

Tiark Rompf

Lawson CS Building, 305 N. University Street, West Lafayette, IN-47907 USA
tiark@purdue.edu, tiarkrompf.github.io, +1 (765) 49-49027

October 1, 2020

Research

Applied and fundamental aspects of programming languages and compilers: language design, program analysis, program transformation, program generation and synthesis, type systems, semantics. PL technology in data management, machine learning, and AI.

Education

- **EPFL, Lausanne, Switzerland**
PhD, Computer Science (Advisor: Martin Odersky) **2012**
- **Universität zu Lübeck, Germany**
MSc (“Dipl. Inf.”), Computer Science/Medical Informatics (Advisor: Walter Dosch) **2008**
- **Universität Bremen, Germany**
BSc (“Vordiplom”), Computer Science **2003**

Experience

- **Purdue University:** Associate Professor (with Tenure), Computer Science **2020 - now**
- **Purdue Center for Programming Principles and Software Systems (PurPL):**
Co-Founder and Co-Director **2019 - now**
- **SambaNova Systems:** Scientific Advisor **2018 - now**
- **Purdue University:** Assistant Professor (Tenure Track), Computer Science **2014 - 2020**
- **Oracle Labs:** Principal Member of Technical Staff **2012 - 2014**
- **EPFL:** Post-Doctoral Researcher **2012 - 2014**
- **Typesafe:** Technical Advisor **2010 - 2012**
- **EPFL:** Research Assistant and PhD Candidate **2008 - 2012**
- **R&S Medizinsysteme:** Co-Founder **2004 - 2010**
- **Nachtlicht-Media:** Co-Founder **2000 - 2006**

Awards and Distinctions

- ACM SIGPLAN Programming Languages Software Award (Scala team) **2019**
- Facebook Probability and Programming Research Award **2019**
- Purdue Seed for Success Award **2019**
- Purdue College of Science Team Award **2019, 2020**
- VMware Systems Research Award **2018**
- Purdue College of Science Professional Achievement Award **2017**
- DOE Early Career Award **2017**
- Google Faculty Research Award **2017, 2018**

• NSF CAREER Award	2016
• OOPSLA Distinguished Artifact Award	2016
• GPCE Invited Keynote Speaker	2016
• Member of IFIP Working Group 2.11: Program Generation	2015
• VLDB Best Paper Award	2014
• Nominated for EPFL and SIGPLAN dissertation awards	2012
• CACM research highlight	2011
• EPFL PhD fellowship	2008

Professional Leadership and Service

Conference Organization

1. PurPL Fest & Midwest PL Summit 2019 (co-chair): Kick-off symposium for new PurPL center
2. GPCE 2018 (PC chair): Conference on Generative Programming: Concepts & Experiences
3. NII Shonan Meeting #113 (co-organizer): Meta-Programming for Statistical Machine Learning
4. DBPL 2017 (PC co-chair): Symposium on Database Programming Languages
5. PLDI 2016 (Sponsorship Chair): Conf. on Programming Language Design and Implementation
6. PEPM 2016 (PC co-chair): Workshop on Partial Evaluation and Program Manipulation
7. Summer School on Domain-Specific Languages 2015 (held at EPFL, co-organizer)
8. Midwest PL Summit 2015 (held at Purdue, first event of a series, lead organizer)
9. FHPC 2015 (PC co-chair): Workshop on Functional High-Performance Computing
10. WGP 2014 (PC co-chair): Workshop on Generic Programming
11. DSLDI 2013 (co-organizer): Workshop on Domain-Specific Languages Design and Impl.
12. JufoCongress 2003 & 2006 (co-organizer): Deutscher JungforscherCongress (German young researchers congress)

Program Committees

1. ICFP 2020 (ERC): International Conference on Functional Programming
2. ECOOP 2020 (ERC): European Conference on Object-Oriented Programming
3. PLDI 2020 (ERC): Conference on Programming Language Design and Implementation
4. POPL 2020: Symposium on Principles of Programming Languages
5. DISPA 2020: Workshop on Distributed Infrastructure, Systems, Programming and AI
6. DBPL 2019: Symposium on Database Programming Languages
7. ICFP 2019 (ERC): International Conference on Functional Programming
8. OOPSLA 2019: Conference on Object-Oriented Programming Languages and Systems
9. CC 2019: Conference on Compiler Construction
10. OOPSLA 2018 (ERC): Conference on Object-Oriented Programming Languages and Systems
11. VLDB 2018: Conference on Very Large Data Bases

12. PLDI 2018: Conference on Programming Language Design and Implementation
13. PEPM 2018: Workshop on Partial Evaluation and Program Manipulation
14. TFP 2018: Symposium on Trends in Functional Programming
15. ManLang 2018: Conference on Managed Languages & Runtimes
16. CADO 2018: HPCS Special Session on Compiler Architecture, Design and Optimization
17. DBTest 2018: Workshop on Testing Database Systems
18. GPCE 2017: Conference on Generative Programming: Concepts & Experiences
19. ECOOP 2017: European Conference on Object-Oriented Programming
20. META 2017: Workshop on Meta-Programming Techniques and Reflection
21. ManLang 2017: Conference on Managed Languages & Runtimes
22. Active 2017: Workshop on Data Management on Virtualized Active Systems
23. MoreVMs 2017: Workshop on Modern Language Runtimes, Ecosystems, and VMs
24. ICFP 2016 (ERC): Conference on Functional Programming
25. GPCE 2016: Conference on Generative Programming: Concepts & Experiences
26. DSLDI 2016: Workshop on Domain Specific Language Design and Implementation
27. ICOOLPS 2016: Workshop on Implementation, Compilation, Optimization of Object-Oriented Languages, Programs and Systems
28. POPL 2016: Symposium on Principles of Programming Languages
29. OOPSLA 2015: Conference on Object-Oriented Programming Languages and Systems
30. SCALA 2015: Symposium on the Scala Programming Language
31. GPCE 2015: Conference on Generative Programming: Concepts & Experiences
32. HLPP 2015: Symposium on High-level Parallel Programming and Applications
33. WOC 2015: Workshop on Continuations
34. PLDI 2015: Conference on Programming Language Design and Implementation
35. DSLDI 2014: Workshop on Domain Specific Language Design and Implementation
36. ICFP 2014: International Conference on Functional Programming
37. GPCE 2014: Conference on Generative Programming: Concepts & Experiences
38. PPDP 2014: Symposium on Principles and Practice of Declarative Programming
39. FHPC 2014: Workshop on Functional High-Performance Computing
40. HLPP 2014: Symposium on High-level Parallel Programming and Applications
41. HART 2014: Workshop on Haskell and Rewriting Techniques
42. PEPM 2014: Workshop on Partial Evaluation and Program Manipulation
43. WGP 2013: Workshop on Generic Programming
44. CW 2011: Continuation Workshop

Funding Agencies

Served on proposal review panels for the National Science Foundation (NSF) in 2016, 2017, 2020. Invited expert for the DARPA ISAT study group in 2020.

Publications

Google Scholar (as of Oct. 1, 2020):

- Citation count: 3422
- H-index: 32

csrankings.org:

- Ranked 3rd among PL researchers in the US for the past 6 years:
- 15 publications 2014-2019 in “top” PL conferences (POPL, PLDI, OOPSLA, ICFP), behind Alex Aiken (Stanford, 16) and Isil Dillig (UT Austin, 20)
- Does not include 5 “top” non-PL papers in 2014-2019 (MICRO, VLDB, SIGMOD, OSDI, NeurIPS)

Note: P denotes author who is postdoctoral researcher (or on other pre-faculty position) *at the time of writing*. Similarly, U and G indicate authors who were undergraduate and graduate students respectively *at the time of writing*. Acceptance rates are shown where available.

Edited Volumes

- [1] Eric Van Wyk, Tiark Rompf. Proceedings of the 17th ACM SIGPLAN International Conference on Generative Programming: Concepts and Experiences, GPCE 2018, Boston, MA, USA, November 5-6. ACM 2018, ISBN 978-1-4503-6045-6
- [2] Tiark Rompf, Alexander Alexandrov. Proceedings of The 16th International Symposium on Database Programming Languages, DBPL 2017, Munich, Germany, September 1, 2017. ACM 2017, ISBN 978-1-4503-5354-0
- [3] Martin Erwig, Tiark Rompf. Proceedings of the 2016 ACM SIGPLAN Workshop on Partial Evaluation and Program Manipulation, PEPM 2016, St. Petersburg, FL, USA, January 20 - 22, 2016. ACM 2016, ISBN 978-1-4503-4097-7
- [4] Tiark Rompf, Geoffrey Mainland. Proceedings of the 4th ACM SIGPLAN Workshop on Functional High-Performance Computing, FHPC@ICFP 2015, Vancouver, BC, Canada, September 3, 2015. ACM 2015, ISBN 978-1-4503-3807-3
- [5] José Pedro Magalhães, Tiark Rompf. Proceedings of the 10th ACM SIGPLAN workshop on Generic programming, WGP 2014, Gothenburg, Sweden, August 31, 2014. ACM 2014, ISBN 978-1-4503-3042-8

Journals

- [1] Tiark Rompf, Nada Amin G . A SQL to C Compiler in 500 Lines of Code. Journal of Functional Programming (32 pages, Volume 29, 2019)
- [2] Martin Odersky, Tiark Rompf P . Unifying Functional and Object-Oriented Programming with Scala. Commun. ACM 57(4): 76-86 (2014)
- [3] Arvind K. Sujeeth G , Kevin J. Brown G , HyoukJoong Lee G , Tiark Rompf P , Hassan Chafi, Martin Odersky, Kunle Olukotun. Delite: A Compiler Architecture for Performance-Oriented Embedded Domain-Specific Languages. ACM Trans. on Embedded Computing Systems (TECS) 13(4s), 134:1-134:25 (2014).

- [4] Tiark Rompf^G, Nada Amin^G, Adriaan Moors^P, Philipp Haller^P, Martin Odersky. Scala-Virtualized: Linguistic Reuse for Deep Embeddings. *Higher-Order and Symbolic Computation (HOSC)* 25(1), 165-207 (2012)
- [5] Tiark Rompf^G, Martin Odersky. Lightweight Modular Staging: A Pragmatic Approach to Runtime Code Generation and Compiled DSLs. *Commun. ACM* 55(6): 121-130 (2012)
- [6] HyoukJoong Lee^G, Kevin J. Brown^G, Arvind K. Sujeeth^G, Hassan Chafi^G, Tiark Rompf^G, Martin Odersky, Kunle Olukotun. Implementing Domain-Specific Languages for Heterogeneous Parallel Computing. *IEEE Micro: Special Issue on CPU, GPU, and Hybrid Computing* 31(5), 42-53 (2011)
- [7] Markus Hinkelmann^G, Andreas Jakoby, Nina Moebius^U, Tiark Rompf^U, Peer Stechert^U. A Cryptographically t-Private Auction System. *Concurrency and Computation: Practice and Experience* 23(12): 1399-1413 (2011)

Conferences and Workshops

- [1] Guannan Wei^G, Oliver Bracevac^P, Tiark Rompf. Compiling Symbolic Execution with Staging and Algebraic Effects. *OOPSLA 2020* (conditionally accepted)
- [2] Ruby Tahboub^G, Tiark Rompf. Architecting a Query Compiler for Spatial Workloads. *SIGMOD 2020*
- [3] Fei Wang^G, Guoyang Chen, Weifeng Zhang, Tiark Rompf. Parallel Training via Computation Graph Transformation. *IEEE BigData 2019*
- [4] Nada Amin, William E. Byrd, Tiark Rompf. Lightweight Functional Logic Meta-Programming. *APLAS 2019*, 20 pages, (22/50 = 44%)
- [5] Alen Stojanov^G, Tiark Rompf, Markus Püschel. A Stage-Polymorphic IR for Compiling MATLAB-Style Dynamic Tensor Expressions. *GPCE 2019*, 14 pages, (13/25 = 52%)
- [6] Guannan Wei^G, Yuxuan Chen^U, Tiark Rompf. Staged Abstract Interpreters. *OOPSLA 2019*, 32 pages, (72/201 = 35%)
- [7] Gregory Essertel^G, Guannan Wei^G, Tiark Rompf. Precise Reasoning with Structured Time, Structured Heaps, and Collective Operations. *OOPSLA 2019*, 30 pages, (72/201 = 35%)
- [8] Fei Wang^G, Daniel Zheng^U, James Decker^G, Xilun Wu^G, Gregory Essertel^G, Tiark Rompf. Demystifying Differentiable Programming: Shift/Reset the Penultimate Backpropagator. *ICFP 2019*, 31 pages, (39/119 = 32%)
- [9] Youyou Cong^G, Leo Osvald^G, Grégory Essertel^G, Tiark Rompf. Compiling with Continuations, or without? Whatever. *ICFP 2019*, (39/119 = 32%)
- [10] Zhanfu Yang^G, Fei Wang^G, Ziliang Chen^G, Guannan Wei^G, Tiark Rompf. Graph Neural Reasoning for 2-Quantified Boolean Formula Solvers. *ICML 2019 Workshop on Learning and Reasoning with Graph-Structured Data*, 7 pages
- [11] Grégory Essertel^G, Ruby Tahboub^G, Fei Wang^G, James Decker^G, Tiark Rompf. Flare & Lantern: Efficiently Swapping Horses Midstream. *VLDB 2019 Demo Track*, 4 pages
- [12] Ruby Tahboub^G, Xilun Wu^G, Grégory Essertel^G, Tiark Rompf. Compiling Graph Queries in Relational Engines. *DBPL 2019*, 12 pages
- [13] Dan Moldovan, James Decker^G, Fei Wang^G, Andrew A Johnson, Brian K Lee, Zachary Nado, D Sculley, Tiark Rompf, Alexander B Wiltschko. AutoGraph: Imperative-Style Coding with Graph-Based Performance. *SysML 2019*, 17 pages, (32/189 = 16%)
- [14] Fei Wang^G, James Decker^G, Xilun Wu^G, Grégory Essertel^G, Tiark Rompf. Backpropagation with Continuation Callbacks: Towards Efficient and Expressive Differentiable Programming. *NeurIPS 2018*, 12 pages, (1019/4854 = 20%)

- [15] Grégory Essertel^G, Ruby Tahboub^G, James Decker^G, Kevin J. Brown^G, Kunle Olukotun, Tiark Rompf. Flare: Optimizing Apache Spark for Scale-Up Architectures and Medium-Size Data. OSDI 2018, 17 pages, (47/257 = 18%)
- [16] Guannan Wei^G, James Decker^G, Tiark Rompf. Refunctionalization of Abstract Abstract Machines (Functional Pearl). ICFP 2018, 28 pages, (40/118 = 33%)
- [17] Ruby Tahboub^G, Grégory Essertel^G, Tiark Rompf. How to Architect a Query Compiler, Revisited. SIGMOD 2018, 16 pages, (90/461 = 19%)
- [18] Fei Wang^G, Tiark Rompf. A Language and Compiler View on Differentiable Programming. ICLR 2018 Workshop Track, 5 pages, (196/346 = 56%)
- [19] Alen Stojanov^G, Ivaylo Toskov^U, Tiark Rompf, Markus Püschel. SIMD Intrinsics on Managed Language Runtimes. CGO 2018, 14 pages, (30/105 = 28%)
- [20] Nada Amin^P, Tiark Rompf. Collapsing Towers of Interpreters. POPL 2018, 33 pages, (66/272 = 24%)
- [21] Georg Ofenbeck^G, Tiark Rompf, Markus Püschel. Staging for Generic Programming in Space and Time. GPCE 2017, 14 pages, (21/56 = 37%)
- [22] Leo Oswald^G, Tiark Rompf. Rust-Like Borrowing with 2nd-Class Values (Short Paper). Scala Symposium 2017, 5 pages, (12/21 = 57%)
- [23] Fei Wang^G, Tiark Rompf. Towards Strong Normalization for Dependent Object Types (DOT). ECOOP 2017, 25 pages, (27/81 = 33%)
- [24] Leo Oswald^G, Tiark Rompf. Flexible Data Views: Design and Implementation. PLDI Workshop on Libraries, Languages, and Compilers for Array Programming (ARRAY 2017), 8 pages, (8/14 = 57%)
- [25] Yiyang Chang^G, Gustavo Petri^P, Sanjay Rao, Tiark Rompf. Composing middlebox and traffic engineering policies in SDNs. IEEE INFOCOM Workshop on Software-Driven Flexible and Agile Networking (SWFAN 2017), 6 pages
- [26] Nada Amin^G, Tiark Rompf. Type Soundness Proofs with Definitional Interpreters. POPL 2017, 14 pages, (64/282 = 22%)
- [27] Nada Amin^G, Tiark Rompf. LMS-Verify: Abstraction without Regret for Verified Systems Programming. POPL 2017, 15 pages, (64/282 = 22%)
- [28] Tiark Rompf, Kevin J. Brown^G. Functional Parallels of Sequential Imperatives (Short Paper). PEPM 2017, 6 pages, (9/21 = 42%)
- [29] Tiark Rompf. Reflections on LMS: Exploring Front-End Alternatives. Scala Symposium 2016, 10 pages, (14/23 = 60%)
- [30] Georg Ofenbeck^G, Tiark Rompf, Markus Püschel. RandIR: Differential Testing for Embedded Compilers. Scala Symposium 2016, 10 pages, (14/23 = 60%)
- [31] Ruby Tahboub^G, Gregory Essertel^G, Tiark Rompf. On Supporting Compilation in Main-Memory Spatial Query Engines (Vision Paper). SIGSPATIAL 2016, 4 pages, (5/13 = 38%)
- [32] Tiark Rompf, Nada Amin^G. Type Soundness for Dependent Object Types. OOPSLA 2016, 18 pages, (52/203 = 25%)
- [33] Leo Oswald^G, Gregory Essertel^G, Xilun Wu^G, Lilliam I Gonzales Alayon^G, Tiark Rompf. Gentrification Gone too Far? Affordable 2nd-Class Values for Fun and (Co-)Effect. OOPSLA 2016 (**distinguished artifact award**), 18 pages, (52/203 = 25%)
- [34] Kevin J. Brown^G, HyoukJoong Lee^G, Christopher de Sa^G, Tiark Rompf, Kunle Olukotun. Have Abstraction and Eat Performance Too: Optimized Heterogeneous Computing with Parallel Patterns. CGO 2016, 12 pages, (25/108 = 23%)

- [35] Tiark Rompf. The Essence of Multi-Stage Evaluation in LMS. In: A List of Successes That Can Change the World—Essays Dedicated to Philip Wadler on the Occasion of His 60th Birthday (Wadlerfest 2016, **invited submission**), 18 pages
- [36] Nada Amin^G, Samuel Grütter^U, Martin Odersky, Tiark Rompf, Sandro Stucki^G. The Essence of Dependent Object Types. In: A List of Successes That Can Change the World—Essays Dedicated to Philip Wadler on the Occasion of His 60th Birthday (Wadlerfest 2016, **invited submission**) 24 pages
- [37] Tiark Rompf, Nada Amin^G. Functional Pearl: a SQL to C compiler in 500 lines of code. ICFP 2015 (**invited for JFP special issue**), 8 pages, (35/119 = 29%)
- [38] Nicolas Stucki^U, Tiark Rompf, Vlad Ureche^G, Phil Bagwell. RRB vector: a practical general purpose immutable sequence. ICFP 2015, 13 pages, (35/119 = 29%)
- [39] Tiark Rompf, Kevin J. Brown^G, HyoukJoong Lee^G, Arvind K. Sujeeth^G, Manohar Jonnalagedda^G, Nada Amin^G, Georg Ofenbeck^G, Alen Stojanov^G, Yannis Klonatos^G, Mohammad Dashti^G, Christoph Koch, Markus Püschel, Kunle Olukotun. Go Meta! A Case for Generative Programming and DSLs in Performance Critical Systems. Summit on Advances in Programming Languages (SNAPL 2015), 24 pages
- [40] HyoukJoong Lee^G, Kevin J. Brown^G, Arvind K. Sujeeth^G, Tiark Rompf^P, Kunle Olukotun. Locality-Aware Mapping of Nested Parallel Patterns on GPUs. MICRO 2014, 12 pages, (53/279 = 18%)
- [41] Nada Amin^G, Tiark Rompf^P, Martin Odersky. Foundations of Path-Dependent Types. OOPSLA 2014, 17 pages, (52/186 = 27%)
- [42] Manohar Jonnalagedda^G, Thierry Coppey^U, Sandro Stucki^G, Tiark Rompf^P, Martin Odersky. Staged Parser Combinators for Efficient Data Processing. OOPSLA 2014, 17 pages, (52/186 = 27%)
- [43] Nithin George^G, Hyoukjoong Lee^G, David Novo^P, Tiark Rompf^P, Kevin J. Brown^G, Arvind K. Sujeeth^G, Martin Odersky, Kunle Olukotun, Paolo Ienne. Hardware System Synthesis from Domain-Specific Languages. International Conference on Field Programmable Logic and Applications (FPL 2014), 8 pages, (62/259 = 23%)
- [44] Nada Amin^G, K. Rustan M. Leino, Tiark Rompf^P. Computing with an SMT Solver. International Conference on Tests & Proofs (TAP 2014), 16 pages, (14/33 = 42%)
- [45] Alen Stojanov^G, Georg Ofenbeck^G, Tiark Rompf^P, Markus Püschel. Abstracting Vector Architectures in Library Generators: Case Study Convolution Filters. PLDI Workshop on Libraries, Languages, and Compilers for Array Programming (ARRAY 2014), 6 pages, (17/25 = 68%)
- [46] Yannis Klonatos^G, Christoph Koch, Tiark Rompf^P, Hassan Chafi. Building Efficient Query Engines in a High-Level Language. VLDB 2014 (**best paper award**), 12 pages, (127/559 = 22%)
- [47] Tiark Rompf^P, Arvind K. Sujeeth^G, Kevin J. Brown^G, HyoukJoong Lee^G, Hassan Chafi, Kunle Olukotun. Surgical Precision JIT Compilers. PLDI 2014, 12 pages, (52/287 = 18%)
- [48] Nithin George^G, David Novo^P, Tiark Rompf^P, Martin Odersky, Paolo Ienne. Making Domain-Specific Hardware Synthesis Tools Cost-Efficient. International Conference on Field-Programmable Technology (FPT 2013), 8 pages, (34/163 = 21%)
- [49] Georg Ofenbeck^G, Tiark Rompf^P, Alen Stojanov^G, Martin Odersky, Markus Püschel. Spiral in Scala: Towards the Systematic Construction of Generators for Performance Libraries. GPCE 2013, 10 pages, (20/59 = 33%)
- [50] Arvind K. Sujeeth^G, Austin Gibbons^G, Kevin J. Brown^G, HyoukJoong Lee^G, Tiark Rompf^P, Martin Odersky, Kunle Olukotun. Forge: Generating a High Performance DSL Implementation from a Declarative Specification. GPCE 2013, 10 pages, (20/59 = 33%)
- [51] Sandro Stucki^G, Nada Amin^G, Manohar Jonnalagedda^G, Tiark Rompf^P What are the Odds? Probabilistic Programming in Scala. Scala Workshop 2013, 9 pages, (12/22 = 55%)

- [52] Arvind Sujeeth^G, Tiark Rompf^P, Kevin Brown^G, HyoukJoong Lee^G, Hassan Chafi^G, Victoria Popic^G, Michael Wu^G, Aleksandar Prokopec^G, Vojin Jovanovic^G, Martin Odersky, Kunle Olukotun. Composition and Reuse with Compiled Domain-Specific Languages. ECOOP 2013, 27 pages, (29/116 = 25%)
- [53] Tiark Rompf^P, Arvind K. Sujeeth^G, Nada Amin^G, Kevin J. Brown^G, Vojin Jovanovic^G, HyoukJoong Lee^G, Martin Odersky, Kunle Olukotun. Optimizing Data Structures in High-Level Programs: New Directions for Extensible Compilers based on Staging. POPL 2013, 14 pages, (43/238 = 18%)
- [54] Stefan Ackermann^G, Vojin Jovanovic^G, Tiark Rompf^P, Martin Odersky. Jet: An Embedded DSL for High Performance Big Data Processing. VLDB Workshop on End-to-end Management of Big Data (BigData 2012), 10 pages
- [55] Grzegorz Kossakowski^G, Nada Amin^G, Tiark Rompf^G, Martin Odersky. JavaScript as an Embedded DSL. ECOOP 2012, 26 pages, (30/140 = 21%)
- [56] Adriaan Moors^P, Tiark Rompf^G, Philipp Haller^P, Martin Odersky. Scala-Virtualized. PEPM 2012 (**selected for HOSC special issue**), 4 pages, (19/37 = 51%)
- [57] Vlad Ureche^G, Tiark Rompf^G, Arvind Sujeeth^G, Hassan Chafi^G, Martin Odersky. StagedSAC: A Case Study in Performance-Oriented DSL Development. PEPM 2012, 10 pages, (19/37 = 51%)
- [58] Kevin J. Brown^G, Arvind K. Sujeeth^G, HyoukJoong Lee^G, Tiark Rompf^G, Hassan Chafi^G, Martin Odersky, Kunle Olukotun. A Heterogeneous Parallel Framework for Domain-Specific Languages. PACT 2011, 12 pages, (35/221 = 15%)
- [59] Tiark Rompf^G, Arvind K. Sujeeth^G, HyoukJoong Lee^G, Kevin J. Brown^G, Hassan Chafi^G, Martin Odersky, Kunle Olukotun. Building-Blocks for Performance Oriented DSLs. DSL 2011, 25 pages, (7/14 = 50%)
- [60] Aleksandar Prokopec^G, Phil Bagwell, Tiark Rompf^G, Martin Odersky. A Generic Parallel Collection Framework. Euro-Par 2011, 12 pages, (81/271 = 29%)
- [61] Arvind K. Sujeeth^G, HyoukJoong Lee^G, Kevin J. Brown^G, Tiark Rompf^G, Hassan Chafi^G, Michael Wu^G, Anand R. Atreya^G, Martin Odersky, Kunle Olukotun. OptiML: An Implicitly Parallel Domain-Specific Language for Machine Learning. ICML 2011, 8 pages, (152/589 = 25%)
- [62] Hassan Chafi^G, Zach DeVito^G, Adriaan Moors^P, Tiark Rompf^G, Arvind K. Sujeeth^G, Pat Hanrahan, Martin Odersky, Kunle Olukotun. Language Virtualization for Heterogeneous Parallel Computing. Onward! 2010, 13 pages, (9/39 = 23%)
- [63] Tiark Rompf^G, Martin Odersky. Lightweight Modular Staging: A Pragmatic Approach to Runtime Code Generation and Compiled DSLs. GPCE 2010 (**selected as CACM research highlight**), 10 pages, (18/59 = 30%)
- [64] Tiark Rompf^G, Ingo Maier^G, Martin Odersky. Implementing First-Class Polymorphic Delimited Continuations by a Type-Directed Selective CPS-Transform. ICFP 2009, 12 pages, (26/85 = 30%)
- [65] Markus Hinkelmann^G, Andreas Jakoby, Nina Moebius^U, Tiark Rompf^U, Peer Stechert^U. A Cryptographically t-Private Auction System. NSS 2009, 8 pages, (53/151 = 35%)

Theses

- [1] Tiark Rompf^G. Lightweight Modular Staging and Embedded Compilers: Abstraction Without Regret for High-Level High-Performance Programming. PhD Thesis, LAMP, EPFL, 2012
- [2] Tiark Rompf^U. Design and Implementation of a Programming Language for Concurrent Interactive Systems. Master Thesis, Institute of Software Technology and Programming Languages, University of Lübeck, 2007

Presentations

Invited Talks

1. A Programming Language and Compiler View on Data Management and ML Systems. Data Engineering Lecture Series, Hasso Plattner Institute, Potsdam, 2020
2. A Programming Language and Compiler View on Data Management and ML Systems. UIUC, Urbana-Champaign, 2019
3. Flare & Lantern: accelerators for Big Data and Deep Learning. NII Shonan Meeting 143 on Programming Language Support for Data-intensive Applications, 2019
4. A Programming Language and Compiler View on AI Systems. C4ML Workshop, Washington DC, 2019
5. Native Compilation as Query Accelerator. SAP HANA Tech Days, Galveston, 2018
6. Demystifying Differentiable Programming. IFIP Working Group 2.11, 18th Meeting, Kyoto University, 2018
7. What Programming Languages Research has to Offer for AI and Vice-Versa. Google DeepMind, London, 2018
8. What Programming Languages Research has to Offer for Big Data and AI. Huawei Midwest Research Summit: Artificial Intelligence, Champaign, 2018
9. Building efficient query engines in a high-level language. SAP HANA Tech Days, Munich, 2017
10. How types can turn a SQL interpreter into a SQL compiler: Building efficient query engines in a high-level language. NII Shonan Meeting 98 on Language integrated queries: towards standard logics for big data analytics, 2017
11. LMS: Generate all the things! GPCE'16 Keynote, Amsterdam, 2016
12. Adding SQL to the mix: faster queries in multi-paradigm DSL applications. NSF sponsored Workshop on Architecture and Software for Emerging Applications (WASEA), Haifa, Israel, 2016
13. LMS-Verify: Abstraction with Regret for Verified Systems Programming. IFIP Working Group 2.11, 16th Meeting, IU Bloomington, 2016
14. DSLs in Scala. Summer School on Domain-Specific Languages, EPFL, Switzerland, 2015
15. A SQL to C compiler in 500 lines of Scala. IFIP Working Group 2.11, 14th Meeting, University of Stellenbosch, South Africa, 2015
16. Abstraction Without Regret for High-Level High-Performance Programming. Professor's Day, Huawei Research, Moscow, 2014
17. A SQL to C compiler in 500 lines of code. Dagstuhl Seminar 14511 Programming Languages for Big Data (PlanBig), 2014
18. Dagstuhl Seminar 14271 on Scripting Languages and Frameworks: Analysis and Verification, 2014
19. Staging with Scala, LMS and Delite (LMS 5 years later). NII Shonan Meeting 2014-7 on Staging and High-Performance Computing: Theory and Practice, 2014
20. Abstraction Without Regret for High-Level High-Performance Programming. Purdue University, 2014
21. Staging for Database Systems: From Query Interpreters to Query Compilers. IFIP Working Group 2.11, 13th Meeting, CMU, 2014
22. Lightweight Modular Staging in Scala. The Future of Programming Symposium, TU Delft, 2014

23. Scala, LMS and Delite: Big Data on Heterogeneous Architectures. NSF DALI Workshop on Dynamic Languages for Scalable Data Analytics, Indianapolis, 2013
24. Surgical Precision JIT Compilers. Splash-I, Indianapolis, 2013
25. DSLs for Parallelism: Problems, Solutions and Open Issues. LaME Workshop on Languages for the Multicore Era, Montpellier, 2013
26. Project Lancet: Surgical Precision JIT Compilers. IFIP Working Group 2.11, 12th Meeting, University of Minnesota, 2013
27. Big Data in Industry (Panel Discussion). XLDB Europe Workshop, CERN, 2013
28. Abstraction Without Regret: Can we Bridge the Productivity vs Performance Gap in Software Development? ETH Zürich, 2013
29. High-Performance Embedded DSLs in Scala. RAWFP Workshop on Productivity and Performance through Resource Aware Functional Programming, Chalmers University, 2012
30. High-Performance Embedded DSLs with Delite. NII Shonan Meeting 2012-4 on Bridging the Theory of Staged Programming Languages and the Practice of High-Performance Computing, 2012

Conference Tutorials

1. Tiark Rompf, Kunle Olukotun, Markus Püschel (organizers). Scala, LMS and Delite for High-Performance DSLs and Program Generators. PLDI 2017
2. Tiark Rompf, Kunle Olukotun, Markus Püschel (organizers). Scala, LMS and Delite for High-Performance DSLs and Program Generators. CGO 2015
3. Tiark Rompf, Sandro Stucki, Manohar Jonnalagedda. Batteries Included: Generative Programming with Scala and LMS. SPLASH 2014 Fall School
4. Tiark Rompf, Nada Amin. Batteries Included: Generative Programming with Scala and LMS. CUFP 2014
5. Tiark Rompf, Kunle Olukotun, Markus Püschel (organizers). Scala, LMS and Delite for High-Performance DSLs and Program Generators. PLDI 2013
6. Tiark Rompf, Arvind K. Sujeeth. Scala, LMS and Delite for High-Performance DSLs and Program Generators. ECOOP 2013 Summer School

Selected Other Talks

1. Flare & Lantern: Accelerators for Spark and Deep Learning. ScalaDays, Lausanne, 2019
2. Flare and TensorFlare: Native Compilation for Spark and TensorFlow Pipelines. Spark+AI Summit, San Francisco, 2018
3. How types can turn a SQL interpreter into a SQL compiler. Curry On!, Barcelona, 2017
4. Flare: Scale up Spark SQL with native compilation and set your data on fire! Spark Summit West, San Francisco, 2017
5. Flare: Scale up Spark SQL with native compilation and set your data on fire! Scala Days, Chicago, 2017
6. Flare: Scale up Spark SQL with native compilation and set your data on fire! Spark Summit EU, Brussels, 2016
7. How types can turn a SQL interpreter into a SQL compiler: Building efficient query engines in a high-level language. Strangeloop, St. Louis, 2014
8. Implicits in Scala. ML Family Workshop 2014, Gothenburg 2014. With Nada Amin.

9. Building a DBMS in Scala or how types can turn a SQL interpreter into a SQL compiler. ScalaDays, Berlin, 2014
10. Project Lancet: Surgical Precision JIT Compilers. ScalaDays, New York City, 2013
11. Delite: Domain Specific Languages for Big Data and Heterogeneous Parallelism. XLDB Europe Workshop, CERN, 2013
12. High-Level High-Performance Programming with Scala-Virtualized, LMS and Delite. ScalaDays, London, 2012
13. Fast Concatenation for Immutable Vectors. ScalaDays, London, 2012
14. Node.js? Done.scala! Implementing Scalable Async IO using Delimited Continuations. ScalaDays, Stanford University, 2011

Funding

PurPL Center: \$100,000 in industry gifts received in 2019 (Facebook, Microsoft, SambaNova Systems, Reservoir Labs). \$250,000 pledged so far for 2020

Grants Awarded

1. HACCLE: High-Assurance Compositional Cryptography: Languages and Environments. PI: Milind Kulkarni, co-PIs: Tiark Rompf, Jeremiah Blocki, Aniket Kate, Benjamin Delaware, Roopsha Samanta. \$10,732,899 (program was discontinued after year one, affecting all funded teams). IARPA-BAA-1705 HECTOR 2019
2. FMitF: Track I: Symbolic Reasoning with Graph Networks. PIs: Tiark Rompf, Jennifer Neville, Bruno Ribeiro. \$750,000. NSF 2019
3. III: Small: In-memory Distributed Graph Data Management. PIs: Walid Aref, Tiark Rompf. \$499,000. NSF 2019
4. NSF Student Travel Grant for 2019 Midwest PL Summit (MWPLS) PIs: Milind Kulkarni, Tiark Rompf \$5,000. NSF 2019
5. Facebook Probability and Programming Research Award. PI: Tiark Rompf. \$67,390. Facebook 2019
6. VMware Systems Research Award. PI: Tiark Rompf. \$100,000. VMware 2018
7. HIRP Open Research Award. PI: Tiark Rompf. \$49,996. Huawei 2018
8. Formal Methods for Robust Machine Learning. PI: Jennifer Neville, co-PIs: Dan Goldwasser, Bruno Ribeiro, Tiark Rompf, Roopsha Samanta. \$265,000. Purdue Integrated Data Science Initiative 2018
9. Google Faculty Research Award. PI: Tiark Rompf. \$80,000. Google 2018
10. Purdue Major Scientific Equipment Award. PIs: Walid Aref, Jennifer Neville (I initiated the proposal but assistant profs were not eligible to serve as PIs.) \$268,000. Purdue Office of the Provost 2017
11. DOE Early Career Award: Program Generators for Exascale and Beyond. PI: Tiark Rompf. \$749,055. DOE 2017
12. Google Faculty Research Award. PI: Tiark Rompf. \$84,787. Google 2017
13. CAREER: Generative Programming and DSLs for Safe Performance Critical Systems. PI: Tiark Rompf. \$517,193. NSF 2016
14. SHF: Medium: Collaborative Research: From Volume to Velocity: Big Data Analytics in Near-Realtime. PIs: Tiark Rompf, Kunle Olukotun, Christopher Ré (Purdue/Stanford). \$1,000,000 (Purdue: \$333,840). NSF 2016
15. Abstraction Without Regret for Program Verification. PI: Tiark Rompf. \$16,714. Purdue Research Foundation 2015

Teaching

Classes Taught

- CS590 Deep Learning for Symbolic Reasoning fall 2018
- CS502 Compilers fall 2019, spring 2019, fall 2016
- CS352 Compilers spring 2020, spring 2018, fall 2017, fall 2015
- CS565 Programming Languages spring 2016, spring 2015
- CS590 Metaprogramming and Program Generation fall 2014

Department Service

- Graduate Admissions 2014, 2015, 2019
- Graduate Committee 2016, 2017, 2018
- GoBoiler Internships 2018, 2019
- Faculty Hiring (Systems) 2019

Current and Past Post-Doctoral Scholars

- Oliver Bračevac (since 2020)
- Yuyan Bao (since 2019)

Current and Past PhD Advisees

- Anxhelo Xhebraj (since 2020)
- Hannah Leung (since 2020)
- Kasra Sadeghi (since 2020)
- Supun Abeysinghe (since 2020)
- Xilun Wu (expected prelim 2021)
- Guannan Wei (expected prelim 2021)
- James Decker
- Fei Wang
- Grégory Essertel (graduated Fall 2019, Strateos)
Improving Performance of Data-Centric Systems through Fine-Grained Code Generation
- Ruby Tahboub (graduated Spring 2019, Visiting Assistant Prof at Purdue)
Architecting Query Compilers for Diverse Workloads
- Leo Osvald (graduated Spring 2018, Facebook)
Lightweight Programming Abstractions for Increased Safety and Performance

Current and Past MS Advisees

- Nitin John Raj (since 2020)
- Nandita Viswanath (graduated 2019, Bloomberg)
- Prahlad Joshi (graduated 2015, Facebook)
- Roman Tsegelskyi (graduated 2015, Microsoft)

PhD Defense Committees

- Wen-Chuan Lee (Advisor: Xiangyu Zhang) 2019-12-12
- Laith Sakka (Advisor: Milind Kulkarni) 2019-11-26
- Shin-Yeh Tsai (Advisor: Yiyang Zhang) 2019-07-16
- Gowtham Kaki (Advisor: Suresh Jagannathan) 2019-07-12
- Alen Stojanov (ETH, advisor: Markus Püschel) 2019-05-27
- Amgad M. Madkour (Advisor: Walid Aref) 2018-11-29
- Mohamed S. Hassan (Advisor: Walid Aref) 2018-07-23
- Lianjie Cao (Advisor: Sonia Fahmy) 2018-04-27
- Georg Ofenbeck (ETH, advisor: Markus Püschel) 2017-04-04
- Keith G. Chapman (Advisor: Tony Hosking) 2016-12-02
- He Zhu (Advisor: Suresh Jagannathan) 2016-07-11
- Kevin J. Brown (Stanford, advisor: Kunle Olukotun) 2016-05-31

Supervised Undergraduate (and MS) Research

- Shangyin Tan 2020
- Xinping Zhang 2020
- Abhishek Sharma 2020
- Akhil Sai Bandrupalli (MS) 2019
- Zhanfu Yang (MS) 2019
- Zequan Wu 2019
- Yuxuan Chen 2018
- Vritant Bhardwaj 2018
- Daniel F. Zheng 2018
- William G. Pierce 2018
- Adam F. Johnston 2018
- Lilliam I. Gonzales Alayon (MS) 2015
- Rohan Barman 2015
- Di Liu 2015
- Cameron Kachur 2015

External Research Interns

- Alexey Kashuba, EPFL 2020
- Philipp Danzinger, TU Vienna 2020
- Thaïs Baudon, ENS Rennes 2019
- Yuanjing Shi, UIUC (MS) 2019
- Lala Shahbandayeva, ADA University Azerbaijan 2019
- Sara Seidl, Univ. of Salzburg (MS) 2019
- Christian Mösl, Univ. of Salzburg 2019
- Anxhelo Xhebraj, Sapienza Univ. of Rome (MS) 2019
- Youyou Cong, Ochanomizu University (PhD) 2018
- Sreekar Garlapati, IIT-Bombay 2018