßmann et al. [91].
- [578] Jeff Gray. A Java-based approach for teaching principles of adaptive and evolvable software. *Science of Computer Programming*, 53(1):51–69, 2004.
- [579] Jeff Gray, Ted Bapty, and Sandeep Neema. Aspectifying constraints in model-integrated computing. In Tarr et al. [1362].
- [580] Jeff Gray, Ted Bapty, Sandeep Neema, and James Tuck. Handling crosscutting constraints in domain-specific modeling. *Comm. ACM*, 44(10):87–93, October 2001.
- [581] Jeff Gray, Yuehua Lin, and Jing Zhang. Levels of independence in aspect-oriented modeling. In *Workshop on Model-driven Approaches to Middleware Applications Development (Middleware)*, June 2003.
- [582] Jeff Gray and Suman Roychoudhury. A technique for constructing aspect weavers using a program transformation engine. In Lieberherr [884], pages 36–45.
- [583] Jeff Gray, Janos Sztipanovits, Douglas C. Schmidt, Ted Bapty, Sandeep Neema, and Aniruddha Gokhale. Two-level aspect weaving to support evolution in model-driven synthesis. In Filman et al. [513], pages 681–709.
- [584] Danny Greefhorst. Separating concerns in software logistics. In Tarr et al. [1362].
- [585] Robin Green and Awais Rashid. An aspect-oriented framework for schema evolution in object-oriented databases. In Coady et al. [327].
- [586] Philip Greenwood and Lynne Blair. Using dynamic aspect-oriented programming

- to implement an autonomic system. In Filman et al. [502], pages 76–88.
- [587] Philip Greenwood, Neil Loughran, Lynne Blair, and Awais Rashid. Dynamic framed aspects for dynamic software evolution. In Cazzola et al. [255], pages 101–110.
- [588] Michael Grier. Motivation for enabling separation of concerns in software product lines. In Ossher et al. [1091].
- [589] Wolfgang Grieskamp, Nicolas Kicillof, and Colin Campbell. Behavioral composition in symbolic domains. In Aldawud et al. [42].
- [590] Martin L. Griss. Implementing product-line features by composing aspects. In *1st Conf. Software Product Lines: Experience and Research Directions*, pages 271–288, Boston, 2000. Kluwer Academic Publishers.
- [591] Bill Griswold, Yoshikiyo Kato, and Jimmy Yuan. Aspect browser: Tool support for managing dispersed aspects. In Tarr et al. [1363].
- [592] William G. Griswold. Coping with cross-cutting software changes using information transparency. In Yonezawa and Matsuoka [1494], pages 250–265.
- [593] William G. Griswold, Yoshikiyo Kato, and Jimmy J. Yuan. Aspect browser: Tool support for managing dispersed aspects. In Ossher et al. [1091].
- [594] William G. Griswold, Jimmy J. Yuan, and Yoshikiyo Kato. Exploiting the map metaphor in a tool for software evolution. In *Proceedings of the 23rd international conference on Software engineering*, pages 265–274. IEEE Computer Society, 2001.
- [595] I. Groher and T. Baumgarth. Aspect-orientation from design to code. In Tekin-erdoĝan et al. [1378], pages 62–68.
- [596] Iris Groher. Managing variabilities with generative approaches. In Beuche et al. [164].
- [597] Iris Groher, Stephan Bleicher, and Christa Schwanninger. Designing features as pluggable collaborations. In Araújo et al. [79].
- [598] Iris Groher, Stephan Bleicher, and Christa Schwanninger. Model-driven development for pluggable collaborations. In Aldawud et al. [42].
- [599] Iris Groher and Stefan Schulze. Generating aspect code from UML models. In Aldawud et al. [44].
- [600] Iris Groher and Stefan Schulze. Generating aspect code from UML models. In Aldawud et al. [43].
- [601] W. Grosso. Aspect-oriented programming and AspectJ. *Dr. Dobbs Journal*, August 2002.
- [602] Paul Grunbacher, Alexander Egyed, and Nenad Medvidovic. Dimensions of concerns in requirements negotiation and architecture modeling. In Tarr et al. [1363].
- [603] J. Grundy. Supporting aspect-oriented component-based systems engineering. In *11th Int'l Conf. Software Engineering and Knowledge Engineering (SEKE)*, pages 388–395, 1999.
- [604] J. Grundy. Multi-perspective specification, design and implementation of software components using aspects. *Int'l Journal of Software Engineering and Knowledge Engineering*, 20(6), 2000.
- [605] John Grundy. Aspect-oriented requirements engineering for component-based software systems. In *4th IEEE International Symposium on Requirements Engineering*, pages 84–91. IEEE Computer Society, 1999.
- [606] John Grundy and Guoling Ding. Automatic validation of deployed J2EE components using aspects. In *The 17th IEEE Int'l Conf. Automated Software Engineering*, pages 47–56, September 2002.
- [607] John Grundy and John Hosking. Developing software components with aspects: Some issues and experiences. In Filman et al. [513], pages 585–604.
- [608] Stephan Gudmundson and Gregor Kiczales. Addressing practical software development issues in AspectJ with a pointcut interface. In Bergmans et al. [154].
- [609] Stephan Gudmundson and Gregor Kiczales. Data abstraction in AspectJ. In Yonezawa and Matsuoka [1494], pages 270–271.
- [610] Rachid Guerraoui. Strategic research directions in object-oriented programming. *ACM Computing Surveys*, 28(4):691–700, December 1996.

- [611] Gurcan Gulesir, Lodewijk Bergmans, Pascal Durr, and Istvan Nagy. Separating and managing dependent concerns. In Tourwé et al. [1399].
- [612] Javier J. Gutiérrez, Dario Villadiego, Maria J. Escalona, and Manuel Mejias. Aplicación de la programación orientada a aspectos en el diseño e implementación de pruebas funcionales. In Fuentes et al. [538].
- [613] K. Gybels. Using a logic language to express cross-cutting through dynamic joinpoints. In Costanza et al. [371]. Technical report IAI-TR-2002-1.
- [614] Kris Gybels and Johan Brichau. Arranging language features for pattern-based crosscuts. In Akşit [26], pages 60–69.
- [615] Kris Gybels, Maja D’Hondt, Istvan Nagy, and Remi Douence, editors. *2nd European Interactive Workshop on Aspects in Software (EIWAS’05)*, September 2005.
- [616] Kris Gybels, Stefan Hanenberg, Stephan Herrmann, and Jan Wloka, editors. *European Interactive Workshop on Aspects in Software (EIWAS)*, September 2004.
- [617] Kris Gybels and Andy Kellens. An experiment in using inductive logic programming to uncover pointcuts. In Gybels et al. [616].
- [618] Kris Gybels and Andy Kellens. Experiences with identifying aspects in Smalltalk using unique methods. In Tourwé et al. [1399].
- [619] Naji Habra. Separation of concerns in software engineering education. In Tarr and Ossher [1366].
- [620] Ouafa Hachani and Daniel Bardou. On aspect-oriented technology and object-oriented design patterns. In Hannemann et al. [640].
- [621] Szaboles Hajdara. An example of generating the synchronization code of a system composed by many similar objects. In Martín et al. [957].
- [622] Charles B. Haley, Robin C. Laney, and Bashar Nuseibeh. Deriving security requirements from crosscutting threat descriptions. In Lieberherr [884], pages 112–121.
- [623] Karl Matthias Hamel and Klaus Ostermann. Modularity and reusability of algorithms—a case study using Caesar. In Bergmans et al. [156].
- [624] Imed Hammouda, Mika Pussinen, Mika Katara, and Tommi Mikkonen. UML-based approach for documenting and specializing frameworks using patterns and concern architectures. In Aldawud et al. [44].
- [625] Haitham S. Hamza. Separation of concerns for evolving systems: a stability-driven approach. In Robillard [1212], pages 1–5.
- [626] Yan Han, Günter Kniesel, and Armin Cremers. Towards visual aspectj by a meta model and modeling notation. In Kandé et al. [737].
- [627] Yan Han, Günter Kniesel, and Armin B. Cremers. A meta model and modeling notation for AspectJ. In Aldawud et al. [38].
- [628] S. Hanenberg. A proposal for classifying tangled code. In Costanza et al. [371]. Technical report IAI-TR-2002-1.
- [629] S. Hanenberg, R. Hirschfeld, R. Unland, and K. Kawamura. Aspect weaving: Using the base language’s introspective facilities to determine join points. In Filman et al. [501].
- [630] Stefan Hanenberg, Boris Bachmendo, and Rainer Unland. An object model for general-purpose aspect languages. In Bosch [195], pages 80–91.
- [631] Stefan Hanenberg and Pascal Costanza. Connecting aspects in AspectJ: Strategies vs. patterns. In Coady et al. [327].
- [632] Stefan Hanenberg, Robert Hirschfeld, and Rainer Unland. Morphing aspects: incompletely woven aspects and continuous weaving. In Lieberherr [884], pages 46–55.
- [633] Stefan Hanenberg and Arno Schmidmeier. Idioms for building software frameworks in AspectJ. In Coady et al. [326].
- [634] Stefan Hanenberg and Rainer Unland. Grouping objects using aspect-oriented adapters. In Rashid [1181].
- [635] Stefan Hanenberg and Rainer Unland. Using and reusing aspects in AspectJ. In De Volder et al. [407].

- [636] Stefan Hanenberg and Rainer Unland. Roles and aspects: Similarities, differences, and synergetic potential. In *8th International Conference on Object-Oriented Information Systems*, September 2002.
- [637] Stefan Hanenberg and Rainer Unland. Specifying aspect-oriented design constraints in AspectJ. In Chu-Carroll et al. [286].
- [638] Stefan Hanenberg and Rainer Unland. Parametric introductions. In Akşit [26], pages 80–89.
- [639] J. Hannemann. The Aspect Mining Tool web site. <http://www.cs.ubc.ca/labs/spl/projects/amt.html>.
- [640] Jan Hannemann, Ruzanna Chitchyan, and Awais Rashid, editors. *Analysis of Aspect-Oriented Software (ECOOP 2003)*, July 2003.
- [641] Jan Hannemann and Gregor Kiczales. Overcoming the prevalent decomposition in legacy code. In Tarr and Ossher [1366].
- [642] Jan Hannemann and Gregor Kiczales. Design pattern implementation in Java and AspectJ. In *Proceedings of the 17th ACM conference on Object-oriented programming, systems, languages, and applications*, pages 161–173. ACM Press, 2002.
- [643] Jan Hannemann, Gail Murphy, and Gregor Kiczales. Role-based refactoring of crosscutting concerns. In Tarr [1361], pages 135–146.
- [644] Bruno Harbulot and John R. Gurd. Using AspectJ to separate concerns in parallel scientific Java code. In Lieberherr [884], pages 122–131.
- [645] Bruno Harbulot and John R. Gurd. A join point for loops in AspectJ. In Leavens et al. [867].
- [646] W. H. Harrison, H. Kilov, H. L. Ossher, and I. Simmonds. From dynamic super-types to subjects: A natural way to specify and develop systems. *IBM Systems Journal*, 35(2):244–256, June 1996.
- [647] William Harrison. Composition and multiple-inheritance in OO design (Where in the madness is the method?). In De Volder et al. [407].
- [648] William Harrison, Vincent Kruskal, Harold Ossher, Peri Tarr, and Frank Tip. Common low-level support for composition and weaving. In Chu-Carroll et al. [286].
- [649] William Harrison and Harold Ossher. Subject-oriented programming—A critique of pure objects. In *Proc. 1993 Conf. Object-Oriented Programming Systems, Languages, and Applications*, pages 411–428, September 1993.
- [650] William Harrison and Harold Ossher. Member-group relationships among objects. In Cytron and Leavens [379], pages 9–16.
- [651] William Harrison, Harold Ossher, Stanley Sutton, and Peri Tarr. Concern modeling in the concern manipulation environment. In Robillard [1212], pages 1–5.
- [652] William Harrison, Harold Ossher, and Peri Tarr. The beginnings of a graphical environment for subject-oriented programming. In Lopes et al. [918].
- [653] William Harrison, Harold Ossher, and Peri Tarr. Using delegation for software and subject composition. Technical Report RC 20946, IBM Thomas J. Watson Research Center, August 1997.
- [654] William Harrison, Harold Ossher, and Peri Tarr. Software engineering tools and environments: A roadmap. In *Conf. Future of Software Engineering*, pages 261–277. ACM, 2000.
- [655] William Harrison, Harold Ossher, and Peri Tarr. Asymmetrically vs. symmetrically organized paradigms for software composition. In Bergmans et al. [153].
- [656] William Harrison, Peri Tarr, and Harold Ossher. A position on considerations in UML design of aspects. In Aldawud et al. [37].
- [657] Alan Hartman, editor. *European Conference on Model Driven Architecture—Foundations and Applications*, November 2005.
- [658] Youssef Hassoun and Constantinos Constantinides. Visibility considerations and code reusability in AspectJ. In Bachmendo et al. [97].

- [659] F. J. Hauck, U. Becker, M. Geier, E. Meier, U. Rasthofer, and M. Steckermeier. AspectIX: A middleware for aspect-oriented programming. In Lopes et al. [914].
- [660] Michael Haupt, Matthew Webster, Siobhán Clarke, Andrew Jackson, and Awais Rashid. Report of current AOSD tools-based activity in the network. Technical Report AOSD-Europe-TUD-3, AOSD-Europe, March 2005.
- [661] Wilke Havinga, István Nagy, and Lodewijk Bergmans. Introduction and derivation of annotations in AOP: Applying expressive pointcut languages to introductions. In Gybels et al. [615].
- [662] Joni Helin and Pertti Kellomäki. Concern-based specification of distributed systems using behaviourally complete views. In Huang et al. [701].
- [663] Juan Hernandez, Michael Papathomas, Juan M. Murillo, and Fernando Sánchez. Coordinating concurrent objects: How to deal with the coordination aspect? In Lopes et al. [918].
- [664] José Luis Herrero, Fernando Sánchez, Fabiola Lucio, and Miguel Toro. Introducing separation of aspects at design time. In Lopes et al. [913].
- [665] José Luis Herrero, Fernando Sánchez, and Miguel Toro. Fault tolerance AOP approach. In Rashid [1181].
- [666] S. Herrmann, C. Hundt, and K. Mehner. Mapping use case level aspects to ObjectTeams/Java. In Moreira et al. [1017].
- [667] S. Herrmann and M. Mezini. Connectors for bridging mismatches between the components of a software engineering environment. *IEE Proceedings-Software*, 148(3):104–111, June 2001.
- [668] S. Herrmann, M. Mezini, and K. Ostermann. Joint efforts to dispel an approaching modularity crisis. In *Sixth International Workshop on Component-Oriented Programming (WCOP)*, 2001.
- [669] Stephan Herrmann. Dynamic view connectors for separating concerns in software engineering environments. In Tarr et al. [1363].
- [670] Stephan Herrmann. Composable designs with UFA. In Aldawud et al. [37].
- [671] Stephan Herrmann. Object teams: Improving modularity for crosscutting collaborations. In Akşit and Mezini [31].
- [672] Stephan Herrmann. Object confinement in object teams—Reconciling encapsulation and flexible integration. In Bachmendo et al. [97].
- [673] Stephan Herrmann, Christine Hundt, Katharina Mehner, and Jan Wloka. Using guard predicates for generalized control of aspect instantiation and activation. In Filman et al. [515], pages 93–101.
- [674] Stephan Herrmann and Mira Mezini. On the need for a unified MDSOC model: Experiences from constructing a modular software engineering environment. In Tarr et al. [1362].
- [675] Stephan Herrmann and Mira Mezini. PIROL: A case study for multidimensional separation of concerns in software engineering environments. In *OOPSLA*, pages 188–207, 2000.
- [676] Stephan Herrmann and Mira Mezini. Combining composition styles in the evolvable language LAC. In Tarr and Ossher [1366].
- [677] Stephan Herrmann and Jan Wloka. Seven steps from an idea to industrial adoption. In Bodkin et al. [185].
- [678] Dirk Heuzeroth, Welf Löwe, Andreas Ludwig, and Uwe Aßmann. Aspect-oriented configuration and adaptation of component communication. In Bosch [195], pages 58–69.
- [679] Patrick Hill, Simon Holland, and Robin C. Laney. Using dynamic aspects in music composition systems. In Filman et al. [502], pages 89–97.
- [680] Rich Hilliard. Aspects, concerns, subjects, views, . . . . In Ossher et al. [1091].
- [681] Erik Hilsdale and Jim Hugunin. Advice weaving in AspectJ. In Lieberherr [884], pages 26–35.
- [682] R. Hirschfeld. AspectS home page. <http://www.prakinf.tu-ilmeneau.de/~hirsch/Projects/Squeak/AspectS/>.

- [683] R. Hirschfeld. Advice activation in aspects. In Costanza et al. [371]. Technical report IAI-TR-2002-1.
- [684] R. Hirschfeld, K. Kawamura, and J. Eastman. Small, rich and adaptable. In Lopes et al. [929].
- [685] Robert Hirschfeld. Aspects - AOP with squeak. In De Volder et al. [407].
- [686] Robert Hirschfeld. Aspect-oriented programming with AspectS. In Akşit and Mezini [31].
- [687] Robert Hirschfeld and Pascal Costanza. Extending advice activation in aspects. In Gybels et al. [615].
- [688] Robert Hirschfeld, Ralf Lämmel, and Matthias Wagner. Design patterns and aspects—Modular designs with seamless run-time integration. In Bachmendo et al. [97].
- [689] Robert Hirschfeld, Kasper Østerbye, and Matthias Wagner. System integration using AOP. In Bachmendo et al. [97].
- [690] Robert Hirschfeld and Matthias Wagner. Metalevel tool support in AspectS. In Chu-Carroll et al. [286].
- [691] Robert Hirschfeld, Matthias Wagner, Wolfgang Kellerer, and Christian Prehofer. AOSD for system integration and personalization. In Bodkin et al. [185].
- [692] Wai Ming Ho, Jean-Marc Jézéquel, Francois Pennaneac’h, and Noël Plouzeau. A toolkit for weaving aspect oriented UML designs. In Kiczales [781], pages 99–105.
- [693] Wai-Ming Ho, Francois Pennaneac’h, Jean-Marc Jézéquel, and Noël Plouzeau. Aspect-oriented design with the UML. In Tarr et al. [1363].
- [694] Christian Hofmann, Robert Hirschfeld, and Jeff Eastman. Flexible call-by-call settlement — an opportunity for dynamic AOP. In Filman et al. [515], pages 19–25.
- [695] Ian M. Holland. Specifying reusable components using contracts. In *European Conf. on Object-Oriented Programming*, pages 287–308. Springer Verlag Lecture Notes 615, 1992.
- [696] Ian M. Holland. *The Design and Representation of Object-Oriented Components*. PhD thesis, Northeastern University, 1993.
- [697] D. Holmes, J. Noble, and J. Potter. Towards reusable synchronisation for object-oriented. In Lopes et al. [914].
- [698] David Holmes, James Noble, and John Potter. Aspects of synchronization. In Lopes et al. [918].
- [699] Pavel Hruby. Dimensions for the separation of concerns in describing software development processes. In Ossher et al. [1091].
- [700] Jie Huang. Experience using AspectJ to implementation cord. In Tarr et al. [1362].
- [701] Minhuan Huang, Hong Mei, and Jianjun Zhao, editors. *International Workshop on Aspect-Oriented Software Development (WAOSD 2004)*, September 2004.
- [702] Minhuan Huang, Chunlei Wang, and Lufeng Zhang. Toward a reusable and generic security aspect library. In De Win et al. [413].
- [703] Frank Hunleth, Ron Cytron, and Christopher Gill. Building customizable middleware using aspect oriented programming. In De Volder et al. [407].
- [704] Frank Hunleth and Ron K. Cytron. Footprint and feature management using aspect-oriented programming techniques. In *Proceedings of the Joint Conference on Languages, Compilers and Tools for Embedded Systems*, pages 38–45. ACM Press, 2002.
- [705] Walter L. Hürsch and Cristina Videira Lopes. Separation of concerns. Technical Report NU-CCS-95-03, College of Computer Science, Northeastern University, Boston, MA, February 1995.
- [706] IBM Research. Hyperspaces. <http://www.research.ibm.com/hyperspace/>.
- [707] Yuuji Ichisugi and Akira Tanaka. Difference-based modules: A class-independent module mechanism. In Magnusson [944], pages 62–88.
- [708] John Irwin, Jean-Marc Loingtier, John R. Gilbert, Gregor Kiczales, John Lamping, Anurag Mendhekar, and Tatiana Shpeisman. Aspect-oriented programming of

- sparse matrix code. In *Int'l Scientific Computing in Object-Oriented Parallel Environments (ISCOPE)*, volume 1343 of *LNCS*. Springer-Verlag, 1997.
- [709] Wes Isberg. Aop@work: Design with pointcuts to avoid pattern density. Technical report, IBM Developer Works, June 2005.
- [710] Takashi Ishio, Toshihiro Kamiya, Shinji Kusumoto, and Katsuro Inoue. Assertion with aspect. In Bergmans et al. [155].
- [711] Masanori Iwamoto and Jianjun Zhao. Refactoring aspect-oriented programs. In Aldawud et al. [44].
- [712] JAC project home page. <http://jac.aopsys.com>.
- [713] Andrew Jackson and Siobhan Clarke. Source-level cross-language aspect-oriented programming. In Karsai and Visser [745], pages 115–135.
- [714] Andrew Jackson and Siobhán Clarke. Towards a generic aspect oriented design process. In Aldawud et al. [42].
- [715] M. Jackson. Problems, subproblems and concerns. In Tekinerdoğan et al. [1378], pages 23–26.
- [716] H.-Arno Jacobsen. Middleware architecture design based on aspects, the open implementation metaphor and modularity. In Rashid [1181].
- [717] Ivar Jacobson. Language support for changeable large real time systems. In *Proceedings of the 1st ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications*, pages 377–384, September 1986.
- [718] Ivar Jacobson. The case for aspects, part 1. *Software Development*, pages 32–37, October 2003.
- [719] Ivar Jacobson. The case for aspects, part 2. *Software Development*, pages 44–48, November 2003.
- [720] Ivar Jacobson. Use cases and aspects—Working seamlessly together. *Journal of Object Technology*, 2(4), July 2003.
- [721] Ivar Jacobson and Pan-Wei Ng. *Aspect-Oriented Software Development with Use Cases*. Addison-Wesley, 2005.
- [722] Vladimir Jakobac, Nenad Medvidovic, and Alexander Egyed. Separating architectural concerns to ease program understanding. In Robillard [1212], pages 1–5.
- [723] A. G. J. Jansen, R. Smedinga, J. van Gurp, and J. Bosch. First class feature abstractions for product derivation. *IEEE Proceedings Software*, 151(4):187–197, August 2004.
- [724] Doug Janzen and Kris De Volder. Navigating and querying code without getting lost. In Akşit [26], pages 178–187.
- [725] Jean-Marc Jézéquel, Noël Plouzeau, Torben Weis, and Kurt Geihs. From contracts to aspects in UML designs. In Aldawud et al. [37].
- [726] JMangler homepage. <http://javalab.cs.uni-bonn.de/research/jmangler/>.
- [727] Michael B. Jones. Interposition agents: Transparently interposing user code at the system interface. In *Proceedings of the 14th ACM Symposium on Operating Systems Principles*, pages 80–93, December 1993.
- [728] Bo Nørregaard Jørgensen, Eddy Truyen, Frank Matthijs, and Wouter Joosen. Customization of object request brokers by application specific policies. In *Proc. Middleware'2000*, April 2000.
- [729] R. K. Joshi and N. Agrawal. AspectJ implementation of dynamically pluggable filter objects in distributed environment. In Costanza et al. [371]. Technical report IAI-TR-2002-1.
- [730] Susanne Jucknath. Software visualization and aspect-oriented software development. In Martín et al. [957].
- [731] Matthias Jung and Ernst W. Biersack. How layering protocol software violates separation of concerns. In Lopes et al. [913].
- [732] Brahmila Kamalakar and Sudipto Ghosh. A middleware transparent approach for developing CORBA-based distributed applications. In Aldawud et al. [38].

- [733] Piotr Kaminski. Applying multi-dimensional separation of concerns to software visualization. In Tarr and Ossher [1366].
- [734] Toshihiro Kamiya. SOMA: A paradigm to evolve software based on separation of concerns. In *Proceedings of the Workshop on Principles of Software Evolution*, pages 124–128. ACM Press, 2002.
- [735] Mohamed Kandé, Omar Aldawud, Grady Booch, and Bill Harrison, editors. *Second International Workshop on Aspect-Oriented Modeling with UML (<<UML>>2002)*, September 2002.
- [736] Mohamed Kande and Valentin Crettaz. Towards patterns for concern-oriented software architecture. In Aldawud et al. [44].
- [737] Mohamed Kandé, Dominik Stein, Omar Aldawud, Tzilla Elrad, Jeff Gray, and Jörg Kienzle, editors. *6th International Workshop on Aspect-Oriented Modeling*, March 2005.
- [738] Mohamed M. Kandé, Karl Lieberherr, and Peri Tarr. Special issue on modeling support for aspect-oriented software development. *International Journal on Software and System Modeling*, xxx.
- [739] Mohamed Mancona Kandé, Jörg Kienzle, and Alfred Strohmeier. From AOP to UML—A bottom-up approach. In Aldawud et al. [37].
- [740] Mohamed Mancona Kandé and Alfred Strohmeier. On the role of multi-dimensional separation of concerns in software architecture. In Tarr et al. [1362].
- [741] Mohamed Mancona Kandé and Alfred Strohmeier. Modeling crosscutting concerns using software connectors. In De Volder et al. [407].
- [742] Matthew Kaplan. Dynamic selection: The discriminating developer’s way to compose. In Tarr et al. [1362].
- [743] Matthew Kaplan, Harold Ossher, William Harrison, and Vincent Kruskal. Subject-oriented design and the Watson subject compiler. In *Proc. OOPSLA’96 Workshop on Subjectivity*, October 1996.
- [744] Murat Karaorman, Urs Hölzle, and John Bruno. jContractor: A reflective Java library to support design by contract. In P. Cointe, editor, *Meta-Level Architectures and Reflection, 2nd Int’l Conf. Reflection*, volume 1616 of *LNCS*, pages 175–196, Berlin, 1999. Springer Verlag.
- [745] Gabor Karsai and Eelco Visser, editors. *Proc. Generative Programming and Component Engineering: Third International Conference*, volume 3286 of *Springer-Verlag Lecture Notes in Computer Science*. Springer, October 2004.
- [746] E. P. Kasten, P. K. McKinley, S. M. Sadjadi, and R. E. K. Stirewalt. Separating introspection and intercession to support metamorphic distributed systems. In Akşit and Choukair [28].
- [747] M. Katara and T. Mikkonen. Aspect-oriented specification architectures for distributed real-time systems. In *7th Int’l Conf. Engineering of Complex Computer Systems (ICECCS)*, pages 180–190. IEEE, 2001.
- [748] Mika Katara. Superposing UML class diagram. In Aldawud et al. [37].
- [749] Mika Katara and Shmuel Katz. Architectural views of aspects. In Akşit [26], pages 1–10.
- [750] Mika Katara, Reino Kurki-Suonio, and Tommi Mikkonen. On the horizontal dimension of software architecture in formal specifications of reactive systems. In Clifton et al. [321], pages 37–44.
- [751] Mika Katara and Tommi Mikkonen. Refinements and aspects in UML. In Kandé et al. [735].
- [752] Shmuel Katz. A superimposition control construct for distributed systems. *ACM Transactions on Programming Languages and Systems*, 15(2):337–356, April 1993.
- [753] Shmuel Katz. Diagnosis of harmful aspects using regression verification. In Clifton et al. [321], pages 1–6.
- [754] Shmuel Katz and Yossi Gil. Aspects and superimpositions. In Lopes et al. [924].
- [755] Shmuel Katz and Awais Rashid. From aspectual requirements to proof obligations

- for aspect-oriented systems. In *Proc. 12th Int'l Requirements Engineering Conference*, pages 43–52. IEEE, September 2004.
- [756] Kazunori Kawauchi and Hidehiko Masuhara. Dataflow pointcut for integrity concerns. In De Win et al. [413].
- [757] Andy Kellens and Kris Gybels. Issues in performing and automating the “extract method calls” refactoring. In Bergmans et al. [156].
- [758] P. Kellomäki. Composing distributed systems from reusable aspects of behaviour. In Akşit and Choukair [28].
- [759] P. Kellomäki. Formal aspects for distributed systems. In Costanza et al. [371]. Technical report IAI-TR-2002-1.
- [760] Pertti Kellomäki. A formal basis for aspect-oriented specification with superposition. In Cytron and Leavens [379], pages 27–32.
- [761] Pertti Kellomäki and Tommi Mikkonen. Separating product variance and domain concepts in the specification of software product lines. In Lopes et al. [913].
- [762] Elizabeth A. Kendall. Agent roles and aspects. In Lopes et al. [914].
- [763] Elizabeth A. Kendall. Aspect-oriented programming for role models. In Lopes et al. [924].
- [764] Elizabeth A. Kendall. Role model designs and implementations with aspect-oriented programming. In *Proceedings of the 1999 ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications*, pages 353–369. ACM Press, 1999.
- [765] Liz Kendall. Reengineering for separation of concerns. In Tarr et al. [1363].
- [766] P. Kenens, S. Michiels, F. Matthijs, B. Robben, E. Truyen, B. Vanhaute, W. Joosen, and P. Verbaeten. An AOP case with static and dynamic aspects. In Lopes et al. [914].
- [767] Mik Kersten. AO tools: State of the (AspectJ) art and open problems. In Chu-Carroll et al. [286].
- [768] Mik Kersten. Tool requirements for commercial development with AspectJ. In Bodkin et al. [185].
- [769] Mik Kersten and Gail Murphy. Mylar: a degree-of-interest model for IDEs. In Tarr [1361], pages 159–168.
- [770] Mik A. Kersten and Gail C. Murphy. Atlas: A case study in building a web-based learning environment using aspect-oriented programming. In *Proc. ACM Conf. Object-oriented Programming, Systems, Languages, and Applications*, pages 340–352. ACM, 1999.
- [771] Emre Kiciman and Armando Fox. Separation of concerns in networked service composition. In Tarr and Ossher [1366].
- [772] G. Kiczales. Towards a new model of abstraction in software engineering. In *Int'l Workshop on New Models in Software Architecture, Reflection and Meta-Level Architecture*, 1992.
- [773] G. Kiczales. Aspect-oriented programming. *ACM Computing Surveys*, 28(4es):154, 1996.
- [774] G. Kiczales. Beyond the black box: Open implementation. *IEEE Software*, 13(1):8–11, January 1996.
- [775] G. Kiczales. Aspect-oriented programming - the fun has just begun. In *Workshop on New Visions for Software Design and Productivity: Research and Applications*, December 2001.
- [776] G. Kiczales, J. des Rivieres, and D. G. Bobrow. *The Art of the Metaobject Protocol*. MIT Press, Cambridge, Massachusetts, 1991.
- [777] G. Kiczales, E. Hilsdale, J. Hugunin, M. Kersten, J. Palm, and W. G. Griswold. Getting started with AspectJ. *Comm. ACM*, 44(10):59–65, October 2001.
- [778] G. Kiczales, E. Hilsdale, J. Hugunin, M. Kersten, J. Palm, and W. G. Griswold. An overview of AspectJ. In J. L. Knudsen, editor, *Proc. ECOOP 2001, LNCS 2072*, pages 327–353, Berlin, June 2001. Springer-Verlag.

- [779] G. Kiczales, J. Lamping, A. Mendhekar, C. Maeda, C. Lopes, J.-M. Loingtier, and J. Irwin. Aspect-oriented programming. Technical Report SPL97-008 P9710042, Xerox PARC, February 1997.
- [780] G. Kiczales and C. V. Lopes. ECOOP '98 aspect-oriented programming tutorial notes. Technical report, AITO, July 1998.
- [781] Gregor Kiczales, editor. *Proc. 1st Int' Conf. on Aspect-Oriented Software Development (AOSD-2002)*. ACM Press, April 2002.
- [782] Gregor Kiczales. The key to AOP. *Software Development*, December 2003.
- [783] Gregor Kiczales. Testing the waters. *Software Development*, November 2003.
- [784] Gregor Kiczales. The AOP report card. *Software Development*, January 2004.
- [785] Gregor Kiczales. Common misconceptions. *Software Development*, March 2004.
- [786] Gregor Kiczales. It's the crosscutting. *Software Development*, February 2004.
- [787] Gregor Kiczales. A little goes a long way. *Software Development*, May 2004.
- [788] Gregor Kiczales. The more the merrier. *Software Development*, October 2004.
- [789] Gregor Kiczales. The new modularity. *Software Development*, September 2004.
- [790] Gregor Kiczales. Palpable progress. *Software Development*, June 2004.
- [791] Gregor Kiczales. Show me the structure. *Software Development*, April 2004.
- [792] Gregor Kiczales. Thinking aspects. *Software Development*, August 2004.
- [793] Gregor Kiczales. Aspect-oriented programming (tutorial). In *ICSE '05: Proceedings of the 27th international conference on Software engineering*, pages 730–730, New York, 2005. ACM Press.
- [794] Gregor Kiczales and Erik Hilsdale. Aspect-oriented programming. In *Proceedings of the 8th European Software Engineering Conference held jointly with 9th Acm Sigsoft Symposium on Foundations of Software Engineering*, page 313. ACM Press, 2001.
- [795] Gregor Kiczales, Jim Hugunin, Mik Kirsten, John Lamping, Cristina Lopes, and William G. Griswold. Semantics-based crosscutting in AspectJ. In Tarr et al. [1363].
- [796] Gregor Kiczales, John Lamping, Anurag Mendhekar, Chris Maeda, Cristina Lopes, Jean-Marc Loingtier, and John Irwin. Aspect-oriented programming. In Mehmet Akşit and Satoshi Matsuoka, editors, *11th European Conf. Object-Oriented Programming*, volume 1241 of *LNCS*, pages 220–242. Springer Verlag, 1997.
- [797] Gregor Kiczales and Mira Mezini. Aspect-oriented programming and modular reasoning. In *ICSE '05: Proceedings of the 27th international conference on Software engineering*, pages 49–58, New York, 2005. ACM Press.
- [798] Jörg Kienzle and Rachid Guerraoui. Aspect oriented software development AOP: Does it make sense? The case of concurrency and failures. In Magnusson [944], pages 37–61.
- [799] Jörg Kienzle, Yang Yu, and Jie Xiong. On composition and reuse of aspects. In Bergmans et al. [153].
- [800] Jörg Kienzle, Yang Yu, and Jie Xiong. On composition and reuse of aspects. In Leavens and Clifton [866].
- [801] Howard Kim and Siobhán Clarke. The relevance of AOP to an applications programmer in an EJB environment. In Coady et al. [327].
- [802] Doug Kimmelman. Multidimensional tree-structured spaces for separation of concerns in software development environments. In Ossher et al. [1091].
- [803] Mik Kirsten. Aop@work: Aop tools comparison, part 1: Language mechanisms. Technical report, IBM Developer Works, February 2005.
- [804] Mik Kirsten. Aop@work: Aop tools comparison, part 2: Development environments. Technical report, IBM Developer Works, February 2005.
- [805] Ivan Kisely. *Aspect-Oriented Programming with AspectJ*. Sams, 2002.

- [806] T. Kishi and N. Noda. Aspect-oriented context modeling for embedded systems. In Tekinerdoğan et al. [1378], pages 68–74.
- [807] Tomoji Kishi and Natsuko Noda. Aspect-oriented analysis for product line architecture. In Tarr et al. [1362].
- [808] Tomoji Kishi and Natsuko Noda. Analyzing concerns used in analysis/design techniques. In Tarr and Ossher [1366].
- [809] H. Klaeren, E. Pulvermueller, A. Rashid, and A. Speck. Aspect composition applying the design by contract principle. In *2nd Int'l Symp. Generative and Component-based Software Engineering (GCSE)*, volume 2177 of *lncs*, pages 57–69, Berlin, 2000. Springer-Verlag.
- [810] Jacques Klein and Jean-Marc Jézéquel. Problems of the semantic-based weaving of scenarios. In Araújo et al. [79].
- [811] Paul Klint and Tijs van der Storm. Reflections on feature oriented software engineering. In Beuche et al. [164].
- [812] Karl Klose and Klaus Ostermann. Back to the future: Pointcuts as predicates over traces. In Leavens et al. [867].
- [813] Günter Kniesel and Michael Austermann. CC4J - code coverage for Java - a load-time adaptation success story. In *Component Deployment - IFIP/Working Conference (CD2002)*, volume 2370 of *lncs*, pages 155–169, Berlin, June 2002. Springer-Verlag.
- [814] Günter Kniesel and Uwe Bardey. Static dependency analysis for conditional program transformations. Technical report, CS Dept. III, University of Bonn, July 2003.
- [815] Günter Kniesel, Pascal Costanza, and Michael Austermann. JMangler—A framework for load-time transformation of Java class files. In *First IEEE Int'l Workshop on Source Code Analysis and Manipulation (SCAM 2001)*, November 2001.
- [816] Günter Kniesel, Pascal Costanza, and Michael Austermann. JMangler—A powerful back-end for aspect-oriented programming. In Filman et al. [513], pages 311–342.
- [817] Günter Kniesel and Tobias Rho. Generic aspect languages - needs, options and challenges. In Seinturier [1265].
- [818] Günter Kniesel, Tobias Rho, and Stefan Hakenberg. Evolvable pattern implementations need generic aspects. In Cazzola et al. [255], pages 111–126.
- [819] Jorgen Knudsen. Aspect-oriented programming in BETA using the fragment system. In Lopes et al. [924].
- [820] Sergei Kojarski, Karl Lieberherr, David H. Lorenz, and Robert Hirschfeld. Aspectual reflection. In Bergmans et al. [153].
- [821] Sergei Kojarski and David H. Lorenz. Plug-gable AOP: Designing aspect mechanisms for third-party composition. In *OOPSLA '05: Proceedings of the 20th annual ACM SIGPLAN conference on Object oriented programming systems languages and applications*, pages 247–263. ACM Press, 2005.
- [822] Sergei Kojarski and David H. Lorenz. A top-down model of an aop weaving process. In Bergmans et al. [156].
- [823] P. Koopmans. On the design and realization of the sina compiler. Master's thesis, University of Twente, 1995.
- [824] P. Koopmans. Sina user's guide and reference manual. Technical report, Dept. of Computer Science, University of Twente, 1995.
- [825] Christian Koppen and Maximilian Störzer. PCDiff: Attacking the fragile pointcut problem. In Gybels et al. [616].
- [826] Jan Kort and Ralf Lämmel. Separating representation concerns in re-engineering applications of rewriting. In Bachmendo et al. [97].
- [827] Jens Krinke and Silvia Breu. Control-flow-graph-based aspect mining. In Tourwé et al. [1398].
- [828] Arvind S. Krishna, Emre Turkaye, Cemal Yilmaz, Douglas C. Schmidt, Aniruddha Gokhale, Atif Memon, and Adam Porter. Model-driven software tools for configuring and customizing middleware for distributed real-time and embedded systems. In Beuche et al. [164].
- [829] Shriram Krishnamurthi, Kathi Fisler, and Michael Greenberg. Verifying aspect advice modularly. In FSE04 [535], pages 137–146.

- [830] Vincent Kruskal. A blast from the past: Using P-EDIT for multidimensional editing. In Tarr et al. [1363].
- [831] U. Kulesza, A. Garcia, F. Bleasby, and C. Lucena. Instantiating and customizing product line architectures using aspects and crosscutting feature models. In Baniassad et al. [109].
- [832] U. Kulesza, A. Garcia, and C. Lucena. Generating aspect-oriented agent architectures. In Tekinerdoğan et al. [1378], pages 42–49.
- [833] U. Kulesza and A. Garcia C. Lucena. Towards a method for the development of aspect-oriented generative approaches. In Moreira et al. [1017].
- [834] Uirá Kulesza and Dilma M. Silva. Reengineering of the JAWS web server design using aspect-oriented programming. In Lopes et al. [913].
- [835] Mario Kupries and Erika Horn. An architecture-based approach substantiating the aspect of interagent connection in platforms. In Choukair [280], pages 127–132.
- [836] Ivan Kurtev. Transformation of XML concerns into applications using AOP-techniques. In Aßmann et al. [91].
- [837] Ramnivas Laddad. I want my AOP! *Java-World magazine*, January 2002.
- [838] Ramnivas Laddad. Aspect-oriented refactoring part 1: Overview and process. Technical report, TheServerSide.com, 2003.
- [839] Ramnivas Laddad. *AspectJ in Action: Practical Aspect-Oriented Programming*. Manning, 2003.
- [840] Ramnivas Laddad. Aop@work: Aop and metadata: A perfect match, part 1—concepts and constructs of metadata-fortified aop. Technical report, IBM Developer Works, March 2005.
- [841] Ramnivas Laddad. Aop@work: Aop and metadata: A perfect match, part 2—multidimensional interfaces with metadata. Technical report, IBM Developer Works, April 2005.
- [842] Robert Laddaga, Paul Robertson, and Howie Shrobe. Aspects of the real-world. In De Volder et al. [407].
- [843] Ralf Laemmel, Eelco Visser, and Joost Visser. Strategic programming meets adaptive programming. In Aksit [26], pages 168–177.
- [844] S. Laemmermann and E. Tyugu. A specification logic for dynamic composition of services. In Choukair [280], pages 157–162.
- [845] Donal Lafferty. Avoiding incorrect and unpredictable behaviour with attribute-based crosscutting. In Coady et al. [328].
- [846] Donal Lafferty and Vinny Cahill. Real world evaluation of Aspect-Oriented Programming with Iguana. In Lopes et al. [913].
- [847] Donal Lafferty and Vinny Cahill. Language-independent aspect-oriented programming. In *Proceedings of the 18th ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications*, pages 1–12. ACM Press, 2003.
- [848] Bert Lagaisse, Wouter Joosen, and Bart De Win. Managing semantic interference with aspect integration contracts. In Bergmans et al. [155].
- [849] Albert Lai and Gail C. Murphy. The structure of features in Java code: An exploratory investigation. In Ossher et al. [1091].
- [850] Albert Lai and Gail C. Murphy. Capturing concerns with conceptual modules. In Tarr and Ossher [1366].
- [851] Albert Lai, Gail C. Murphy, and Robert J. Walker. Separating concerns with HyperJ: An experience report. In Tarr et al. [1363].
- [852] Patrick Lam, Viktor Kuncak, and Martin Rinard. Crosscutting techniques in program specification and analysis. In Tarr [1361], pages 169–180.
- [853] Ralf Lämmel. Declarative aspect-oriented programming. In *ACM SIGPLAN Workshop on Partial Evaluation and Semantics-Based Program Manipulation (PEPM 99)*, 1999.
- [854] Ralf Lämmel. A semantical approach to method-call interception. In Kiczales [781], pages 41–55.

- [855] Ralf Lämmel. Adding superimposition to a language semantics. In Leavens and Clifton [866].
- [856] Ralf Lämmel, Günter Riedewald, and Wolfgang Lohmann. Adaptation of functional object programs. In Lopes et al. [924].
- [857] Ralf Lämmel and Kris De Schutter. What does aspect-oriented programming mean to Cobol? In Tarr [1361], pages 99–110.
- [858] John Lamping. The interaction of components and aspects. In Lopes et al. [918].
- [859] John Lamping. The role of base in aspect-oriented programming. In Lopes et al. [924].
- [860] John Lamping. The role of the base in aspect-oriented programming. In Ossher et al. [1091].
- [861] Robin Laney, Janet Van der Linden, and Pete Thomas. Evolution of aspects for legacy system security concerns. In De Win et al. [413].
- [862] David Larochelle, Karl Scheidt, and Kevin Sullivan. Join point encapsulation. In Bergmans et al. [153].
- [863] Kung-Kiu Lau and Vladyslav Ukis. A component model for separation of control flow from computation in component-based systems. In Aksit et al. [35].
- [864] Julia L. Lawall, Gilles Muller, and Richard Urunuela. Tarantula: Killing driver bugs before they hatch. In Coady et al. [328].
- [865] Andrew Le Gear, Brendan Cleary, Jim Buckley, J.J. Collins, and Chris Exton. Making a reuse aspectual view explicit in existing software. In Tourwé et al. [1399].
- [866] Gary T. Leavens and Curtis Clifton, editors. *FOAL: Foundations of Aspect-Oriented Languages*, March 2003.
- [867] Gary T. Leavens, Curtis Clifton, and Ralf Lämmel, editors. *Foundations of Aspect-Oriented Languages*, March 2005.
- [868] Diana Lee and Robert Filman. Verification of compositional software architectures. In Linden and Thompson [901].
- [869] Fabrice Legond-Aubry, Gerard Florin, and Lionel Seinturier. An AOP layer to abstract programming with distributed components. In Huang et al. [701].
- [870] Otávio Lemos, José Carlos Maldonado, and Paulo Cesar Masiero. Data flow integration testing criteria for aspect-oriented programs. In Garcia et al. [549].
- [871] László Lengyel, Tihamér Levendovszky, and Hassan Charaf. Eliminating crosscutting constraints from visual model transformation rules. In Aldawud et al. [42].
- [872] W. Leong. The Aspect Browser web site. <http://www.cs.ucsd.edu/users/wgg/Software/AB>.
- [873] W.-Y. Leong. Using the atlas metaphor to assist cross-cutting software changes. Master's thesis, University of California, San Diego, 2002.
- [874] H. Leroux, C. Mingins, and A. Requile-Romanczuk. JACOT: A UML-based tool for the runtime-inspection of concurrent Java programs. In Filman et al. [501].
- [875] N. Lesiecki. Test flexibility with AspectJ and mock objects. Technical report, Java Technology Zone for IBM's Developer Works, May 2002.
- [876] Nicholas Lesiecki. Improve modularity with aspect-oriented programming: Aspectj brings aop to the java language. Technical report, IBM Developer Works, 2002.
- [877] Nicholas Lesiecki. Test flexibly with aspectj and mock objects: Enhance unit testing with test-only behavior. Technical report, IBM Developer Works, May 2002.
- [878] Nicholas Lesiecki. Aop@work: Enhance design patterns with aspectj, part 1. aop makes patterns lighter, more flexible, and easier to (re)use. Technical report, IBM Developer Works, May 2005.
- [879] Nicholas Lesiecki. Unit test your aspects. Technical report, IBM Developer Works, November 2005.
- [880] Wu-Hon F. Leung. Untangle with nonprocedural programming. In Bergmans et al. [156].

- [881] B. Lewis. Recording events to analyze programs. In Filman et al. [501].
- [882] Harry C. Li, Shriram Krishnamurthi, and Kathi Fisler. Interfaces for modular feature verification. In *Proc. 17th IEEE Int'l Conf. on Automated Software Engineering*, pages 195–204, September 2002.
- [883] Jingyue Li, Siv Hilde Houmb, and Axel Anders Kvale. A process to combine AOM and AOP: A proposal based on a case study. In Aldawud et al. [38].
- [884] Karl Lieberherr, editor. *Proc. 3rd Int' Conf. on Aspect-Oriented Software Development (AOSD-2004)*. ACM Press, March 2004.
- [885] Karl Lieberherr, David Lorenz, and Mira Mezini. Programming with Aspectual Components. Technical Report NU-CCS-99-01, College of Computer Science, Northeastern University, Boston, MA, March 1999.
- [886] Karl Lieberherr and David H. Lorenz. Coupling aspect-oriented and adaptive programming. In Filman et al. [513], pages 145–164.
- [887] Karl Lieberherr, David H. Lorenz, and Johan Ovlinger. Aspectual collaborations: Combining modules and aspects. *The Computer Journal*, 46(5):542–565, September 2003.
- [888] Karl Lieberherr, David H. Lorenz, and Pengcheng Wu. A case for statically executable advice: Checking the law of Demeter with AspectJ. In Akşit [26], pages 40–49.
- [889] Karl Lieberherr, Doug Orleans, and Johan Ovlinger. Aspect-oriented programming with adaptive methods. *Comm. ACM*, 44(10):39–41, October 2001.
- [890] Karl Lieberherr, Boaz Pat-Shamir, and Doug Orleans. Traversals of object structures: Specification and efficient implementation. *ACM Transactions on Programming Languages and Systems*, 26(2):370–412, March 2004.
- [891] Karl J. Lieberherr. Object-oriented programming with class dictionaries. *Journal on Lisp and Symbolic Computation*, 1(2):185–212, 1988.
- [892] Karl J. Lieberherr. Component enhancement: An adaptive reusability mechanism for groups of collaborating classes. In J. van Leeuwen, editor, *Information Processing '92, 12th World Computer Congress*, pages 179–185. Elsevier, 1992.
- [893] Karl J. Lieberherr. *Adaptive Object-Oriented Software: the Demeter Method with Propagation Patterns*. PWS Publishing Company, Boston, 1996.
- [894] Karl J. Lieberherr and Ian Holland. Assuring good style for object-oriented programs. *IEEE Software*, pages 38–48, September 1989.
- [895] Karl J. Lieberherr, Walter Hürsch, Ignacio Silva-Lepe, and Cun Xiao. Experience with a graph-based propagation pattern programming tool. In Gene Forte and Nazim Madhavji, editors, *Int'l Workshop on CASE*, pages 114–119, Montréal, Canada, 1992. IEEE Computer Society.
- [896] Karl J. Lieberherr and Doug Orleans. Preventive program maintenance in Demeter/Java (research demonstration). In *Int'l Conf. Software Engineering*, pages 604–605, Boston, MA, 1997. ACM Press.
- [897] Karl J. Lieberherr, Doug Orleans, and Johan Ovlinger. Aspect-oriented programming with adaptive methods. Technical Report NU-CCS-2001-01, College of Computer Science, Northeastern University, Boston, February 2001.
- [898] Karl J. Lieberherr, Jeffrey Palm, and Ravi Sundaram. Expressiveness and complexity of crosscut languages. In Leavens et al. [867].
- [899] Karl J. Lieberherr, Ignacio Silva-Lepe, and Cun Xiao. Adaptive object-oriented programming using graph-based customization. *Comm. ACM*, 37(5):94–101, May 1994.
- [900] Karl J. Lieberherr and Cun Xiao. Object-oriented software evolution. *IEEE Trans. Soft. Eng.*, 19(4):313–343, April 1993.
- [901] Ted Linden and Craig Thompson, editors. *OMG-DARPA Workshop on Compositional Software Architectures*, January 1998.
- [902] Jean Marie Lions, Didier Simoneau, Gilles Pitette, and Imed Moussa. Extending OpenTool/UML using metamodeling: An aspect-oriented programming case study. In Kandé et al. [735].

- [903] Martin Lippert and Cristina Videira Lopes. A study on exception detecton and handling using aspect-oriented programming. In *Proceedings of the 22nd International Conference on Software Engineering*, pages 418–427. ACM Press, 2000.
- [904] Chunjian Robin Liu, Celina Gibbs, and Yvonne Coady. SONAR: System optimization and navigation with aspects at runtime. In Filman et al. [515], pages 57–62.
- [905] Jing (Janet) Liu, Robyn R. Lutz, and Jeffrey M. Thompson. Mapping concern space to software architecture: a connector-based approach. In Robillard [1212], pages 1–5.
- [906] R. Liu and Y. Coady. Modularization of Jini services in pervasive systems: Conventional bottle versus contemporary aspect. In Lopes et al. [929].
- [907] Cidiane Lobato, Alessandro Garcia, Alexander Romanovsky, Cláudio Sant’Anna, and Uirá Kulesza. Mobility as an aspect: The AspectM framework. In Garcia et al. [549].
- [908] D. Lohmann and J. Ebert. A generalization of the Hyperspace approach using meta-models. In Araújo et al. [81].
- [909] Daniel Lohmann, Georg Blaschke, and Olaf Spinczyk. Generic advice: On the combination of AOP with generative programming in AspectC++. In Karsai and Visser [745], pages 55–74.
- [910] Daniel Lohmann, Wasif Gilani, and Olaf Spinczyk. On adaptable aspect-oriented operating systems. In Olaf Spinczyk, Andreas Gal, and Michael Schoettner, editors, *ECOOP Workshop on Programming Languages and Operating Systems*, June 2004.
- [911] Daniel Lohmann and Olaf Spinczyk. On typesafe aspect implementations in C++. In Gybels et al. [616].
- [912] Daniel Lohmann, Olaf Spinczyk, and Wolfgang Schröder-Preikschat. On the configuration of non-functional properties in operating system product lines. In Coady et al. [328].
- [913] C. Lopes, L. Bergmans, M. D’Hondt, and P. Tarr, editors. *Workshop on Aspects and Dimensions of Concerns (ECOOP 2000)*, June 2000.
- [914] Crista Lopes, Gregor Kiczales, Bedir Tekinerdoğan, Wolfgang De Meuter, and Marco Meijers, editors. *Workshop on Aspect Oriented Programming (ECOOP 1998)*, June 1998.
- [915] Crista Videira Lopes and Gregor Kiczales. Recent developments in AspectJ. In Lopes et al. [914].
- [916] Cristina Lopes, Paul Dourish, David Lorenz, and Karl Lieberherr. Beyond AOP: Toward naturalistic programming. In *Onward! in the 18th ACM conference on Object-oriented programming, systems, languages, and applications*, 2003.
- [917] Cristina Lopes, Erik Hilsdale, Jim Hugunin, Mik Kersten, and Gregor Kiczales. Illustrations of crosscutting. In Lopes et al. [913].
- [918] Cristina Lopes, Gregor Kiczales, Bedir Tekinerdoğan, and Kim Mens, editors. *Workshop on Aspect Oriented Programming (ECOOP 1997)*, June 1997.
- [919] Cristina Videira Lopes. Graph-based optimizations for parameter passing in remote invocations. In Luis-Felipe Cabrera and Marvin Theimer, editors, *4th Int’l Workshop on Object Orientation in Operating Systems*, pages 179–182. IEEE Computer Society Press, August 1995.
- [920] Cristina Videira Lopes. Adaptive parameter passing. In *2nd Int’l Symposium on Object Technologies for Advanced Software*, pages 118–136. Springer-Verlag, March 1996.
- [921] Cristina Videira Lopes. *D: A Language Framework for Distributed Programming*. PhD thesis, College of Computer Science, Northeastern University, 1997.
- [922] Cristina Videira Lopes. AOP: A historical perspective (What’s in a name?). In Filman et al. [513], pages 97–122.
- [923] Cristina Videira Lopes and Sushil Krishna Bajracharya. An analysis of modularity in aspect oriented design. In Tarr [1361], pages 15–26.
- [924] Cristina Videira Lopes, Andrew Black, Liz Kendall, and Lodewijk Bergmans, editors. *Int’l Workshop on Aspect-Oriented Programming (ECOOP 1999)*, June 1999.

- [925] Cristina Videira Lopes and Gregor Kiczales. D: A language framework for distributed programming. Technical Report SPL-97-010, Palo Alto Research Center, 1997.
- [926] Cristina Videira Lopes, Gregor Kiczales, and Gail Murphy, editors. *Int'l Workshop on Aspect Oriented Programming (ICSE 1998)*, April 1998.
- [927] Cristina Videira Lopes and Karl Lieberherr. AP/S++: Case-study of a MOP for purposes of software evolution. In G. Kiczales, editor, *Reflection '96*, April 1996.
- [928] Cristina Videira Lopes and Karl J. Lieberherr. Abstracting process-to-function relations in concurrent object-oriented applications. In Tokoro and Pareschi [1393], pages 81–99.
- [929] Cristina Videira Lopes, Steffen Schaefer, Siobhán Clarke, Anind Dey, Tzilla Elrad, Robert E. Filman, Jens Jahnke, and Tony Willis, editors. *Building Software for Pervasive Computing at OOPSLA '04*, October 2004.
- [930] Roberto E. Lopez-Herrejon and Don Batory. Improving incremental development in aspectj by bounding quantification. In Bergmans et al. [156].
- [931] D. H. Lorenz. Visitor beans: An aspect-oriented pattern. In Lopes et al. [914].
- [932] David H. Lorenz and Yvonne Coady, editors. *ACP4IS: Aspects, Components, and Patterns for Infrastructure Software*, March 2004.
- [933] Nicolas Lorient, Marc Ségura-Devillechaise, and Jean-Marc Menaud. Software security patches: Audit, deployment and hot update. In Coady et al. [328].
- [934] N. Loughran and A. Rashid. Mining aspects. In Rashid et al. [1191].
- [935] Neil Loughran, Nikos Parlavantzas, Monica Pinto, Lidia Fuentes Fernández, Pablo Sánchez, Matthew Webster, and Adrian Colyer. Survey of aspect-oriented middleware. Technical Report AOSD-Europe-ULANC-10, AOSD-Europe, June 2005.
- [936] Neil Loughran and Awais Rashid. Relational database support for aspect-oriented programming. In Aksit and Mezini [31].
- [937] Neil Loughran and Awais Rashid. Managing variability throughout the software development lifecycle. In Beuche et al. [164].
- [938] Neil Loughran, Awais Rashid, Weishan Zhang, and Stan Jarzabek. Supporting product line evolution with framed aspects. In Lorenz and Coady [932].
- [939] Tina Low. Designing, modelling and implementing a toolkit for aspect-oriented tracing (TAST). In Aldawud et al. [37].
- [940] Welf Löwe and Mario Südholt, editors. *5th International Workshop on Software Composition*, March 2006.
- [941] Welf Löwe and Mario Südholt, editors. *5th International Workshop on Software Composition (ETAPS 2006)*, March 2006.
- [942] Markus Lumpe, Franz Achermann, and Oscar Nierstrasz. A formal language for composition. In Gary T. Leavens and Murali Sitaraman, editors, *Foundations of component-based systems*, pages 69–90. Cambridge University Press, Cambridge, UK, 2000.
- [943] C. P. Lunau. Is composition of metaobjects = aspect oriented programming. In Lopes et al. [914].
- [944] B. Magnusson, editor. *ECOOP 2002—Object-Oriented Programming: 16th European Conference, LNCS 2374*. Springer Verlag, June 2002.
- [945] Tom Mahieu, Bart Vanhaute, Karel De Vlamincq, Gerda Janssens, and Wouter Joosen. Using AOP to build complex data centric component frameworks. In Tarr et al. [1362].
- [946] Mark Mahoney, Atef Bader, Tzilla Elrad, and Omar Aldawud. Using aspects to abstract and modularize statecharts. In Aldawud et al. [38].
- [947] Mark Mahoney and Tzilla Elrad. Modeling platform specific attributes of a system as crosscutting concerns using aspect-oriented statecharts and virtual finite state machines. In Kandé et al. [737].

- [948] Mark Mahoney and Tzilla Elrad. Weaving crosscutting concerns into live sequence charts using the play engine. In Aldawud et al. [42].
- [949] Daniel Mahrenholz, Olaf Spinczyk, and Wolfgang Schröder-Preikschat. Program instrumentation for debugging and monitoring with AspectC++. In *5th Int'l Symp. Object-oriented Real-time Distributed Computing*, pages 249–256. IEEE, April 2002.
- [950] Jacques Malenfant, Maria-Teresa Segarra, and Françoise André. Dynamic adaptability: The MoléNE experiment. In Yonezawa and Matsuoka [1494], pages 110–117.
- [951] Frank Manola. Providing systemic properties (ilities) and quality of service in component-based systems. Technical report, OBJS, 1999.
- [952] Jeremy Manson, Jan Vitek, and Suresh Jaganathan. Dynamic aspects for runtime fault determination and recovery. In Filman et al. [515], pages 27–32.
- [953] Xiaoguang Mao and John May. A framework of integration testing using AspectJ. In Huang et al. [701].
- [954] Marius Marin. Language support for refactoring to aspects. In Tourwé et al. [1398].
- [955] Marius Marin, Leon Moonen, and Arie van Deursen. An approach to aspect refactoring based on crosscutting concern types. In Robillard [1212], pages 1–5.
- [956] Joshua Marshall, Doug Orleans, and Karl J. Lieberherr. DJ: Dynamic structure-shy traversal in pure Java. Technical report, Northeastern University, May 1999.
- [957] Pedro José Clemente Martín, Sergio Lujan Mora, Miguel Ángel Pérez Toledano, Hans Resiser, and Sergio Soares, editors. *The 13th Workshop for PhD Students in Object-Oriented Systems (ECOOP 2003)*, July 2003.
- [958] T. Massoni, V. Alves, S. Soares, and P. Borba. PDC: Persistent data collections pattern. In *First Latin American Conference on Pattern Languages of Programming — SugarLoafPLOP*, October 2001.
- [959] Tiago Massoni, Augusto Sampaio, and Paulo Borba. Progressive implementation of aspects. In De Volder et al. [407].
- [960] H. Masuhara and G. Kiczales. Modular crosscutting in aspect-oriented mechanisms. In Luca Cardelli, editor, *ECOOP 2003—Object-Oriented Programming, 17th European Conference*, volume 2743 of *lncs*, pages 2–28, Berlin, July 2003. Springer-Verlag.
- [961] Hidehiko Masuhara and Gregor Kiczales. A modeling framework for aspect-oriented mechanisms; draft. <http://www.cs.ubc.ca/~gregor/masuhara-aop-framework.pdf>.
- [962] Hidehiko Masuhara, Gregor Kiczales, and Chris Dutchyn. Compilation semantics of aspect-oriented programs. In Cytron and Leavens [379], pages 17–26.
- [963] Frank Matthijs, Wouter Joosen, Bart Vanhaute, Bert Robben, and Pierre Verbaeten. Aspects should not die. In Lopes et al. [918].
- [964] Sean McDirmid and Wilson C. Hsieh. Aspect-oriented programming with Jiazzi. In Akşit [26], pages 70–79.
- [965] Sean McDirmid and Wilson C. Hsieh. Splice: Aspects that analyze programs. In Karsai and Visser [745], pages 19–38.
- [966] Nathan McEachen and Roger Alexander. Distributing classes with woven concerns—an exploration of potential fault scenarios. In Tarr [1361], pages 192–200.
- [967] David L. McReynolds, Sheryl L. Duggins, Doreen L. Galli, and John H. Mayer. Distributed characteristics of subject oriented programming: An evaluation with the process and object paradigms. In *Proceedings of the 37th Annual Southeast Regional Conference (CD-ROM)*, page 19. ACM Press, 1999.
- [968] Tariq Mehmood, Naveed Ashraf, and S. Tauseef-ur-Rehman. Framework for separating performance concerns and improved modularity in multi agent systems (mas) using aspect oriented programming (aop). In Bergmans et al. [156].
- [969] K. Mehner and G. Taentzner. Supporting aspect oriented modeling with graph transformations. In Clements et al. [320].

- [970] Katharina Mehner and Awais Rashid. Towards a standard interface for runtime inspection in AOP environments. In Chu-Carroll et al. [286].
- [971] Katharina Mehner and Annika Wagner. On the role of method families in aspect-oriented programming. In Lopes et al. [924].
- [972] François Mekerke, Geri Georg, Robert France, and Roger Alexander. Tool support for aspect-oriented design. In J.-M. Bruel and Z. Bellahsene, editors, *Advances in Object-Oriented Information Systems OOIS 2002 Workshops, LNCS 2426*, pages 280–289, September 2002.
- [973] Francois Mekerke, Wolfgang Theurer, and Joel Champeau. Non-functional aspects management for craft-oriented design. In *<<UML>> 2004 Workshop on Models for Non-functional Aspects of Component-Based Software*, October 2004.
- [974] Stephen J. Mellor. A framework for aspect-oriented modeling. In Aldawud et al. [44].
- [975] Juri Memmert. Application development in Java: From OOP to SOP. In Tarr et al. [1363].
- [976] Juri Memmert. Separation of concerns at the source. In Tarr and Ossher [1366].
- [977] Juri Memmert. Designing with Cosmos. In Glandrup et al. [564].
- [978] Juri Memmert. Is AOSD really worth is? In Bodkin et al. [185].
- [979] Juri Memmert. Concern patterns and analysis. In Robillard [1212], pages 1–5.
- [980] Anurag Mendhekar, Gregor Kiczales, and John Lamping. RG: A case-study for aspect-oriented programming. Technical Report SPL-97-009, Palo Alto Research Center, 1997.
- [981] K. Mens. Architectural aspects. In Rashid et al. [1191].
- [982] Kim Mens. Multiple cross-cutting architectural views. In Tarr et al. [1363].
- [983] Kim Mens, Tom Mens, and Michel Wermelinger. Supporting software evolution with intentional software views. In *Proceedings of the Workshop on Principles of Software Evolution*, pages 138–142. ACM Press, 2002.
- [984] Tom Mens and Michel Wermelinger. Separation of concerns for software evolution. *Journal of Software Maintenance and Evolution: Research and Practice*, 14(5):311–315, 2002.
- [985] Paulo Merson. Representing aspects in the software architecture practical considerations. In Baniassad et al. [109].
- [986] C. Mesquita, S. Barbosa, and C. de Lucena. Towards the identification of concerns in personalization mechanisms via scenarios. In Rashid et al. [1191].
- [987] B. Meyer. Tell less, say more: The power of implicitness. *IEEE Computer*, 31(7):97–98, July 1998.
- [988] M. Mezini. Dynamic metaclass construction for an explicit specialization interface. In G. Kiczales, editor, *Reflection '96*, pages 203–219, April 1996.
- [989] M. Mezini. Incremental redefinition of open implementations. In C. Zimmermann, editor, *Advances in Object-Oriented Metalevel Architectures and Reflection*, pages 265–290. CRC Press, Boca Raton, 1996.
- [990] M. Mezini. Dynamic object modification without name collisions. In Mehmet Akşit and Satoshi Matsuoka, editors, *11th European Conf. Object-Oriented Programming*, volume 1241 of *LNCS*, pages 190–219. Springer Verlag, 1997.
- [991] M. Mezini and K. Ostermann. Modules for crosscutting models. In *8th International Conference on Reliable Software Technologies (Ada-Europe '03)*. Springer-Verlag Lecture Notes in Computer Science, June 2003.
- [992] Mira Mezini. Maintaining the consistency of class libraries during their evolution. In *Proc. 1997 ACM SIGPLAN conference on Object-Oriented Programming Systems, Languages and Applications*, pages 1–21. ACM Press, 1997.
- [993] Mira Mezini. *Variation-Oriented Programming Beyond Classes and inheritance*. PhD thesis, University of Siegen, 1997.

- [994] Mira Mezini and Karl Lieberherr. Adaptive plug-and-play components for evolutionary software development. In C. Chambers, editor, *OOPSLA*, pages 97–116. ACM, October 1998.
- [995] Mira Mezini and Klaus Ostermann. Integrating independent components with on-demand remodularization. In *Proceedings of the 17th ACM conference on Object-oriented programming, systems, languages, and applications*, pages 52–67. ACM Press, 2002.
- [996] Mira Mezini and Klaus Ostermann. Variability management with aspects. In FSE04 [535], pages 127–136.
- [997] Mira Mezini and Klaus Ostermann. Untangling crosscutting models with CAESAR. In Filman et al. [513], pages 165–199.
- [998] Mira Mezini, Linda Seiter, and Karl Lieberherr. Component integration with pluggable composite adapters. In Mehmet Akşit, editor, *Symposium on Software Architectures and Component Technology: The State of the Art in Research and Practice*. Kluwer Academic Publishers, October 2001.
- [999] Isabel Michiels, Theo D’Hondt, Kris De Schutter, and Ghislain Hoffman. Using dynamic aspects to distill business rules from legacy code. In Filman et al. [502], pages 98–102.
- [1000] T. Mikkonen. On objects, aspects, and specifications addressing their collaboration. In Rashid et al. [1191].
- [1001] H. Mili, A. Elkharraz, and H. Mcheick. Understanding separation of concerns. In Tekinerdoğan et al. [1378], pages 75–84.
- [1002] H. Mili, H. Mcheick, and S. Sadou. Distribution and aspects. In Akşit and Choukair [28].
- [1003] H. Mili, H. Mcheick, and S. Sadou. Distribution and aspects. In Costanza et al. [371]. Technical report IAI-TR-2002-1.
- [1004] Hafedh Mili, Joumana Dargham, and Salah Bendelloul. Separation of concerns and typing: A first stab. In Ossher et al. [1091].
- [1005] Hafedh Mili, William Harrison, and Harold Ossher. Supporting subject-oriented programming in smalltalk. In *TOOLS USA 96*, August 1996.
- [1006] Naftaly Minsky. Law-governed regularities in object systems; Part 1: Principles. *Theory and Practice of Object Systems (TOPAS)*, 2(4), 1996.
- [1007] Naftaly Minsky and Partha Pal. Law-governed regularities in object systems; Part 2: The Eiffel case. In *Theory and Practice of Object Systems*, 3(2), 1997.
- [1008] Valérie Monfort, Muhammad Usman Bhatti, and Assia Ait Ali Slimane. Weaving aspects to support high reliable systems: Developing a blood plasma analysis automaton. In Coady et al. [328].
- [1009] Mattia Monga. Concern-specific aspect-oriented programming with Malaj. In Tarr et al. [1363].
- [1010] Mattia Monga. On aspect-oriented approaches. In Gybels et al. [616].
- [1011] Mattia Monga. Aspect-oriented programming as model driven evolution. In Tourwé et al. [1399].
- [1012] Miguel Monteiro and João Fernandes. Towards a catalog of aspect-oriented refactorings. In Tarr [1361], pages 111–122.
- [1013] Miguel P. Monteiro and Joao M. Fernandes. The search for aspect-oriented refactorings must go on. In Tourwé et al. [1399].
- [1014] Miguel Pessoa Monteiro and João Miguel Fernandes. Pitfalls of AspectJ implementations of some of the gang-of-four design patterns. In Fuentes et al. [538].
- [1015] Alexandra Weber Morales and John Reitano. The 15% solution. *Software Development*, 10(5):34–38, May 2002.
- [1016] Ana Moreira, João Araújo, and Isabel Brito. Crosscutting quality attributes for requirements engineering. In *Proceedings of the 14th International Conference on Software Engineering and Knowledge Engineering*, pages 167–174. ACM Press, 2002.
- [1017] Ana Moreira, Awais Rashid, Elisa Baniassad, Bedir Tekinerdoğan, Paul Clements, and João Araújo, editors. *Early Aspects 2004: Aspect-Oriented Requirements Engineering and Architecture Design Workshop at OOPSLA 2004*, October 2004.

- [1018] Farida Mostefaoui and Julie Vachon. Modélisation et vérification formelle de la composition des aspects. In Seinturier [1265].
- [1019] M. Mousavi, G. Russello, M. Chaudron, M. A. Reniers, T. Basten, A. Corsaro, S. Shukla, R. Gupta, and D. Schmid. Aspects + GAMMA = AspectGAMMA: A formal framework for aspect-oriented specification. In Rashid et al. [1191].
- [1020] Alexis Muller. Reusing functional aspects: From composition to parameterization. In Aldawud et al. [38].
- [1021] Jurgen K. Muller. Aspect-design in the building-block method. In Lopes et al. [918].
- [1022] Therthala Murali, Renaud Pawlak, and Houman Younessi. Applying aspect orientation to J2EE business tier patterns. In Lorenz and Coady [932].
- [1023] Gail Murphy and Christa Schwanninger. Special issue on aspect-oriented programming. *IEEE Software*, January 2006.
- [1024] Gail Murphy and Christa Schwanninger. Special issue on aspect-oriented programming. *IEEE Software*, January 2006.
- [1025] Gail C. Murphy, William G. Griswold, Martin P. Robillard, Jan Hannemann, and Wesley Leong. Design recommendations for concern elaboration tools. In Filman et al. [513], pages 507–530.
- [1026] Gail C. Murphy, Albert Lai, Robert J. Walker, and Martin P. Robillard. Separating features in source code: An exploratory study. In *Proc. 23rd Int'l Conf. Software Engineering*, pages 275–284. IEEE Computer Society, 2001.
- [1027] Gail C. Murphy, Robert J. Walker, and Elisa L. A. Baniassad. Evaluating emerging software development technologies: Lessons learned from assessing aspect-oriented programming. *IEEE Transactions on Software Engineering*, 25(4):438–455, 1999.
- [1028] Gail C. Murphy, Robert J. Walker, Elisa L. A. Baniassad, Martin P. Robillard, Albert Lai, and Mik A. Kersten. Does aspect-oriented programming work? *Comm. ACM*, 44(10):75–77, October 2001.
- [1029] István Nagy, Mehmet Akşit, and Lodewijk Bergmans. Composition graphs, a foundation for reasoning about aspect-oriented composition. In Leavens and Clifton [866].
- [1030] István Nagy and Lodewijk Bergmans. Towards semantic composition in aspect-oriented programming. In Gybels et al. [616].
- [1031] István Nagy, Lodewijk Bergmans, and Mehmet Akşit. Declarative aspect composition. In Bergmans et al. [155].
- [1032] Istvan Nagy, Lodewijk Bergmans, Gurcan Gulesir, Pascal Durr, and Mehmet Akşit. Generic, property based queries for evolvable weaving specifications. In Bergmans et al. [156].
- [1033] S. Nakajima and T. Tamai. Weaving in role-based aspect-oriented design models. In Moreira et al. [1017].
- [1034] Shin Nakajima. Separation of concerns in early stage of framework development. In Ossher et al. [1091].
- [1035] Shin Nakajima and Tetsuo Tamai. Lightweight formal analysis of aspect-oriented models. In Aldawud et al. [38].
- [1036] E. Navarro, P. Letelier, and I. Ramos. Goals and quality characteristics: Separating concerns. In Moreira et al. [1017].
- [1037] Elena Navarro, Patricio Letelier, and Isidro Ramos. UML visualization for an aspect and goal-oriented approach. In Aldawud et al. [38].
- [1038] Elena Navarro, Patricio Letelier, Isidro Ramos, and Bárbara Álvarez. Orientación a aspectos y orientación a objetivos: Una propuesta para su integración. In Fuentes et al. [538].
- [1039] A. Navasa, K. Palma, J. M. Murillo, and Y. Eterovic. Dos modelos arquitectónicos para el DSOA. In Fuentes et al. [538].
- [1040] A. Navasa, M. A. Pérez, and J. M. Murillo. Using an ADL to design aspect oriented systems. In Martín et al. [957].
- [1041] A. Navasa, M. A. Perez, J. M. Murillo, and J. Hernandez. Aspect-oriented software architecture: A structural perspective. In Rashid et al. [1191].

- [1042] Amparo Navasa, Miguel A. Pérez, and Juan M. Murillo. Developing component based systems using AOP concepts. In Bergmans et al. [154].
- [1043] R. D. Nebbe. Coordination and composition: The two paradigms underlying AOP? In Lopes et al. [914].
- [1044] Andrey Nechypurenko. Using design patterns to improve aspect reusability and dynamics. In Coady et al. [327].
- [1045] Torsten Nelson, Paulo Alencar, and Donald Cowan. Verifying multiple-perspective composition. In Tarr et al. [1362].
- [1046] Torsten Nelson, Donald Cowan, and Paulo Alencar. Towards a formal model of object-oriented hyperslices. In Ossher et al. [1091].
- [1047] Torsten Nelson, Donald Cowan, and Paulo Alencar. Formal verification of a bounded buffer with three separate concerns. In Tarr and Ossher [1366].
- [1048] Torsten Nelson, Donald Cowan, and Paulo Alencar. Supporting formal verification of crosscutting concerns. In Yonezawa and Matsuoka [1494], pages 153–169.
- [1049] Christian Nentwich, Wolfgang Emmerich, and Anthony Finkelstein. Edit, compile, debug—From hacking to distributed engineering. In Glandrup et al. [564].
- [1050] P. Netinant, C. A. Constantinides, T. Elrad, and M. E. Fayad. Aspect-oriented frameworks (poster session): The design of adaptable operating systems. In *Addendum to the 2000 proceedings of the conference on Object-oriented programming, systems, languages, and applications (Addendum)*, pages 61–62. ACM Press, 2000.
- [1051] Paniti Netinant, Constantinos Constantinides, Tzilla Elrad, and Mohamed Fayad. Supporting aspectual decomposition in the design of operating systems. In *Workshop on Object-Oriented and Operating Systems (ECOOP)*, 2000.
- [1052] Paniti Netinant, Tzilla Elrad, and Mohamed E. Fayad. A layered approach to building open aspect-oriented systems: A framework for the design of on-demand system demodularization. *Comm. ACM*, 44(10):83–85, October 2001.
- [1053] Elissa Newman. Localizing views for separation of concerns. In Tarr and Ossher [1366].
- [1054] Elissa Newman. Query-based modeling and fluid. In Bergmans et al. [156].
- [1055] Elissa Newman and William L. Scherlis. Toward query-based constraints. In Bergmans et al. [153].
- [1056] Oscar Nierstrasz and Franz Achermann. Separation of concerns through unification of concepts. In Lopes et al. [913].
- [1057] Oscar Nierstrasz and Franz Achermann. Supporting compositional styles for software evolution. In *Proceedings Int'l Symposium on Principles of Software Evolution (ISPSE 2000)*, pages 11–19. IEEE, November 2000.
- [1058] Oscar Nierstrasz and Franz Achermann. Separating concerns with first-class namespaces. In Filman et al. [513], pages 243–259.
- [1059] Muga Nishizawa, Shigeru Chiba, and Michiaki Tatsubori. Remote pointcut: a language construct for distributed AOP. In Lieberherr [884], pages 7–15.
- [1060] Natsuko Noda and Tomoji Kishi. On aspect-oriented design—Applying “multi-dimensional separation of concerns” on designing quality attributes. In Ossher et al. [1091].
- [1061] Natsuko Noda and Tomoji Kishi. Implementing design patterns using advanced separation of concerns. In De Volder et al. [407].
- [1062] Martin E. Nordberg III. Aspect-oriented dependency inversion. In De Volder et al. [407].
- [1063] Martin E. Nordberg III. Aspect-oriented dependency management. In Filman et al. [513], pages 557–584.
- [1064] D. Notkin, D. Garlan, W. G. Griswold, and K. Sullivan. Adding implicit invocation to languages: Three approaches. In S. Nishio and A. Yonezawa, editors, *Object Technologies for Advanced Software: 1st JSSST Int'l Symposium (ISOTAS)*, volume 742 of *lncs*, pages 489–510, Berlin, November 1993. Springer-Verlag.
- [1065] A. Noutash, M van Sinderen, M. Akşit, and L. Bergmans. QoS provisioning in CORBA by introducing aspect-oriented transport reflection point. In *ECOOP'2000 Workshop on Quality of Service*, June 2000.

- [1066] Bashar Nuseibeh. Crosscutting requirements. In Lieberherr [884], pages 3–4.
- [1067] Alexander Nyssen, Shmuel Tyszberowicz, and Thomas Weiler. Are aspects useful for managing variability in software product lines? a case study. In Araújo et al. [79].
- [1068] Brian Odgers and Simon Thompson. Aspect-oriented process engineering. In Lopes et al. [924].
- [1069] Graham O’Regan. Introduction to aspect-oriented programming. *O’Reilly On-Java.com*, January 2004.
- [1070] Doug Orleans. Separating behavioral concerns with predicate dispatch, or, if statement considered harmful. In De Volder et al. [407].
- [1071] Doug Orleans. Incremental programming with extensible decisions. In Kiczales [781], pages 56–64.
- [1072] Doug Orleans and Karl Lieberherr. DJ: Dynamic adaptive programming in Java. In Yonezawa and Matsuoka [1494], pages 73–80.
- [1073] Doug Orleans and Karl J. Lieberherr. AP library: The core algorithms of AP. Technical report, Northeastern University, May 1999.
- [1074] Doug Orleans and Karl J. Lieberherr. DemeterJ. Technical report, Northeastern University, 2001.
- [1075] Doug Orleans and Karl J. Lieberherr. DAJ: Demeter in AspectJ. Technical report, Northeastern University, January 2003.
- [1076] Guadalupe Ortiz, Juan Hernaández, and Pedro J. Clemente. Configuración de propiedades no funcionales en los servicios web usando técnicas orientadas a aspecto. In Fuentes et al. [538].
- [1077] H. Ossher, M. Kaplan, A. Katz, W. Harrison, and V. Kruskal. Specifying subject-oriented composition. *Theory and Practice of Object Systems*, 2(3), 1996.
- [1078] H. Ossher and P. Tarr. Operation-level composition: A case in (join) point. In Lopes et al. [914].
- [1079] H. Ossher and P. Tarr. Multi-dimensional separation of concerns using Hyperspaces. Technical Report 21452, IBM Research Report, April 1999.
- [1080] H. Ossher and P. Tarr. Multi-dimensional separation of concerns and the Hyperspace approach. In Aksit [12].
- [1081] H. Ossher and P. Tarr. The shape of things to come: Using multi-dimensional separation of concerns with Hyper/J to (re)shape evolving software. *Comm. ACM*, 44(10):43–50, October 2001.
- [1082] Harold Ossher and William Harrison. Combination of inheritance hierarchies. In *Proc. 1992 Conf. Object-Oriented Programming Systems, Languages, and Applications*, October 1992.
- [1083] Harold Ossher, William Harrison, Frank Budinsky, and Ian Simmonds. Subject-oriented programming: Supporting decentralized development of objects. In *Proc. 7th IBM Conf. Object-Oriented Technology*, July 1994.
- [1084] Harold Ossher, Matthew Kaplan, William Harrison, Alexander Katz, and Vincent Kruskal. Subject-oriented composition rules. In *Proc. 1995 Conf. Object-Oriented Programming Systems, Languages, and Applications*, October 1995.
- [1085] Harold Ossher and Peri Tarr. Multi-dimensional separation of concerns in hyperspace. In Lopes et al. [924].
- [1086] Harold Ossher and Peri Tarr. Using subject-oriented programming to overcome common problems in object-oriented software development/evolution. In *Proc. 21st Int’l Conf. Software Engineering*, pages 687–688. IEEE Computer Society Press, 1999.
- [1087] Harold Ossher and Peri Tarr. Hyper/J: Multi-dimensional separation of concerns for Java. In *Proc. 22nd Int’l Conf. Software Engineering*, pages 734–737. ACM Press, 2000.
- [1088] Harold Ossher and Peri Tarr. On the need for on-demand remodularization. In Lopes et al. [913].
- [1089] Harold Ossher and Peri Tarr. Hyper/J: Multi-dimensional separation of concerns for Java. In *Proc. 23rd Int’l Conf. on Software*

- Engineering*, pages 729–730. IEEE Computer Society, 2001.
- [1090] Harold Ossher and Peri Tarr. Some micro-reuse challenges. In Bergmans et al. [154].
- [1091] Harold Ossher, Peri Tarr, and Gail Murphy, editors. *Workshop on Multi-Dimensional Separation of Concerns (OOPSLA 1999)*, November 1999.
- [1092] Klaus Ostermann. Dynamically composable collaborations with delegation layers. In Magnusson [944], pages 89–110.
- [1093] Klaus Ostermann and Günter Kniesel. Independent extensibility—An open challenge for AspectJ and Hyper/J. In Lopes et al. [913].
- [1094] Klaus Ostermann and Mira Mezini. Object-oriented composition is tangled. In Bergmans et al. [154].
- [1095] Klaus Ostermann and Mira Mezini. Object-oriented composition untangled. In *Proc. OOPSLA '01 Conf. Object Oriented Programming Systems Languages and Applications*, pages 283–299. ACM Press, 2001.
- [1096] Klaus Ostermann and Mira Mezini. Conquering aspects with Caesar. In Akşit [26], pages 90–99.
- [1097] Johan Ovlinger. From aspect oriented model to implementation watch out for impedance and mismatch. In Aldawud et al. [43].
- [1098] Johan Ovlinger, Karl Lieberherr, and David Lorenz. Aspects and modules combined. Technical Report NU-CCS-02-03, College of Computer Science, Northeastern University, Boston, MA, March 2002.
- [1099] Robert R. Painter and David Coppit. A model for software plans. In Robillard [1212], pages 1–5.
- [1100] Jeffrey D. Palm, Kenneth M. Anderson, and Karl J. Lieberherr. Investigating the relationship between violations of the law of demeter and software maintainability. In Bergmans et al. [153].
- [1101] Jens Palsberg. Class-graph inference for adaptive programs. *Theory and Practice of Object Systems*, 3(2):75–85, April 1997.
- [1102] Jens Palsberg, Boaz Patt-Shamir, and Karl Lieberherr. A new approach to compiling adaptive programs. In Hanne Riis Nielson, editor, *European Symposium on Programming*, pages 280–295. Springer Verlag Lecture Notes in Computer Science 1058, April 1996.
- [1103] Jens Palsberg, Boaz Patt-Shamir, and Karl Lieberherr. A new approach to compiling adaptive programs. *Science of Computer Programming*, 29(3):303–326, 1997.
- [1104] Jens Palsberg, Cun Xiao, and Karl Lieberherr. Efficient implementation of adaptive software. *ACM Transactions on Programming Languages and Systems (TOPLAS)*, 17(2):264–292, 1995.
- [1105] Thomas Panas, Jesper Andersson, and Uwe Aßmann. The editing aspect of aspects. In I. Hussain, editor, *Software Engineering and Applications (SEA2002)*, Cambridge, MA, November 2002. ACTA Press.
- [1106] Thomas Panas, Jonas Karlsson, and Magnus Högberg. Aspect-jEdit for inline aspect support. In Bachmendo et al. [97].
- [1107] J. Pang and L. Blair. An adaptive run time manager for the dynamic integration and interaction resolution of features. In Akşit and Choukair [28].
- [1108] J. Pang and L. Blair. Refining feature driven development—A methodology for early aspects. In Tekinerdoğan et al. [1378], pages 85–90.
- [1109] Marta Pantoquilha and Ana Moreira. Aspect-oriented logical architecture design a layered perspective applied to data warehousing. In Fuentes et al. [538].
- [1110] Holger Papajewski, Olaf Spinczyk, and Danilo Beuche. Success factors for AOSD in commercial applications. In Bodkin et al. [185].
- [1111] Odysseas Papapetrou and George A. Papadopoulos. Aspect oriented programming for a component-based real life application: a case study. In *Proceedings of the 2004 ACM symposium on Applied computing*, pages 1554–1558. ACM Press, 2004.

- [1112] Daesung Park and Sungwon Kang. Design phase analysis of software performance using aspect-oriented programming. In Aldawud et al. [38].
- [1113] D. L. Parnas. On the criteria to be used in decomposing systems into modules. *Comm. ACM*, 15(12):1053–1058, December 1972.
- [1114] Jaime Pavlich-Mariscal, Laurent Michel, and Steven Demurjian. Role slices and runtime permissions: Improving an aop-based access control schema. In Aldawud et al. [42].
- [1115] R. Pawlak. *AOSD with JAC*. PhD thesis, CNAM Paris, December 2002.
- [1116] R. Pawlak, L. Seinturier, and L. Duchien. JAC milestone 2003. Technical Report 2003-04, Computer Science Laboratory of Lille, 2003.  
<http://www.lifl.fr/~duchien/duchien.ps>.
- [1117] Renaud Pawlak. Nature and benefits of aspect-oriented programming. In Lopes et al. [913].
- [1118] Renaud Pawlak and Remi Douence, editors. *First workshop on Aspect Orientation for Middleware Development (AOMD 2005)*, November 2005.
- [1119] Renaud Pawlak, Laurence Duchien, and Gerard Florin. An automatic aspect weaver with a reflective programming language. In *2nd Int'l Conf. Meta-Level Architectures and Reflection*, volume 1616 of *lncs*, pages 147–149, Berlin, 1999. Springer-Verlag.
- [1120] Renaud Pawlak, Laurence Duchien, Gerard Florin, Fabrice Legond-Aubry, Lionel Seinturier, and Laurent Martelli. A UML notation for aspect-oriented software design. In Aldawud et al. [37].
- [1121] Renaud Pawlak, Laurence Duchien, Gerard Florin, and Lionel Seinturier. Dynamic wrappers: Handling the composition issue with JAC. In *39th Int'l Conf. and Exhibition on Technology of Object-Oriented Languages and Systems (TOOLS39)*, pages 56–65. IEEE, 2001.
- [1122] Renaud Pawlak, Jean-Philippe Retraillé, and Lionel Seinturier. *Programmation orientée aspect pour Java/J2EE*. Eyrolles, 2004.
- [1123] Renaud Pawlak, Jean-Philippe Retraillé, and Lionel Seinturier. *Foundations of AOP for J2EE Development*. APress, 2005.
- [1124] Renaud Pawlak, Lionel Seinturier, Laurence Duchien, and Gérard Florin. JAC: A flexible solution for aspect-oriented programming in Java. In Yonezawa and Matsuoka [1494], pages 1–24.
- [1125] Renaud Pawlak, Lionel Seinturier, Laurence Duchien, Laurent Martelli, Fabrice Legond-Aubry, and Gérard Florin. Aspect-oriented software development with Java Aspect Components. In Filman et al. [513], pages 343–369.
- [1126] Renaud Pawlak and Houman Younessi. On getting use cases and aspects to work together. *Journal of Object Technology*, 3(1), January 2004.
- [1127] José J. Pazos-Arias, Jorge García-Duque, and Martín López-Nores. Locating crosscutting concerns in the formal specification of distributed reactive systems. In Robillard [1212], pages 1–5.
- [1128] Luca Pazzi. Explicit aspect composition by part-whole state charts. In Lopes et al. [924].
- [1129] Jani Pesonen, Mika Katara, and Tommi Mikkonen. Evaluating an aspect-oriented approach for production-testing software. In Coady et al. [328].
- [1130] Nicolas Pessemier. Fractal aspect component. In Seinturier [1265].
- [1131] Nicolas Pessemier, Lionel Seinturier, and Laurence Duchien. Components, ADL and AOP: Towards a common approach. In Cazola et al. [255], pages 61–70.
- [1132] Ilka Philippow, Matthias Riebisch, and Kai Böllert. The Hyper/UML approach for feature based software design. In Aldawud et al. [44].
- [1133] G. Piccinelli and L. Mokrushin. Dynamic e-service composition in Dysco. In Choukair [280], pages 88–96.
- [1134] G. Piccinelli and M. Stearns. Managing interaction concerns in web-service systems. In Akşit and Choukair [28].

- [1135] Roman Pichler, Klaus Ostermann, and Mira Mezini. On aspectualizing component models. *Software Practice and Experience*, 33(10):957–974, 2003.
- [1136] Scott M. Pike. Binary trees: A challenge problem for separating concerns. In Tarr and Ossher [1366].
- [1137] M. Pinto, M. Amor, L. Fuentes, and J. M. Troya. Collaborative virtual environment development: An aspect-oriented approach. In Choukair [280], pages 97–102.
- [1138] M. Pinto, M. Amor, L. Fuentes, and J. M. Troya. Run-time coordination of components: Design patterns vs. component-aspect based platforms. In Bergmans et al. [154].
- [1139] Monica Pinto, Lidia Fuentes, Mohamed Fayad, and Jose Maria Troya. Separation of coordination in a dynamic aspect oriented framework. In Kiczales [781], pages 134–140.
- [1140] Mónica Pinto, Daniel Jiménez, and Lidia Fuentes. Developing dynamic and adaptable applications with CAM/DAOP: A virtual office application. In Glück and Lowry [565], pages 438–441.
- [1141] Eduardo Kessler Piveta and Augusto Jun Devegil. Aspects in the rational unified process’ analysis and design workflow. In Glan-drup et al. [564].
- [1142] Eduardo Kessler Piveta and Luiz Carlos Zancanella. Aurélia: Aspect oriented programming using a reflective approach. In Bergmans et al. [154].
- [1143] Robyn Polan and Elizabeth Kendall. A behavioural model of cross-cutting concerns in domain specific frameworks. In Huang et al. [701].
- [1144] Gary Pollice. A look at aspect-oriented programming. Technical report, IBM Developer Works, 2004.
- [1145] Adrian Pop, Ilie Savga, Uwe Aßmann, and Peter Fritzson. Composition of XML dialects: A ModelicaXML case study. In Uwe Aßmann and Adrian Pop, editors, *Software Composition Workshop (ETAPS)*, pages 126–141, March 2004.
- [1146] A. Popovici, G. Alonso, and T. Gross. AOP support for mobile systems. In De Volder et al. [407].
- [1147] Andrei Popovici, Gustavo Alonso, and Thomas Gross. Just in time aspects. In Aksit [26], pages 100–109.
- [1148] Andrei Popovici, Gustavo Alonso, and Thomas Gross. Dynamic aspect-oriented infrastructure. In Filman et al. [513], pages 711–730.
- [1149] Andrei Popovici, Thomas Gross, and Gustavo Alonso. Dynamic weaving for aspect-oriented programming. In Kiczales [781], pages 141–147.
- [1150] M. Devi Prasad. Typecasting as a new join point in AspectJ. In Martín et al. [957].
- [1151] M. Devi Prasad and B. D. Chaudhary. AOP support for C#. In Coady et al. [326].
- [1152] R. M. Pratap, F. Hunleth, and R. K. Cytron. Building fully customisable middleware using an aspect-oriented approach. *IEE Proceedings Software*, 151(4):199–216, August 2004.
- [1153] Christian Prehofer. Feature interactions in statechart diagrams or graphical composition of components. In Kandé et al. [735].
- [1154] Christian Prehofer. Graphical composition of components with feature interactions. In Aldawud et al. [37].
- [1155] Jane Pryor and Natalio Bastán. A reflective architecture for the support of aspect-oriented programming in Smalltalk. In Lopes et al. [924].
- [1156] E. Pulvermueller, A. Speck, and A. Rashid. Implementing collaboration-based designs using aspect-oriented programming. In *Technology of Object-Oriented Languages and Systems (TOOLS USA)*, pages 95–104, 2000.
- [1157] Elke Pulvermüller, Andreas Speck, and James O. Coplien. A version model for aspect dependency management. In Bosch [195], pages 70–79.
- [1158] E. Putrycz and G. Bernard. Client side reconfiguration on software components for load balancing. In Choukair [280], pages 111–116.

- [1159] E. Putrycz and G. Bernard. Using aspect oriented programming to build a portable load balancing service. In Akşit and Choukair [28].
- [1160] Miloud Rached, Jean-Paul Bodeveix, Mamoun Filfali, and Odile Nasr. Real time aspects: Specification and composition in b. In Aldawud et al. [42].
- [1161] Hridesh Rajan and Kevin Sullivan. Eos: instance-level aspects for integrated system design. In *ESEC/FSE-11: Proceedings of the 9th European software engineering conference held jointly with 11th ACM SIG-SOFT international symposium on Foundations of software engineering*, pages 297–306, New York, NY, USA, 2003. ACM Press.
- [1162] Hridesh Rajan and Kevin Sullivan. Need for instance level aspect language with rich pointcut language. In Bergmans et al. [153].
- [1163] Hridesh Rajan and Kevin Sullivan. Aspect language features for concern coverage profiling. In Tarr [1361], pages 181–191.
- [1164] Hridesh Rajan and Kevin J. Sullivan. Classpects: unifying aspect- and object-oriented language design. In *ICSE '05: Proceedings of the 27th international conference on Software engineering*, pages 59–68, New York, 2005. ACM Press.
- [1165] Rajeev R. Raje, Ming Zhong, and Tongyu Wang. Case study: A distributed concurrent system with AspectJ. *ACM SIGAPP Applied Computing Review*, 9(2):17–23, 2001.
- [1166] Rafael Ramirez and Andrew E. Santosa. An aspect-oriented methodology for concurrent and distributed applications. In Bachmendo et al. [97].
- [1167] A. Rashid. *A Database Evolution Approach for Object-Oriented Databases*. PhD thesis, Lancaster University, 2000.
- [1168] A. Rashid. On to aspect persistence. In *2nd International Symposium on Generative and Component-Based Software Engineering, LNCS 2177*, pages 453–463. Springer-Verlag Lecture Notes in Computer Science, October 2000.
- [1169] A. Rashid. Aspect-oriented and component-based software engineering. *IEE Proceedings-Software*, 148(3):87–88, June 2001.
- [1170] A. Rashid. A hybrid approach to separation of concerns: The story of SADES. In *3rd Int'l Conf. Meta-Level Architectures and Separation of Concerns (Reflection 2001)*, volume 2192 of *lncs*, pages 231–249, Berlin, 2001. Springer-Verlag.
- [1171] A. Rashid. Multi-paradigm implementation of an object database evolution system. In *Workshop on Multi-Paradigm Programming with OO Languages (MPOOL) (ECOOP)*, pages 1–9. John von Neuman Institute for Computing, 3-00-007968-8, 2001.
- [1172] A. Rashid. A framework for customizable schema evolution in object-oriented databases. In *Int'l Data Engineering and Applications Symposium (IDEAS)*, pages 342–346. IEEE, 2003.
- [1173] A. Rashid and G. Kortuem. Adaptation as an aspect in pervasive computing. In Lopes et al. [929].
- [1174] A. Rashid, A. Moreira, and B. Tekinerdogan. Editorial - early aspects: aspect-oriented requirements engineering and architecture design. *IEE Proceedings Software*, 151(4):153–155, August 2004.
- [1175] A. Rashid and E. Pulvermueller. From object-oriented to aspect-oriented databases. In *11th International Conference on Database and Expert Systems Applications — DEXA 2000, LNCS 1873*, pages 125–134. Springer-Verlag Lecture Notes in Computer Science, September 2000.
- [1176] A. Rashid and P. Sawyer. Object database evolution using separation of concerns. *SIG-MOD Record*, 29(4):26–33, 2000.
- [1177] A. Rashid, P. Sawyer, A. Moreira, and J. Araújo. Early aspects: A model for aspect-oriented requirements engineering. In *Joint Int'l Conf. Requirements Engineering*, pages 199–202. IEEE, 2002.
- [1178] A. Rashid, P. Sawyer, and E. Pulvermueller. A flexible approach for instance adaptation during class versioning. In *Symp. Objects and Databases (ECOOP)*, volume 1944 of *lncs*, pages 101–113, Berlin, 2000. Springer-Verlag.

- [1179] Awais Rashid. A database evolution approach for object-oriented databases. In *Proc. IEEE Int'l Conf. Software Maintenance (ICSM 2001)*, pages 561–564, November 2001.
- [1180] Awais Rashid. A hybrid approach to separation of concerns: The story of SADES. In Yonezawa and Matsuoka [1494], pages 231–249.
- [1181] Awais Rashid, editor. *Workshop on Aspect-Oriented Programming and Separation of Concerns (Lancaster)*, August 2001.
- [1182] Awais Rashid. Weaving aspects in a persistent environment. *ACM SIGPLAN Notices*, 37(2), February 2002.
- [1183] Awais Rashid. *Aspect-Oriented Database Systems*. Springer-Verlag, 2004.
- [1184] Awais Rashid. Aspect-oriented programming for database systems. In Filman et al. [513], pages 657–680.
- [1185] Awais Rashid and Lynne Blair. Aspect-oriented programming and separation of crosscutting concerns. *The Computer Journal*, 46(5):527–528, September 2003.
- [1186] Awais Rashid and Ruzanna Chitchyan. Persistence as an aspect. In Akşit [26], pages 120–129.
- [1187] Awais Rashid and Gerald Kotonya. Risk management in component-based development: A separation of concerns perspective. In Bergmans et al. [154].
- [1188] Awais Rashid and Nicholas Leidenfrost. Supporting flexible object database evolution with aspects. In Karsai and Visser [745], pages 75–94.
- [1189] Awais Rashid, Ana Moreira, and João Araújo. Modularisation and composition of aspectual requirements. In Akşit [26], pages 11–20.
- [1190] Awais Rashid and Peter Sawyer. Aspect-orientation and database systems: An effective customisation approach. *IEE Proceedings - Software*, 148(5):156–164, October 2001.
- [1191] Awais Rashid, Bedir Tekinerdoğan, Ana Moreira, João Araújo, Jeff Gray, Jan Gerben Wijnstra, and Paul Clements, editors. *Workshop on Early Aspects: Aspect-Oriented Requirements Engineering and Architecture Design (AOSD-2002)*, March 2002.
- [1192] U. Rasthofer and F. Bellosa. Component-based software engineering for distributed embedded real-time systems. *IEE Proceedings-Software*, 148(3):99–103, June 2001.
- [1193] Andreas Rausch, Bernhard Rumpe, and Lucien Hoogendoorn. Aspect-oriented framework modeling. In Aldawud et al. [44].
- [1194] I. Ray, R. France, N. Li, and G. Georg. An aspect-based approach to modeling access control concern. *Information and Software Technology*, pages 557–587, July 2004.
- [1195] Raghu Reddy, Robert France, and Geri Georg. An aspect oriented approach to analyzing dependability features. In Kandé et al. [737].
- [1196] Raghu Reddy, Robert France, Sudipto Ghosh, Franck Fleurey, and Benoit Baudry. Model composition - a signature-based approach. In Aldawud et al. [42].
- [1197] Barry Redmond and Vinny Cahill. Supporting unanticipated dynamic adaptation of application behaviour. In Magnusson [944], pages 205–230.
- [1198] John Regehr and Alastair Reid. Lock inference for systems software. In Coady et al. [326].
- [1199] A. M. Reina. Separating the navigational aspect. In Akşit and Choukair [28].
- [1200] A. M. Reina and J. Torres. Analysing the navigational aspect. In Costanza et al. [371]. Technical report IAI-TR-2002-1.
- [1201] A. M. Reina, J. Torres, and M. Toro. Aspect-oriented web development vs. non aspect-oriented web development. In Hanemann et al. [640].
- [1202] A. M. Reina, J. Torres, and M. Toro. Separating concerns by means of UML-profiles and metamodels in PIMs. In Aldawud et al. [38].

- [1203] A. M. Reina, J. Torres, M. Toro, and M. J. Escalona. Modelando aspectos con lenguajes específicos de dominio. In Fuentes et al. [538].
- [1204] R. Q. Reis, C. A. Lima Reis, H. Schlebbe, and D. J. Nunes. Towards an aspect-oriented approach to improve the reusability of software process models. In Rashid et al. [1191].
- [1205] Karen Renaud. HERCULE: Non-invasively tracking Java component-based application activity. In Bertino [163], pages 447–471.
- [1206] Joel Richardson and Peter Schwarz. Aspects: Extending objects to support multiple, independent roles. In *Proceedings of the 1991 ACM SIGMOD International Conference on Management of Data*, pages 298–307. ACM Press, 1991.
- [1207] Mark Richters and Martin Gogolla. Aspect-oriented monitoring of UML and OCL constraints. In Aldawud et al. [44].
- [1208] Martin Rinard, Alexandru Salcianu, and Suhabe Bugrara. A classification system and analysis for interactions in aspect-oriented programs. In FSE04 [535], pages 147–158.
- [1209] Bert Robben and Patrick Steyaert. Aspects on TV. In Lopes et al. [913].
- [1210] M. P. Robillard. The Jex home page. <http://www.cs.ubc.ca/~mrobilla/jex/>.
- [1211] M. P. Robillard and G. C. Murphy. Concern Graphs: Finding and describing concerns using structural program dependencies. In *Int'l Conf. Software Engineering (ICSE)*, pages 406–416, 2002.
- [1212] Martin Robillard, editor. *ICSE Workshop on the Modeling and Analysis of Concerns in Software (MACS 2005)*, May 2005.
- [1213] Martin P. Robillard. Workshop on the modeling and analysis of concerns in software (macs 2005). *SIGSOFT Softw. Eng. Notes*, 30(4):1–3, 2005.
- [1214] Martin P. Robillard and Gail C. Murphy. Migrating a static analysis tool to AspectJ. In Ossher et al. [1091].
- [1215] Martin P. Robillard and Gail C. Murphy. An exploration of a lightweight means of concern separation. In Lopes et al. [913].
- [1216] Martin P. Robillard and Gail C. Murphy. Analyzing concerns using class member dependencies. In Tarr and Ossher [1366].
- [1217] Martin P. Robillard and Gail C. Murphy. Capturing concern descriptions during program navigation. In Chu-Carroll et al. [286].
- [1218] Martin P. Robillard and Gail C. Murphy. Just-in-time concern modeling. In Robillard [1212], pages 1–3.
- [1219] Martin P. Robillard and Peri Tarr. First international workshop on the modeling and analysis of concerns in software (macs 2005). In *ICSE '05: Proceedings of the 27th international conference on Software engineering*, pages 685–685, New York, 2005. ACM Press.
- [1220] Philip Robinson, Maarten Rits, and Roger Kilian-Kehr. An aspect of application security management. In De Win et al. [413].
- [1221] André Rocha, Valter Vieira Camargo, and Paulo Cesar Masiero. Uso de aspectos para verificar regras de instanciação de frameworks. In Garcia et al. [549].
- [1222] André Rocha, Adenilso da Silva Simão, José Carlos Maldonado, and Paulo Cesar Masiero. Teste funcional: Uma abordagem auxiliada por aspectos. In Garcia et al. [549].
- [1223] Maria del Pilar Romay Rodriguez, Carlos E. Cuesta, Pablo de la Fuente, and Manuel Barrio Soloórzano. Descripción de aspectos mediante conectores UML 2.0. In Fuentes et al. [538].
- [1224] L. Rosenhainer. Identifying crosscutting concerns in requirements specifications. In Moreira et al. [1017].
- [1225] Lars Rosenhainer. The discern method: Dealing separately with crosscutting concerns. In Baniassad et al. [109].
- [1226] Yves Roudier and Yuuji Ichisugi. Mixin composition strategies for the modular implementation of aspect weaving. In Lopes et al. [926].
- [1227] Isabelle Rouvellou, Stanley M. Sutton, Jr., and Stefan Tai. Multidimensional separation of concerns in middleware. In Tarr et al. [1363].

- [1228] Suman Roychoudhury and Jeff Gray. Towards language independent weaving using grammar adapters. In Tourwé et al. [1399].
- [1229] Daniel Sabbah. Aspects: from promise to reality. In Lieberherr [884], pages 1–2.
- [1230] Motoshi Saeki and Haruhiko Kaiya. Transformation based approach for weaving use case models in aspect-oriented requirements analysis. In Aldawud et al. [44].
- [1231] Kouhei Sakurai, Hidehiko Masuhara, Naoyasu Ubayashi, Saeko Matsuura, and Seiichi Komiya. Association aspects. In Lieberherr [884], pages 16–25.
- [1232] Mazen Saleh and Hassan Gomaa. Separation of concerns in software product line engineering. In Robillard [1212], pages 1–5.
- [1233] P. Salinas. Adding systemic crosscutting and super-imposition to Composition Filters. Master’s thesis, Vrije Universiteit Brussel, August 2001.
- [1234] A. Sampaio, N. Loughran, A. Rashid, and P. Rayson. Mining aspects in requirements. In Clements et al. [320].
- [1235] Americo Sampaio and Awais Rashid. Report on tools for knowlegde repository. Technical Report AOSD-Europe-ULANC-4, AOSD-Europe, April 2005.
- [1236] Americo Sampaio and Awais Rashid. Report on tools for web portal construction. Technical Report AOSD-Europe-ULANC-3, AOSD-Europe, April 2005.
- [1237] F. Sánchez, J. Hernandez, J. M. Murillo, and E. Pedarza. Run-time adaptability of synchronization policies in concurrent object-oriented languages. In Lopes et al. [914].
- [1238] Cláudio Sant’Anna, Alessandro Garcia, Christina Chavez, Carlos Lucena, and Arndt von Staa. On the reuse and maintenance of aspect-oriented software: An assessment framework. In *XVII Brazilian Symposium on Software Engineering*, October 2003.
- [1239] Naomi Sapir, Shmuel Tyszberowicz, and Amiram Yehudai. Extending UML with aspect usage constraints in the analysis and design phases. In Kandé et al. [735].
- [1240] A. Sassen, G. Amoros, P. Donth, K. Geihs, J. Jézéquel, K. Odent, N. Plouzeau, and T. Weis. QCCS: A methodology for the development of contract-aware components based on aspect-oriented design. In Rashid et al. [1191].
- [1241] Yoshiki Sato and Shigeru Chiba. Negligent class loaders for software evolution. In Cazola et al. [255], pages 53–60.
- [1242] L. Savarese. Aspect oriented software development applied to track modeling for command and control. Master’s thesis, University of Twente, 2003.
- [1243] Juha Savolainen. Improving product line development with subject-oriented programming. In Tarr et al. [1363].
- [1244] Juha Savolainen. Towards multi dimensional methods. In Tarr et al. [1362].
- [1245] Vibha Sazawal. Separation of concerns for ubiquitous computing. In Tarr and Ossher [1366].
- [1246] Thorsten Schafer, Michael Eichberg, and Mira Mezini. Towards exploring crosscutting concerns. In Tourwé et al. [1399].
- [1247] R. Schantz, D. Bakken, D. Karr, J. Loyall, and J. Zinky. Distributed objects with quality of service: An organizing architecture for integrated system properties. In *OMG-DARPA-MCC Workshop on Compositional Software Architectures*, January 1998.
- [1248] Nathanael Schärli and Franz Achermann. Partial evaluation of inter-language wrappers. In *Workshop on Composition Languages, WCL’01 (ESEC/FSE)*, September 2001.
- [1249] Hans Schippers and Dirk Janssens. A model driven approach to multi-dimensional separation of concerns with OC. In Aksit et al. [35].
- [1250] Arno Schmidmeier. Transferring persistence concepts in Java ODBMSs to AspectJ based on ODMG standards. In Rashid [1181].
- [1251] Arno Schmidmeier. Hard times are good times for AOSD. In Bodkin et al. [185].
- [1252] Arno Schmidmeier, Stefan Hanenberg, and Rainer Unland. Known concepts implemented in AspectJ. In Bachmendo et al. [97].

- [1253] Phillip Schmidt, Sergio Alvarado, Jaime Milstein, Gregory Mulert, Robert Duvall, and Jesus Rivera. A systems engineering perspective of aspect-oriented software architectural analysis using UML. In Aldawud et al. [43].
- [1254] R. Schmidt and U. Aßmann. Extending aspect-oriented programming in order to flexibly support workflows. In Lopes et al. [926].
- [1255] Andreas I. Schmied and Franz J. Hauck. Composing non-orthogonal aspects. In Martín et al. [957].
- [1256] S. Schonger, E. Pulvermueller, and S. Sarstedt. Aspect oriented programming and component weaving: Using XML representations of abstract syntax trees. In Costanza et al. [371]. Technical report IAI-TR-2002-1.
- [1257] W. Schult and A. Polze. Aspect-oriented programming with C# and .NET. In *Int'l Symp. Object-oriented Real-time distributed Computing (ISORC)*, pages 241–248, 2002.
- [1258] Wolfgang Schult and Andreas Polze. Speed vs. memory usage - an approach to deal with contrary aspects. In Coady et al. [326].
- [1259] Karsten Schulz and Maria E. Orlowska. Architectural issues for cross-organizational B2B interactions. In Choukair [280], pages 79–87.
- [1260] Mario Schüpany, Christa Schwanninger, and Egon Wuchner. Aspect-oriented programming for .NET. In Coady et al. [327].
- [1261] Christa Schwanninger, Michael Kircher, and Egon Wuchner. Encapsulating crosscutting concerns in system software. In Lorenz and Coady [932].
- [1262] Christa Schwanninger and Egon Wuchner. Commercialization. In Bodkin et al. [185].
- [1263] Marc Segura-Devillechaise, Jean-Marc Menaud, Gilles Muller, and Julia L. Lawall. Web cache prefetching as an aspect: Towards a dynamic-weaving based solution. In Akşit [26], pages 110–119.
- [1264] Lionel Seinturier. JST: An object synchronization aspect for Java. In Lopes et al. [924].
- [1265] Lionel Seinturier, editor. *2nd French Workshop on Aspect-Oriented Software Development (JFDLPA 2005)*, September 2005.
- [1266] Linda Seiter. *Design Patterns for Managing Evolution*. PhD thesis, Northeastern University, 1996.
- [1267] Linda Seiter, Mira Mezini, and Karl Lieberherr. Dynamic component gluing. In Ulrich Eisenecker and Krzysztof Czarnecki, editors, *First Int'l Symposium on Generative and Component-Based Software Engineering*. Springer, 1999.
- [1268] Linda Seiter, Mira Mezini, and Karl Lieberherr. Dynamic component gluing. In Ossher et al. [1091].
- [1269] Linda M. Seiter, Jens Palsberg, and Karl J. Lieberherr. Evolution of Object Behavior using Context Relations. In David Garlan, editor, *Symposium on Foundations of Software Engineering, SIGSOFT*, pages 46–57. ACM Press (SIGSOFT), 1996.
- [1270] Linda M. Seiter, Jens Palsberg, and Karl J. Lieberherr. Evolution of object behavior using context relations. *IEEE Trans. Soft. Eng.*, 24(1):79–92, January 1998.
- [1271] Damien Sereni and Oege de Moor. Static analysis of aspects. In Akşit [26], pages 30–39.
- [1272] Viren Shah. ‘Commercializing AOP: Outsourcing the stable doors’ and ‘Software security: AOSD as the savior’. In Bodkin et al. [185].
- [1273] Viren Shah and Frank Hill. An aspect-oriented security framework: Lessons learned. In De Win et al. [413].
- [1274] David Shepherd. The fast prototyping and evaluation of aspect mining analyses via timna. In Tourwé et al. [1398].
- [1275] David Shepherd and Lori Pollock. Interfaces, aspects, and views. In Tourwé et al. [1399].
- [1276] David Shepherd, Lori Pollock, and Tom Tourwé. Using language clues to discover crosscutting concerns. In Robillard [1212], pages 1–6.

- [1277] Mati Shomrat and Amiram Yehudai. Obvious or not? Regulating architectural decisions using aspect-oriented programming. In Kiczales [781], pages 3–9.
- [1278] Macneil Shonle, Karl Lieberherr, and Ankit Shah. XAspects: An extensible system for domain-specific aspect languages. In *Companion of the 18th annual ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications*, pages 28–37. ACM Press, 2003.
- [1279] M. Sicilia and E. Garcia. On imperfection in information as an “early” crosscutting concern and its mapping to aspect-oriented design. In Tekinerdoğan et al. [1378], pages 91–96.
- [1280] Marcelo Sihman and Shmuel Katz. A calculus of superimpositions for distributed systems. In Kiczales [781], pages 28–40.
- [1281] Marcelo Sihman and Shmuel Katz. Model checking applications of aspects and superimpositions. In Leavens and Clifton [866].
- [1282] Marcelo Sihman and Shmuel Katz. Superimpositions and aspect-oriented programming. *The Computer Journal*, 46(5):529–541, September 2003.
- [1283] Antonio Rito Silva. Separation and composition of overlapping and interacting concerns. In Ossher et al. [1091].
- [1284] Clayton Silva and Nabor das Chagas Mendonça. Uma abordagem para integrar aspectos e serviços web. In Garcia et al. [549].
- [1285] Ignacio Silva-Lepe. Abstracting graph-based specifications of object-oriented programs. In *ACM Computer Science Conference, Symposium on Applied Computing*, pages 447–451. ACM, 1994.
- [1286] D. Simmonds, S. Ghosh, and R. France. An aspect-oriented model driven architectural framework for middleware transparency. In Araújo et al. [81].
- [1287] Ian Simmonds. Clues in the search for ever more valuable separations of concern. In Tarr et al. [1363].
- [1288] Ian Simmonds and David Ing. Clues in the search for ever more valuable separations of concern. In Tarr et al. [1362].
- [1289] Henny B. Sipma. A formal model for crosscutting modular transition systems. In Leavens and Clifton [866].
- [1290] Harvey Siy, Mansour Zand, and Victor Winter. The role of aspects in domain engineering. In Araújo et al. [79].
- [1291] Mark Skipper. The Watson subject compiler and AspectJ (A critique of practical objects). In Ossher et al. [1091].
- [1292] Mark Skipper. A model of composition oriented programming. In Tarr et al. [1363].
- [1293] Mark Skipper. Semantics of an object-oriented language with aspects and advice. In Bergmans et al. [154].
- [1294] Therapon Skotiniotis, Karl Lieberherr, and David Lorenz. Aspect instances and their interactions. In Bergmans et al. [153].
- [1295] Pawel Slowikowski and Krzysztof Zieliński. Comparison study of aspect-oriented and container managed security. In Hannemann et al. [640].
- [1296] Yannis Smaragdakis and Don Batory. Mixin layers: An object-oriented implementation technique for refinements and collaboration-based designs. *ACM Transactions on Software Engineering and Methodology (TOSEM)*, 11(2):215–255, 2002.
- [1297] Adel Smeda, Tahar Khammaci, and Mourad Oussalah. Software connectors in COSA approach. In Lorenz and Coady [932].
- [1298] Douglas R. Smith. A generative approach to aspect-oriented programming. In Karsai and Visser [745], pages 39–54.
- [1299] Rik Smoody. Aspects can be objects, too. In Chu-Carroll et al. [286].
- [1300] Gregor Snelting and Frank Tip. Theory and formal techniques semantics-based composition of class hierarchies. In Magnusson [944], pages 562–584.
- [1301] S. Soares and P. Borba. PaDA: A pattern for distribution aspects. In *Second Latin American Conference on Pattern Languages of Programming — SugarLoafPLOP*, August 2002.

- [1302] Sérgio Soares and Paulo Borba. Concurrency control with Java and relational databases. In *Proceedings of V Brazilian Symposium on Programming Languages, SBLP 2001*, pages 252–267, May 2001. (in Portuguese).
- [1303] Sérgio Soares and Paulo Borba. Concurrency manager. In *First Latin American Conference on Pattern Languages Programming, SugarLoafPLoP 2001*, October 2001. Published in UERJ Magazine: Special Issue on Software Patterns, June 2002, pages 221–231.
- [1304] Sérgio Soares and Paulo Borba. AspectJ—Aspect-oriented programming in Java. In *Tutorial in Proceedings of VI Brazilian Symposium on Programming Languages, SBLP 2002*, pages 39–55, June 2002. (in Portuguese).
- [1305] Sérgio Soares and Paulo Borba. Concurrency control with Java and relational databases. In *Proceeding of 26th Annual IEEE Computer Software and Applications Conference, COMPSAC 2002*, pages 843–849, August 2002.
- [1306] Sérgio Soares and Paulo Borba. PIP: Progressive implementation pattern. In *1st Workshop on Software Development Process Patterns (OOPSLA02)*, November 2002.
- [1307] Sérgio Soares, Eduardo Laureano, and Paulo Borba. Implementing distribution and persistence aspects with AspectJ. In *Proceedings of the 17th ACM conference on Object-oriented programming, systems, languages, and applications*, pages 174–190. ACM Press, 2002.
- [1308] Dennis Sosnoski. Classworking toolkit: Performance tracing with aspects. Technical report, IBM Developer Works, April 2005.
- [1309] Dennis Sosnoski. Classworking toolkit: Putting aspects to work. Technical report, IBM Developer Works, March 2005.
- [1310] SOUL homepage.  
<http://prog.vub.ac.be/research/DMP/soul/>.
- [1311] G. Sousa, S. Soares, P. Borba, and J. Castro. Separation of crosscutting concerns from requirements to design: Adapting the use case driven approach. In Tekinerdoğan et al. [1378], pages 97–106.
- [1312] A. Speck, M. Clauss, and B. Franczyk. Concerns of variability in bottom-up product-lines. In Costanza et al. [371]. Technical report IAI-TR-2002-1.
- [1313] Andreas Speck, Elke Pulvermüller, and Mira Mezini. Reusability of concerns. In Lopes et al. [913].
- [1314] Eva Spies, Julia Rüger, and Ana Moreira. Using i\* to identify candidate aspects. In Aldawud et al. [38].
- [1315] Olaf Spinczyk, Andreas Gal, and Wolfgang Schröder-Preikschat. AspectC++: An aspect-oriented extension to the C++ programming language. In *Proceedings of the Fortieth International Conference on Tools Pacific*, pages 53–60. Australian Computer Society, Inc., 2002.
- [1316] V. Spurlin. Aspect-oriented programming with sun ONE studio. Technical report, Sun Microsystems, October 2002.
- [1317] Owen Stampflee, Celina Gibbs, and Yvonne Coady. RADAR: Really low level aspects for dynamic analysis and reasoning. In Olaf Spinczyk, Andreas Gal, and Michael Schoettner, editors, *ECOOP Workshop on Programming Languages and Operating Systems*, June 2004.
- [1318] John Stankovic, Ruiqing Zhu, Ram Poor-nalingam, Chenyang Lu, Zhendong Yu, Marty Humphrey, and Brian Ellis. VEST: An aspect-based composition tool for real-time systems. In *Real-Time Applications Symposium*, pages 58–69, May 2003.
- [1319] Erlend Stav. Extending a visual tool. A challenging problem for extending object-orientation. In Lopes et al. [913].
- [1320] Friedrich Steimann. Why most domain models are aspect free. In Aldawud et al. [38].
- [1321] D. Stein, S. Hanenberg, and R. Unland. Modeling pointcuts. In Tekinerdoğan et al. [1378], pages 107–110.
- [1322] Dominik Stein and Stefan Hanenberg. Why aspect-oriented software development and model-driven development are not the same? In Aksit et al. [35].

- [1323] Dominik Stein, Stefan Hanenberg, and Rainer Unland. Designing aspect-oriented crosscutting in UML. In Aldawud et al. [37].
- [1324] Dominik Stein, Stefan Hanenberg, and Rainer Unland. On representing join points in the UML. In Kandé et al. [735].
- [1325] Dominik Stein, Stefan Hanenberg, and Rainer Unland. An UML-based aspect-oriented design notation. In Kiczales [781], pages 106–112.
- [1326] Dominik Stein, Stefan Hanenberg, and Rainer Unland. Position paper on aspect-oriented modeling: Issues on representing crosscutting features. In Aldawud et al. [43].
- [1327] Dominik Stein, Stefan Hanenberg, and Rainer Unland. Visualizing join point selections for stateful aspects. In Gybels et al. [615].
- [1328] Peter Sties and Wolfgang Kellerer. A generic and implementation independent service description model. In Choukair [280], pages 163–168.
- [1329] Maximilian Stoerzer and Stefan Hanenberg. A classification of pointcut language constructs. In Bergmans et al. [156].
- [1330] Zoran Stojanovic and Ajantha Dahanayak. Components and viewpoints as integrated separations of concerns in system designing. In Glandrup et al. [564].
- [1331] Maximilian Störzer. Analysis of AspectJ programs. In Bachmendo et al. [97].
- [1332] Maximilian Störzer and Jens Krinke. Interference analysis for AspectJ. In Leavens and Clifton [866].
- [1333] Maximilian Störzer, Jens Krinke, and Silvia Breu. Trace analysis for aspect application. In Hannemann et al. [640].
- [1334] R. Stroud and Z. Wue. Using metaobject protocols to satisfy non-functional requirements. In Chris Zimmermann, editor, *Advances in Object-Oriented Metalevel Architectures and Reflection*, pages 31–52. CRC Press, 1996.
- [1335] R. Suen and E. Baniassad. Isolating concerns in requirements using latent semantic analysis. In Baniassad et al. [109].
- [1336] G. T. Sullivan. Aspect-oriented programming using reflection and meta-object protocols. *Comm. ACM*, 44(10):95–97, October 2001.
- [1337] Gregory T. Sullivan. Aspect-oriented programming using reflection. In De Volder et al. [407].
- [1338] Kevin Sullivan, Lin Gu, and Yuanfang Cai. Non-modularity in aspect-oriented languages: Integration as a crosscutting concern for AspectJ. In Kiczales [781], pages 19–27.
- [1339] Thanwadee Sunetnanta and Anthony Finkelstein. Automated consistency checking for multiperspective software specifications. In Tarr and Ossher [1366].
- [1340] John Sung. Aspectual concepts. Master’s thesis, Northeastern University, June 2002.
- [1341] S. Sutton, Jr. Concerns in a requirements model - a small case study. In Araújo et al. [81].
- [1342] S. M. Sutton, Jr. Early-stage concern modeling. In Rashid et al. [1191].
- [1343] S. M. Sutton, Jr. and I. Rouvellou. Advanced separation of concerns for component evolution. In *Workshop on Engineering Complex Object-Oriented Systems for Evolution (ECOOSE) (OOPSLA)*, October 2001.
- [1344] Stanley Sutton, Jr. and Isabelle Rouvellou. Modeling of software concerns in Cosmos. In Kiczales [781], pages 127–133.
- [1345] Stanley M. Sutton, Jr. Multiple dimensions of concern in software testing. In Ossher et al. [1091].
- [1346] Stanley M. Sutton, Jr. and Isabelle Rouvellou. Concerns in the design of a software cache. In Tarr et al. [1362].
- [1347] Stanley M. Sutton, Jr. and Isabelle Rouvellou. Applicability of categorization theory to multidimensional separation of concerns. In De Volder et al. [407].
- [1348] Stanley M. Sutton, Jr. and Isabelle Rouvellou. Issues in the design and implementation of a concern-space modeling schema. In Tarr and Ossher [1366].

- [1349] Stanley M. Sutton, Jr. and Isabelle Rouvelou. Concern modeling for aspect-oriented software development. In Filman et al. [513], pages 479–505.
- [1350] Stanley M. Sutton, Jr. and Peri Tarr. Aspect-oriented design needs concern modeling. In Glandrup et al. [564].
- [1351] Davy Suvée, Bruno De Fraine, and Wim Vanderperren. FuseJ: An architectural description language for unifying aspects and components. In Bergmans et al. [156].
- [1352] Davy Suvée and Wim Vanderperren. JAsCo: An aspect-oriented approach tailored for component based software development. In Akşit [26], pages 21–29.
- [1353] Davy Suvée, Wim Vanderperren, Dennis Wagelaar, and Viviane Jonckers. There are no aspects. In Uwe Aßmann and Adrian Pop, editors, *Software Composition Workshop (ETAPS)*, pages 142–162, March 2004.
- [1354] Junichi Suzuki and Yoshikazu Yamamoto. Extending UML with aspects: Aspect support in the design phase. In Lopes et al. [924].
- [1355] Kedar Swadi, editor. *Second MetaOCaml Workshop (GPCE05)*, September 2005.
- [1356] É. Tanter and P. Ebraert. A flexible approach to interactive runtime inspection. In Filman et al. [501].
- [1357] Éric Tanter. Metalevel facilities for multi-language AOP. In Gybels et al. [615].
- [1358] Éric Tanter and Jacques Noyé. Motivation and requirements for a versatile AOP kernel. In Gybels et al. [616].
- [1359] Éric Tanter and Jacques Noyé. A versatile kernel for multi-language AOP. In Glück and Lowry [565], pages 173–188.
- [1360] P. Tarr and H. Ossher. Hyper/J user and installation manual. Technical report, IBM T. J. Watson Research Center, 2000.
- [1361] Peri Tarr, editor. *Proc. 4rd Int’ Conf. on Aspect-Oriented Software Development (AOSD-2005)*. ACM Press, March 2005.
- [1362] Peri Tarr, Lodewijk Bergmans, Martin Griss, and Harold Ossher, editors. *Workshop on Advanced Separation of Concerns (OOPSLA 2000)*, October 2000.
- [1363] Peri Tarr, Anthony Finkelstein, William Harrison, Bashar Nuseibeh, Harold Ossher, and Dewayne Perry, editors. *Workshop on Multi-Dimensional Separation of Concerns in Software Engineering (ICSE 2000)*, June 2000.
- [1364] Peri Tarr, William Harrison, Harold Ossher, Anthony Finkelstein, Bashar Nuseibeh, and Dewayne Perry. Workshop summary. In Tarr et al. [1363].
- [1365] Peri Tarr, William Harrison, Harold Ossher, Anthony Finkelstein, Bashar Nuseibeh, and Dewayne Perry. Workshop on multi-dimensional separation of concerns in software engineering. *ACM SIGSOFT Software Engineering Notes*, 26(1):78–81, 2001.
- [1366] Peri Tarr and Harold Ossher, editors. *Workshop on Advanced Separation of Concerns in Software Engineering (ICSE 2001)*, May 2001.
- [1367] Peri Tarr, Harold Ossher, William Harrison, and Stanley M. Sutton, Jr. N degrees of separation: Multi-dimensional separation of concerns. In *Proc. 21st Int’l Conf. Software Engineering (ICSE’1999)*, pages 107 – 119. IEEE Computer Society Press, May 1999.
- [1368] Peri Tarr, Harold Ossher, and Johannes Henkel. Visualization as an aid to compositional software engineering. In De Volder et al. [407].
- [1369] Peri Tarr, Harold Ossher, Stanley M. Sutton, Jr., and William Harrison. N degrees of separation: Multi-dimensional separation of concerns. In Filman et al. [513], pages 37–61.
- [1370] M. Tatsubori. Separation of distribution concerns in distributed Java programming. In *OOPSLA’01 Doctoral Symposium*, October 2001.
- [1371] Hideaki Tatsuzawa, Hidehiko Masuhara, and Akinori Yonezawa. Aspectual caml: an aspect-oriented functional language. In *International Conference on Functional Programming (ICFP)*, pages 320–330. ACM, September 2005.
- [1372] Hideaki Tatsuzawa, Hidehiko Masuhara, and Akinori Yonezawa. Aspectual caml: An aspect-oriented functional language. In Leavens et al. [867].

- [1373] B. Tekinerdoğan. ASAAM: Aspectual software architecture analysis method. In Araújo et al. [81].
- [1374] B. Tekinerdoğan and M. Akşit. Adaptability in object-oriented development. In M. Muhlhauser, editor, *Special Issues in Object-Oriented Programming*, pages 7–11, 1996.
- [1375] B. Tekinerdoğan and M. Akşit. Deriving design aspects from canonical models. In Lopes et al. [914].
- [1376] B. Tekinerdoğan, Ana Moreira, J. Araújo, and P. Clements. Early aspects: Aspect-oriented requirements engineering and architecture design—Workshop report. In Tekinerdoğan et al. [1378], pages 3–14.
- [1377] Bedir Tekinerdoğan and Mehmet Akşit. Separation and composition of concerns through synthesis-based design. In Tarr et al. [1362].
- [1378] Bedir Tekinerdoğan, Paul Clements, Ana Moreira, and João Araújo, editors. *Early Aspects: Aspect-Oriented Requirements Engineering and Architecture Design*, March 2004.
- [1379] Aleksandra Tesanovic, Mehdi Amirijoo, Mikael Björk, and Jörgen Hansson. Empowering configurable QoS management in real-time systems. In Tarr [1361], pages 39–50.
- [1380] Aleksandra Tesanovic, Jörgen Hansson, Dag Nyström, and Christer Norström. Integrating symbolic worst-case execution time analysis with aspect-oriented system development. In Chu-Carroll et al. [286].
- [1381] Aleksandra Tesanovic, Peng Mu, and Jörgen Hansson. Development environment for configuration and analysis of embedded and real-time systems. In Coady et al. [328].
- [1382] Francis Tessier, Linda Badri, and Mourad Badri. Towards a formal detection of semantic conflicts between aspects: A model-based approach. In Aldawud et al. [38].
- [1383] Francis Tessier, Mourad Badri, and Linda Badri. A model-based detection of conflicts between crosscutting concerns: Towards a formal approach. In Huang et al. [701].
- [1384] A. Tešanovic. Towards aspectual component-based real-time system development. Master’s thesis, Linköping University, June 2003.
- [1385] A. Tešanovic, D. Nyström, J. Hansson, C. Norström, and P. Uhlin. Aspect-level WCET analyzer: A tool for automated WCET analysis of a real-time software composed using aspects and components. In *3rd International Workshop on Worst-Case Execution Time Analysis (WCET 2003)*, July 2003.
- [1386] A. Tešanovic, K. Sheng, and J. Hansson. Application-tailored database systems: A case of aspects in an embedded database. In *Proceedings of the 8th International Database Engineering and Applications Symposium (IDEAS’04)*. IEEE Computer Society, July 2004.
- [1387] Aleksandra Tešanovic, Dag Nyström, Jörgen Hansson, and Christer Norström. Aspect-level worst-case execution time analysis of real-time systems compositioned using aspects and components. In *Proceedings of the 27th IFAC/IFIP/IEEE Workshop on Real-Time Programming (WRTP’03)*. Elsevier, May 2003.
- [1388] Aleksandra Tešanovic, Dag Nyström, Jörgen Hansson, and Christer Norström. Towards aspectual component-based development of real-time systems. In *Proceeding of the 9th International Conference on Real-Time and Embedded Computing Systems and Applications (RTCSA 2003)*. Springer-Verlag, February 2003.
- [1389] Aleksandra Tešanovic, Dag Nyström, Jörgen Hansson, and Christer Norström. Aspects and components in real-time system development: Towards reconfigurable and reusable software. *Journal of Embedded Computing*, February 2004.
- [1390] Craig Thompson, Paul Pazandak, Venu Vasudevan, Frank Manola, Mark Palmer, Gil Hansen, and Tom Bannon. Intermediary architecture: Interposing middleware object services between web client and server. *ACM Computing Surveys*, 31(2es):14, 1999.
- [1391] Kresten Krab Thorup. Contextual class extensions. In Lopes et al. [918].

- [1392] Maria Tkatchenko and Gregor Kiczales. Uniform support for modeling crosscutting structure. In Kandé et al. [737].
- [1393] M. Tokoro and R. Pareschi, editors. *Proc. 8th European Conf. Object-Oriented Programming*. Springer Verlag LNCS 821, July 1994.
- [1394] Søren Top, Bo Jørgensen, Claus Thybo, and Peter Thusgaard. Meta-level architectures for fault tolerant control (FTC) in embedded software systems. In Bergmans et al. [154].
- [1395] Soren Top, Bo Nørregaard Jørgensen, and Christo Angelov. Separation of fault tolerant control concern in embedded control system. In Glandrup et al. [564].
- [1396] Björn Törnqvist and Rune Gustavsson. On adaptive aspect-oriented coordination for critical infrastructures. In Canal et al. [238].
- [1397] Tom Tourwé, Johan Brichau, and Kris Gybels. On the existence of the AOSD-evolution paradox. In Bergmans et al. [153].
- [1398] Tom Tourwé, Magiel Bruntink, Marius Marin, and David Shepherd, editors. *Workshop on Aspect Reverse Engineering (WARE)*, November 2004.
- [1399] Tom Tourwé, Andy Kellens, Mariano Ceccato, and David Shepherd, editors. *Linking Aspect Technology and Evolution*, March 2005.
- [1400] Tom Tourwé, Andy Kellens, Wim Vanderperren, and Frederik Vannieuwenhuyse. Inductively generated pointcuts to support refactoring to aspects. In Bergmans et al. [155].
- [1401] E. Truyen, B. Vanhaute, W. Joosen, P. Verbaeten, and B. Nørregaard Jørgensen. A dynamic customization model for distributed component-based systems. In Choukair [280], pages 147–156.
- [1402] Eddy Truyen, Wouter Joosen, Bo Nørregaard Jørgensen, and Pierre Verbaeten. A generalization and solution to the common ancestor dilemma problem in delegation-based object systems. In Filman et al. [502], pages 103–119.
- [1403] Eddy Truyen, Wouter Joosen, and Pierre Verbaeten. Run-time support for aspects in distributed system infrastructure. In Coady et al. [327].
- [1404] Eddy Truyen, Bo Nørregaard Jørgensen, Wouter Joosen, and Pierre Verbaeten. Aspects for run-time component integration. In Lopes et al. [913].
- [1405] Eddy Truyen, Bo Nørregaard Jørgensen, and Wouter Joosen. Customization of object request brokers through dynamic reconfiguration. In *Proc. of Tools Europe 2000*, June 2000.
- [1406] Eddy Truyen, Bart Vanhaute, Wouter Joosen, Pierre Verbaeten, and Bo Nørregaard Jørgensen. Customization of on-line services with simultaneous client-specific views. In Bergmans et al. [154].
- [1407] Eddy Truyen, Bart Vanhaute, Wouter Joosen, Pierre Verbaeten, and Bo Nørregaard Jørgensen. Customization of on-line services with simultaneous client-specific views. In Tarr and Ossher [1366].
- [1408] Eddy Truyen, Bart Vanhaute, Wouter Joosen, Pierre Verbaeten, and Bo Nørregaard Jørgensen. Dynamic and selective combination of extensions in component-based applications. In *Proc. 23rd Int'l Conf. Software Engineering (ICSE'2001)*, May 2001.
- [1409] David B. Tucker and Shriram Krishnamurthi. Pointcuts and advice in higher-order languages. In Akşit [26], pages 158–167.
- [1410] Naoyasu Ubayashi, Hidehiko Masuhara, and Tetsuo Tamai. An AOP implementation framework for extending join point models. In Cazzola et al. [255], pages 71–82.
- [1411] Naoyasu Ubayashi and Tetsuo Tamai. Separation of concerns in mobile agent applications. In Yonezawa and Matsuoka [1494], pages 89–109.
- [1412] Naoyasu Ubayashi and Tetsuo Tamai. Aspect oriented programming with model checking. In Kiczales [781], pages 148–154.
- [1413] Naoyasu Ubayashi and Tetsuo Tamai. Concern management for constructing model compilers. In Robillard [1212], pages 1–5.

- [1414] Naoyasu Ubayashi, Tetsuo Tamai, Shinji Sano, Yusaku Maeno, and Satoshi Murakami. Model compiler construction based on aspect-oriented mechanisms. In Glück and Lowry [565], pages 109–124.
- [1415] David Ungar. The limits to factoring. In Ossher et al. [1091].
- [1416] Naval Vaidya and R. K. Joshi. Implementation of filter configurations using method call pointcuts in AspectJ. In Huang et al. [701].
- [1417] Marco Tulio Valente, Rodrigo Palhares, and Fabio Tirelo. Especificação de táticas para invocação remota de métodos usando orientação por aspectos. In Garcia et al. [549].
- [1418] Klaas van den Berg and José María Conejero. Disentangling crosscutting in AOSD: A conceptual framework. In Gybels et al. [615].
- [1419] Arie van Deursen, Marius Marin, and Leon Moonen. AJHotDraw: A showcase for refactoring to aspects. In Tourwé et al. [1399].
- [1420] Peter van Roy, Seif Haridi, Per Brand, Gert Smolka, Michael Mehl, and Ralf Scheidhauer. Using mobility to make transparent distribution practical. In Lopes et al. [918].
- [1421] Koenraad Vandenborre, Muna Matar, and Ghislain Hoffman. Orthogonal persistence using aspect oriented programming. In Coady et al. [327].
- [1422] Glenn Vanderburg. Position paper. In *OOP-SLA 2001 Software Archaeology Workshop*, October 2001.
- [1423] Wim Vanderperren. A pattern based approach to separate tangled concerns in component based development. In Coady et al. [327].
- [1424] Wim Vanderperren and Davy Suvée. JAsCoAP: Adaptive programming for component-based software engineering. In Lorenz and Coady [932].
- [1425] Wim Vanderperren and Davy Suvée. Optimizing JAsCo dynamic AOP through HotSwap and Jutta. In Filman et al. [502], pages 120–134.
- [1426] Wim Vanderperren, Davy Suvée, and Viviane Jonckers. Invasive composition adapters: An aspect-oriented approach for visual component-based development. In Coady et al. [326].
- [1427] Wim Vanderperren, Davy Suvée, Bart Verheecke, María Agustina Cibrán, and Viviane Jonckers. Adaptive programming in JAsCo. In Tarr [1361], pages 75–86.
- [1428] Bart Vanhaute, Bart De Win, and Bart De Decker. Building frameworks in AspectJ. In Bergmans et al. [154].
- [1429] Bart Vanhaute, Eddy Truyen, Wouter Joosen, and Pierre Verbaeten. Composing non-orthogonal meta-programs. In Ossher et al. [1091].
- [1430] Aditya Varma and T. V. Prabhakar. Dependency issues in aspect weaving. In Filman et al. [515], pages 63–69.
- [1431] Alexandre Vasseur. Dynamic AOP and runtime weaving for Java—How does AspectWerkz address it? In Filman et al. [502], pages 135–145.
- [1432] Matthias Veit and Stephan Herrmann. Model-view-controller and object teams: A perfect match of paradigms. In Akşit [26], pages 140–149.
- [1433] Tine Verhanneman, Frank Piessens, Bart De Win, and Wouter Joosen. View connectors for the integration of domain specific access control. In De Win et al. [413].
- [1434] Bart Verheecke and María Agustina Cibrán. Dynamic aspects for web service management. In Filman et al. [502], pages 146–152.
- [1435] Bart Verheecke and María Agustina Cibrán. Dynamic aspects in large scale distributed applications—an experience report. In Bergmans et al. [156].
- [1436] John Viega. Separation of concerns for security. In Tarr et al. [1363].
- [1437] Luis Vinuesa and Francisco Ortín. Tejedor dinámico de aspectos sobre la plataforma .NET. In Fuentes et al. [538].
- [1438] D. Vollmann. Visibility of join-points in AOP and implementation languages. In Costanza et al. [371]. Technical report IAI-TR-2002-1.

- [1439] Christina von Flach G. Chavez and Carlos J. P. de Lucena. Design-level support for aspect-oriented software development. In De Volder et al. [407].
- [1440] Valentino Vranić. AspectJ paradigm model: A basis for multi-paradigm design for AspectJ. In Bosch [195], pages 48–57.
- [1441] D. Wagelaar. A concept-based approach for early aspect modelling. In Araújo et al. [81].
- [1442] Dennis Wagelaar and Lodewijk Bergmans. Using a concept-based approach to aspect-oriented software design. In Glandrup et al. [564].
- [1443] David Walker, Steve Zdancewic, and Jay Ligatti. A theory of aspects. In *Proceedings of the ACM SIGPLAN International Conference on Functional Programming*, August 2003.
- [1444] R. J. Walker, E. L. A. Baniassad, and G. Murphy. Assessing aspect-oriented programming and design. In Lopes et al. [914].
- [1445] Robert Walker and Kevin Viggers. Implementing protocols via declarative event patterns. In FSE04 [535], pages 159–169.
- [1446] Robert J. Walker. Supporting inconsistent world views. In Bergmans et al. [153].
- [1447] Robert J. Walker, Elisa L. A. Baniassad, and Gail C. Murphy. An initial assessment of aspect-oriented programming. In *Proc. 21st Int'l Conf. Software Engineering (ICSE '99)*, pages 120–130, 1999.
- [1448] Robert J. Walker, Elisa L. A. Baniassad, and Gail C. Murphy. An initial assessment of aspect-oriented programming. In Filman et al. [513], pages 531–556.
- [1449] Robert J. Walker and Gail C. Murphy. Join-points as ordered events: Towards applying implicit context to aspect-orientation. In Tarr and Ossher [1366].
- [1450] S. Walton and E. Eide. Feature types in early aspects. In Baniassad et al. [109].
- [1451] M. Wand. A semantics for advice and dynamic join points in aspect-oriented programming. In *Semantics, Applications and Implementation of Program Generation (SAIG)(PLI)*, volume 2196 of *lncs*, page 45, Berlin, September 2001. Springer-Verlag.
- [1452] M. Wand, G. Kiczales, and C. Dutchn. A semantics for advice and dynamic join points in aspect-oriented programming. *ACM Transactions on Programming Languages and Systems*, 26(5):890–910, September 2004.
- [1453] Mitchell Wand. Understanding aspects: Extended abstract. In *Proceedings of the Eighth ACM SIGPLAN International Conference on Functional Programming*, pages 299–300. ACM Press, 2003.
- [1454] Mitchell Wand, Gregor Kiczales, and Chris Dutchn. A semantics for advice and dynamic join points in aspect-oriented programming. In Cytron and Leavens [379], pages 1–8.
- [1455] Lin-Chuan Wang, Xue-Fei Tang, and Li-Jia Zhang. Aspect-oriented: a candidate for the biologically-inspired programming paradigm for neural networks and evolvable software. In Huang et al. [701].
- [1456] Matthew Webster. A dynamic join point model for Java object lifecycle. In Filman et al. [502], pages 153–155.
- [1457] Hans Wegener and Ahmed Rida. Reengineering of metalevel abstractions with data mining methods. In Tarr et al. [1362].
- [1458] Ian Welch and Robert Stroud. Load-time application of aspects to Java COTS software. In Lopes et al. [924].
- [1459] Ian Welch, Robert J. Stroud, and Alexander Romanovsky. Aspects of exceptions at the meta-level. In Yonezawa and Matsuoka [1494], pages 280–282.
- [1460] Ian S. Welch and Robert J. Stroud. Security and aspects: A metaobject protocol viewpoint. In Coady et al. [327].
- [1461] Ian S. Welch and Robert J. Stroud. Re-engineering security as a crosscutting concern. *The Computer Journal*, 46(5):578–589, September 2003.
- [1462] Ian S. Welch, Robert J. Stroud, and Alexander Romanovsky. Aspects of exceptions at the meta-level. In Rashid [1181].
- [1463] M. Wermelinger, J. L. Fiadeiro, L. Andrade, G. Koutsoukos, and J. Gouveia. Separation

- of core concerns: Computation, coordination, and configuration. In De Volder et al. [407].
- [1464] J. Whittle and J. Araujo. Scenario modelling with aspects. *IEEE Proceedings Software*, 151(4):157–171, August 2004.
- [1465] Jon Whittle, João Araújo, and Dae-Kyoo Kim. Modeling and validating interaction aspects in UML. In Aldawud et al. [44].
- [1466] J. C. Wichman. The development of a pre-processor to facilitate composition filters in the Java language. Master’s thesis, University of Twente, 1999.
- [1467] Edward Willink and Vyacheslav Muchnick. Weaving a way past the C++ one definition rule. In Lopes et al. [924].
- [1468] Jan Wloka. Refactoring in the presence of aspects. In Martín et al. [957].
- [1469] Jan Wloka. Aspect-aware refactoring tool support. In Tourwé et al. [1399].
- [1470] Eric Wohlstadter and Prem Devanbu. A lazy approach to separating architectural concerns. In Tarr and Ossher [1366].
- [1471] Eric Wohlstadter, Aaron Keen, Stoney Jackson, and Premkumar Devanbu. Accommodating evolution in AspectJ. In De Volder et al. [407].
- [1472] Eric Wohlstadter, Stefan Tai, and Prem Devanbu. Two party aspect agreement using a COTS solver. In Coady et al. [328].
- [1473] Eric Wohlstadter, Brian Toone, and Prem Devanbu. A framework for flexible evolution in distributed heterogeneous systems. In *Proceedings of the Workshop on Principles of Software Evolution*, pages 39–42. ACM Press, 2002.
- [1474] Pengcheng Wu. How to compile aspects with Real-Time Java. In Leavens et al. [867].
- [1475] Pengcheng Wu, Shriram Krishnamurthi, and Karl Lieberherr. Traversing recursive object structures: The functional visitor in Demeter. In Bergmans et al. [153].
- [1476] Pengcheng Wu and Karl J. Lieberherr. Shadow programming: Reasoning about programs using lexical join point information. In Glück and Lowry [565], pages 141–156.
- [1477] Pengcheng Wu and Mitchell Wand. An empirical study of the Demeter system. In Bergmans et al. [155].
- [1478] Xiaoqing Wu, Barrett R. Bryant, Jeff Gray, and Marjan Mernik. Pattern transformation for two-dimensional separation of concerns. In Robillard [1212], pages 1–5.
- [1479] Egon Wuchner. Reflections on feature oriented software engineering. In Beuche et al. [164].
- [1480] Xerox Corporation. The AspectJ primer. <http://dev.eclipse.org/viewcvs/indextech.cgi/~checkout~/aspectj-home/doc/progguide/ch01.html>.
- [1481] Xerox Corporation. The AspectJ programming guide. <http://dev.eclipse.org/viewcvs/indextech.cgi/~checkout~/aspectj-home/doc/progguide/index.html>.
- [1482] Xerox PARC. AspectJ home page. <http://eclipse.org/aspectj/>.
- [1483] Cun Xiao. *Adaptive Software: Automatic Navigation Through Partially Specified Data Structures*. PhD thesis, Northeastern University, 1994.
- [1484] Yingfei Xiong and Feng Wan. CCC: An aspect-oriented intermediate language on .Net platform. In Huang et al. [701].
- [1485] Jia Xu, Hridesh Rajan, and Kevin Sullivan. Aspect reasoning by reduction to implicit invocation. In Clifton et al. [321], pages 31–36.
- [1486] L. Xu, H. Ziv, D. Richardson, and Z. Liu. Towards modeling non-functional requirements in software architecture. In Clements et al. [320].
- [1487] Kiyoshi Yamada and Takuo Watanabe. An aspect-oriented approach to modular behavioural specification. In Aksit et al. [35].
- [1488] Kiyoshi Yamada and Takuo Watanabe. Moxa: An aspect-oriented approach to modular behavioral specifications. In Bergmans et al. [156].
- [1489] Susumu Yamazaki, Michihiro Matsumoto, Tsuneo Nakanishi, Teruaki Kitasuka, and Akira Fukuda. Aspect-oriented design and implementation in Java bytecode analyzer framework. In Lorenz and Coady [932].

- [1490] Yoshisato Yanagisawa, Shigeru Chiba, and Kenichi Kourai. A source-level kernel profiler based on dynamic aspect orientation. In Filman et al. [515], pages 41–45.
- [1491] Zhen Yao, Qi long Zheng, and Guo-Liang Chen. AOP++: A generic aspect-oriented programming framework in C++. In Glück and Lowry [565], pages 94–108.
- [1492] Annie T. T. Ying, Gail C. Murphy, Raymond T. Ng, and Mark C. Chu-Carroll. Using version information for concern interface and code-assist. In Chu-Carroll et al. [286].
- [1493] Annie T. T. Ying, James L. Wright, and Steven Abrams. An exploration of how comments are used for marking related code fragments. In Robillard [1212], pages 1–4.
- [1494] A. Yonezawa and S. Matsuoka, editors. *Metalevel Architectures and Separation of Crosscutting Concerns 3rd Int'l Conf. (Reflection 2001)*, LNCS 2192. Springer-Verlag, September 2001.
- [1495] Yang Yu and Jörg Kienzle. Towards an efficient aspect precedence model. In Filman et al. [502], pages 156–167.
- [1496] Yijun Yu, J. C. Sampai do Prado Leite, and J Mylopoulos. From goals to aspects: Discovering aspects from requirements goal models. In *Proc. 12th Int'l Requirements Engineering Conference*, pages 33–42. IEEE, September 2004.
- [1497] A. Zaidman and S. Demeyer. Using a variant of sliding window to reduce event trace data. In Filman et al. [501].
- [1498] Andy Zaidman, Toon Calders, Serge Demeyer, and Jan Paredaens. Selective introduction of aspects for program comprehension. In Tourwé et al. [1398].
- [1499] Aida Atef Zakaria and Hoda Hosny. Metrics for aspect-oriented software design. In Aldawud et al. [43].
- [1500] Aida Atef Zakaria, Hoda Hosny, and Amir Zeid. A UML extension for modeling aspect-oriented systems. In Kandé et al. [735].
- [1501] U. Zdun. Pattern language for the design of aspect languages and aspect composition frameworks. *IEE Proceedings Software*, 151(2):67–83, April 2004.
- [1502] Uwe Zdun. Using architecture recovery and tracing patterns for aspect composition. In Bachmendo et al. [97].
- [1503] C. Zhang, H. A. Jacobsen, and Y. Yu. Linking goals to aspects. In Clements et al. [320].
- [1504] Charles Zhang, Gilbert Gao, and Hans-Arno Jacobsen. Towards just-in-time middleware architectures. In Tarr [1361], pages 63–74.
- [1505] Charles Zhang and H-A. Jacobsen. Quantifying aspects in middleware platforms. In Akşit [26], pages 130–139.
- [1506] Charles Zhang and Hans-Arno Jacobsen. TinyC<sup>2</sup>: Towards building a dynamic weaving aspect language for C. In Leavens and Clifton [866].
- [1507] Charles Zhang, Hans-Arno Jacobsen, Julie Waterhouse, and Adrian Colyer. Aspect refactoring verifier. In Tourwé et al. [1399].
- [1508] Gefei Zhang, Hubert Baumeister, Nora Koch, and Alexander Knapp. Aspect-oriented modeling of access control in web applications. In Kandé et al. [737].
- [1509] Hongyu Zhang, Stan Jarzabek, and Soe Myat Swe. X-frames approach for handling variants within concerns. In Tarr and Ossher [1366].
- [1510] Hongyu Zhang, Stan Jarzabek, and Soe Myat Swe. XVCL approach to separating concerns in product family assets. In Bosch [195], pages 36–47.
- [1511] Jing Zhang, Jeff Gray, and Yuehua Lin. A model-driven approach to enforce crosscutting assertion checking. In Robillard [1212], pages 1–5.
- [1512] Yan Zhang, Anna Liu, and Wei Qu. Implementing performance “tactics” using aspect-oriented programming. In Huang et al. [701].
- [1513] J. Zhao. Towards a metrics suite for aspect-oriented software. Technical Report SE-136-25, Information Processing Society of Japan (IPSJ), March 2002.
- [1514] J. Zhao. Unit testing for aspect-oriented programs. Technical Report SE-141-6, Information Processing Society of Japan (IPSJ), May 2003.

- [1515] J. Zhao and M. Rinard. Pipa: A behavioral interface specification language for AspectJ. In *Proc. Fundamental Approaches to Software Engineering (FASE'2003)*, LNCS 2621, pages 150–165. Springer-Verlag, April 2003.
- [1516] J. Zhao and M. Rinard. System dependence graph construction for aspect-oriented programs. Technical Report MIT-LCS-TR-891, Laboratory for Computer Science, MIT, March 2003.
- [1517] Jianjun Zhao. Change impact analysis for aspect-oriented software evolution. In *Proceedings of the Workshop on Principles of Software Evolution*, pages 108–112. ACM Press, 2002.
- [1518] Jianjun Zhao. Slicing aspect-oriented software. In *Proc. 10th IEEE International Workshop on Program Comprehension*, pages 251–260, June 2002.
- [1519] Jianjun Zhao. Tool support for unit testing of aspect-oriented software. In Chu-Carroll et al. [286].
- [1520] Jianjun Zhao. Data-flow-based unit testing of aspect-oriented programs. In *27th Annual International Computer Software and Applications Conference (COMPSAC)*, pages 188–197. IEEE Computer Society, November 2003.
- [1521] John Zinky, Joe Loyall, Partha Pal, Richard Shapiro, Richard Schantz, James Megquier, Michael Atighetchi, Craig Rodrigues, and David Karr. An AOP challenge problem: Managing QoS on interactions between distributed objects. In Lopes et al. [913].
- [1522] John Zinky and Richard Shapiro. The aspect-oriented interceptors' pattern for crosscutting and separation of concerns using conventional object oriented programming languages. In Coady et al. [326].
- [1523] John Zinky, Richard Shapiro, Joe Loyall, Partha Pal, and Michael Atighetchi. Separation of concerns for reuse of systemic adaptation in QuO 3.0. In Bergmans et al. [154].
- [1524] John A. Zinky, David E. Bakken, and Richard D. Schantz. Architectural support for quality of service for CORBA objects. *Theory and Practice of Object Systems*, 3(1):55–73, January 1997.
- [1525] David Zook, Shan Shan Huang, and Yannis Smaragdakis. Generating AspectJ programs with Meta-AspectJ. In Karsai and Visser [745], pages 1–18.