GEORGE STELLE

stelleg@lanl.gov • +1 (505) 750 8506

EMPLOYMENT

2016-Present

Research Scientist
Los Alamos National Laboratory
Research and develop compilers and programming languages for heterogeneous high performance computing and scaling of ML inference and training

2013-2016

Graduate Student Researcher
Sandia National Laboratories
Research and develop lightweight threading libraries for high performance computing on NUMA architectures

2009-2013

Graduate Research Assistant
University of New Mexico
Research and develop call-by-need compilers, network models, control flow analysis, and type theory

EDUCATION

2019

PhD : Computer Science
University of New Mexico • Department of Computer Science
Thesis: Shared-Environment Call-by-Need • Advisor: Darko Stefanovic

2013

MS : Computer Science
University of New Mexico • Department of Computer Science
Advisor: Stephanie Forrest

2008

BS : Computational Intelligence and Design
University of British Columbia • Department of Cognitive Systems

PUBLICATIONS

2025

Kaetlyn Gibson, Po-E Li, Valerie Li, Martha Dix, Li-Wei Hung, George Widgery Stelle, Michal Babinski, Patrick Chain, and Bin Hu. Prime: Protein representation inference for mutation evaluation. *Nature Biotechnology*, 2025 (in submission)

2024

Romain Pereira, George Stelle, and Patrick Carribault. Taskgrind: Heavyweight dynamic binary instrumentation for parallel programs analysis. In *SC24-W: Workshops of the International Conference for High Performance Computing, Networking, Storage and Analysis*, pages 214–221. IEEE, 2024

2018

George Stelle and Darko Stefanovic. Verifiably lazy: Verified compilation of call-by-need. In *Proceedings of the 30th Symposium on Implementation and Application of Functional Languages*, IFL 2018, page 49–58, New York, NY, USA, 2018. Association for Computing Machinery

W. Lee, G. Stelle, P. McCormick, and A. Aiken. Correctness of dynamic dependence analysis for implicitly parallel tasking systems. In *2018 IEEE/ACM 2nd International Workshop on Software Correctness for HPC Applications (Correctness)*, pages 17–24, Nov 2018

2017

George Stelle, William S. Moses, Stephen L. Olivier, and Patrick McCormick. OpenMPIR: implementing openmp tasks with tapir. In *Proceedings of the Fourth Workshop on the LLVM Compiler Infrastructure in HPC*, LLVM-HPC'17, pages 3:1–3:12, New York, NY, USA, 2017. ACM

Noah Evans, Stephen L. Olivier, Richard Barrett, and George Stelle. Scheduling chapel tasks with qthreads on manycore: A tale of two schedulers. In *Proceedings of the 7th International Workshop on Runtime and Operating Systems for Supercomputers ROSS 2017*, ROSS '17, pages 4:1–4:8, New York, NY, USA, 2017. ACM

2016 George Stelle, Darko Stefanovic, Stephen L. Olivier, and Stephanie Forrest. Cactus environment machine. In David Van Horn and John Hughes, editors, *Trends in Functional Programming*, pages 24–43, Cham, 2016. Springer International Publishing

2014 G. Stelle, S. L. Olivier, D. Stark, A. F. Rodrigues, and K. S. Hemmert. Using a complementary emulation-simulation co-design approach to assess application readiness for processing-in-memory systems. In *2014 Hardware-Software Co-Design for High Performance Computing*, pages 64–71, Nov 2014

2013 Steven Hofmeyr, Tyler Moore, Stephanie Forrest, Benjamin Edwards, and George Stelle. Modeling internet-scale policies for cleaning up malware. In Bruce Schneier, editor, *Economics of Information Security and Privacy III*, pages 149–170, New York, NY, 2013. Springer New York

2012 Benjamin Edwards, Tyler Moore, George Stelle, Steven Hofmeyr, and Stephanie Forrest. Beyond the blacklist: Modeling malware spread and the effect of interventions. In *Proceedings of the 2012 New Security Paradigms Workshop*, NSPW '12, pages 53–66, New York, NY, USA, 2012. ACM

TALKS

2024 Dominance is not a Tree
LLVM Performance Workshop at CGO

2019 Concurrency in LLVM
SIAM CSE 2019

LLVM and LANL ATDM
ECP 2019 Annual Meeting

2018 Preventing Data Races with Refinement Types
FHPC 2018

Verifiably Lazy: Verified Compilation of Call-by-Need
IFL 2018

2017 OpenMPIR: Impelementing OpenMP Tasks with Tapir
LLVM-HPC 2017

Programming Quantum Annealers
OBT 2017

2016 Cactus Environment Machine
TFP 2016

SOFTWARE

2017-Present Kitsune • *Lead Developer*
Contribution: Multiple frontends and backends to Tapir
`https://github.com/lanl/kitsune`

2017-Present Tapir • *Contributor*
Contribution: Multiple backends, bug fixes

https://github.com/wsmoses/tapir-llvm
https://github.com/wsmosts/tapir-clang

2012-2018 Cactus Environment Machine • *Creator*
Contribution: Shared-environment call-by-need implementations, including a native code compiler and a verified compiler
https://github.com/stelleg/cem • https://github.com/stelleg/cem_coq

2018 Equites • *Creator*
Contribution: High level, type-safe interface to legion
https://github.com/stelleg/equites

2017-2018 Lightweight 3D Printer Interface • *Creator*
Contribution: Low resource G-code communication pipe
https://github.com/stelleg/l3dp

2017 Coq • *Contributor*
Contribution: Misc. vector functions and lemmas to standard library
https://github.com/coq/coq

2017 HSAPI • *Creator*
Contribution: Haskell bindings to D-Wave SAPI interface
https://github.com/lanl/hsapi

2011 Braids in Classical Dynamics • *Creator*
Contribution: Implementation of gradient descent algorithm to find fixed point solutions to 2D n-body problems starting with a drawing
http://cs.unm.edu/~stelleg/braids