School of Computer Science
Institute for Software Research
Carnegie Mellon University
5000 Forbes Ave
Pittsburgh, PA 15213
USA

Phone: +1 (412) 268-2748
heather.miller@cs.cmu.edu
http://heather.miller.am

HEATHER MILLER

Citizenship

USA

Research Interests

Concurrent, distributed, eventually-consistent (edge computing), data-centric, and data-intensive (big data) programming, from the perspective of programming languages. I work on both theoretical ideas & implementations typically in/for the Scala programming language. **My goal is to reduce the burden of building distributed systems.**

Education

EPFL, *Lausanne, Switzerland* — *2009 – 2015*
Ph.D. in Computer Science
Advisor: Martin Odersky — *2011 – 2015*

University of Miami, *Coral Gables, FL* — *2006 – 2009*
BSEE in Electrical Engineering, Audio Engineering, *with honors, May 2009*

Cooper Union for the Advancement of Science and Art, *New York, NY* — *2004 – 2006*

Employment

Carnegie Mellon University, *Pittsburgh, PA, USA* — *8/2018 –*
Assistant Professor
School of Computer Science, Institute for Software Research

Northeastern University, *Boston, MA, USA* — *9/2016 – 7/2018*
Assistant Clinical Professor
College of Computer and Information Science

Scala Center, EPFL, *Lausanne, Switzerland* — *10/2015 – 7/2018*
Executive Director, Research Scientist
Founded a new not-for-profit center dedicated to research, open source development, and education surrounding the Scala programming language.

Databricks, *Berkeley, CA, USA* — *8/2014 – 11/2014*
Research Intern
Supervisor: Matei Zaharia
Integrated Scala Pickling, our framework for fast, boilerplate-free, extensible serialization focused on distributed programming (OOPSLA'13), into Spark. Developed new function-passing programming model and framework, can be thought of as a generalization of Spark/MapReduce programming model (JFP'18).

Teaching Experience (Classroom)

Co-Instructor, — *Spring 2019*
17-356: Software Engineering for Startups — *Carnegie Mellon*

Instructor, Designer, *Spring 2018*
CS4240: Large-Scale Parallel Data Processing *Northeastern*

Instructor, Designer, *Fall 2016*
CS7680: Programming Models for Distributed Computation *Northeastern*

Co-Instructor, Co-Designer, (*with Viktor Kunčak & Martin Odersky)* *Spring 2016*
CS 206: Parallelism & Concurrency *EPFL*

Co-Instructor, Co-Designer, (*with Viktor Kunčak & Martin Odersky)* *Spring 2015*
CS 212: Reactive Programming & Parallelism *EPFL*

(Lead) Teaching Assistant, *Fall 2011-2014*
CS 201: Functional Programming *EPFL*

Teaching Experience (MOOCs)

Instructor, Designer, *Big Data Analysis with Scala and Spark* *2017 –*
Popular Coursera MOOC on big data analysis using Spark. *Coursera*

- Designed lectures and produced lecture videos. Designed exercises and developed cloud-hosted automated graders.
- Between March-November 2017, over 120,000 registered learners.

Lead, *Scala Specialization (mini-degree)* *2015 –*
Responsible for EPFL's offering of a Scala *mini-degree* on Coursera. *Coursera*

- Assembled offering of 4 Scala MOOCs, topped off with a capstone project. Taught and produced 1 course in the specialization and managed the development of the remaining 3 courses and the project.

Lead, *Functional Programming Principles in Scala* *2012 – 2014*
Popular Coursera MOOC on functional programming in Scala. *Coursera*

- Lead teaching staff member, organized a team of graduate students, managed content production, designed course exercises with cloud-hosted grading, production of lecture videos, etc.
- >400,000 learners across iterations & largest completion rate for a course its size (~19%)

Book

Distributed Programming *MIT Press 2018/2019*
Heather Miller, Nat Dempkowski, James Larisch,
Christopher Meiklejohn, and Philipp Haller

A textbook about the building blocks we use to build distributed systems. These range from the small, RPC, futures, actors, to the large; systems built up of these components like MapReduce and Spark. We explore issues and concerns central to distributed systems like consistency, availability, and fault tolerance, from the lens of the programming models and frameworks that the programmer uses to build these systems.
Source (draft)

Publications: Journals

A Programming Model and Foundation for Lineage-Based Distributed *JFP 2018*
Heather Miller, Philipp Haller, Normen Müller
Journal of Functional Programming
Special Issue: Programming Languages for Big Data

Publications: Conferences

Simplicitly: Foundations and Applications of Implicit Function Types *POPL 2018*
Martin Odersky, Olivier Blanvillain, Fengyun Liu, Aggelos Biboudis
Heather Miller, Sandro Stucki
ACM SIGPLAN Symposium on Principles of Programming Languages

Function Passing: A Model for Typed, Distributed Functional Programming *SPLASH 2016*
Heather Miller, Philipp Haller, Normen Müller, Joceyln Boullier
ACM SIGPLAN International Symposium on New Ideas, New Paradigms, and Reflections on Programming & Software

Spores: A Type-Based Foundation for Closures in the Age of Concurrency and Distribution *ECOOP 2014*
Heather Miller, Philipp Haller, Martin Odersky
European Conference on Object Oriented Programming

Functional Programming For All! Scaling a MOOC for Students And Professionals Alike *ICSE 2014*
Heather Miller, Philipp Haller, Lukas Rytz, Martin Odersky
ACM SIGSOFT International Conference on Software Engineering

Instant Pickles: Generating Object-Oriented Pickler Combinators for Fast and Extensible Serialization *OOPSLA 2013*
Heather Miller, Philipp Haller, Eugene Burmako, Martin Odersky
ACM SIGPLAN Conference on Object Oriented Programming, Systems, Languages and Applications

Publications: Workshops

Distributed Programming via Safe Closure Passing *PLACES 2015*
Philipp Haller, Heather Miller
Programming Language Approaches to Communication and Concurrency Centric Systems

RAY: Integrating Rx and Async for Direct-Style Reactive Streams *REM 2013*
Philipp Haller, Heather Miller
ACM SPLASH Workshop on Reactivity, Events and Modularity

FlowPools: A Lock-Free Deterministic Concurrent Dataflow Abstraction *LCPC 2012*
Aleksandar Prokopec, Heather Miller, Tobias Schlatter,
Philipp Haller, Martin Odersky
International Workshop on Languages and Compilers for Parallel Computing

Invited to Revised Selected Papers on the 25th International Workshop on Languages and Compilers for Parallel Computing, Lecture Notes in Computer Science, Vol. 7760, 2013

Tools and Frameworks for Big Learning in Scala: Leveraging the Language for High Productivity and Performance *BigLearn 2011*
Heather Miller, Philipp Haller, Martin Odersky
NIPS Workshop on Parallel and Large-Scale Machine Learning

Parallelizing Machine Learning – Functionally: A Framework and Abstractions for Parallel Graph Processing *Scala 2011*
Philipp Haller, Heather Miller
Scala Workshop

Submitted/In Preparation

Monotonicity Types
Kevin Clancy, Heather Miller, Christopher Meiklejohn

The Essence of Coordination-Free Distributed Computation
Christopher Meiklejohn, Kevin Clancy, Heather Miller

Selected Tech Reports

The Function Passing Model: Types, Proofs, and Semantics *May 2016*
Philipp Haller, Normen Müller, Heather Miller

Specialising Parsers for Queries *April 2016*
Manohar Jonnalagedda, Jorge Vicente Cantero, Heather Miller, Martin Odersky

Improving Human-Compiler Interaction Through Customizable Type Feedback *December 2014*
Hubert Plociniczak, Heather Miller, Martin Odersky

Self-Assembly: Lightweight Language Extension and Datatype Generic Programming, All-in-One! *August 2014*
Heather Miller, Philipp Haller, Bruno C. d. S. Oliveira

Spores, Formally *December 2013*
Heather Miller, Philipp Haller

FlowPools: A Lock-Free Deterministic Concurrent Dataflow Abstraction – Proofs *June 2012*
Aleksandar Prokopec, Heather Miller, Philipp Haller

External Service

General Chair and/or Program Chair:
Curry On (**Curry On**) *2015, 2016, 2017, 2018, 2019*
Workshop on Principles and Practice of Consistency for Distributed Data (**PaPoC**) *2019*
Trends in Functional Programming in Education (**TFPIE**) *2018*
Scala Symposium (**Scala**) *2013, 2014, 2017*
Programming Models & Languages for Distributed Computation (**PMLDC**) *2016, 2017*

Organizing Committee Member:
Object-Oriented Programming, Systems, Languages & Applications (**OOPSLA**) *2018*
European Conference on Object-Oriented Programming (**ECOOP**) *2015 – 2019*

Program Committee Member:
Object-Oriented Programming, Systems, Languages & Applications (**OOPSLA**) *2019*
European Conference on Object-Oriented Programming (**ECOOP**) *2019*
Symposium on Principles of Programming Languages (**POPL**) *2019*

International Conference on Functional Programming (**ICFP**) *2018*
Off the Beaten Track (**OBT**) *2018*
Object-Oriented Programming, Systems, Languages & Applications (**OOPSLA**) *2017*
Scala Symposium (**Scala**) *2016*
Symposium on Trends in Functional Programming (**TFP**) *2016*
Software Language Engineering (**SLE**) *2016*
Symposium on Applied Computing (**SAC**) *2016*
Programming Language Evolution (**PLE**) *2015*
Domain-Specific Language Design and Implementation (**DSLDI**) *2015*

External Review Committee Member:
PLDI 2018, ECOOP 2016, ECOOP 2013, Scala 2013

Artifact Evaluation Committee: POPL 2015

Diversity & Outreach

ScalaBridge Organizer
Organizer of free full-day workshops on the weekends aimed at teaching women and underrepresented minorities in computing how to think computationally and how to program in Scala.

ScalaBridge Chapters: Basel (CH), Zürich (CH), Copenhagen (DK), Boston (US).

Open Source

Scala Programming Language, *member of the Scala team* *2011 –*

- **Scala Spores** (Scala Improvement Proposal SIP-21), ***project lead***
  novel type-based abstraction for using closures safely in concurrent and distributed environments
- **Scala Pickling**, ***project lead***
  novel framework for fast, boilerplate-free, extensible serialization. Adopted by sbt, the most widely-used build tool for Scala. Popular open-source project on GitHub with >820 stars & dozens of contributors
- **Scala Futures & Promises** (Scala Improvement Proposal SIP-14), ***team member***
  unified non-blocking concurrency substrate for Scala, Akka, Play, and others
- **Scala Documentation**, ***creator, writer, lead maintainer***
  a central website for community-driven documentation for the Scala programming language and core libraries
- **Scaladoc**, ***co-maintainer***
  documentation tool for Scala's official API documentation

Honors

US National Science Foundation Graduate Research Fellowship *2011 – 2014*
EPFL Outstanding Teaching Award *2012*
EPFL Computer Science Fellowship *2009 – 2010*
Most Outstanding Audio Engineering Student, University of Miami *2009*
Most Outstanding Eta Kappa Nu Student, University of Miami *2009*
Information Technology Scholarship, University of Miami *2006 – 2009*
John Farina Family Scholarship, University of Miami *2006 – 2009*
Eta Kappa Nu *2008*

Tau Beta Pi	*2008*
SMART US Department of Defense Scholarship Alternate	*2007*
Cooper Union Full Tuition Scholarship	*2004 – 2006*

Selected Talks

Towards Language Support for Distributed Systems — *Code Mesh 2018* ***(invited)***
London, UK. November 9, 2018

What Happened to Distributed Programming Languages? — *SPLASH-I 2018* ***(invited)***
Boston, MA, USA. November 6, 2018

Towards Language Support for Distributed Systems — *Strange Loop 2018*
St. Louis, MO, USA. September 27, 2018

I'm a Young Assistant Professor: AMA. + Heather's Unsolicited Advice About Grad School — *PLMW 2018* ***(invited)***
St. Louis, MO, USA. September 23, 2018

We're Building On Hollowed Foundations: Worrying Trends in Open Source and What You Can Actually Do About It — *Lambda Days 2018* ***(keynote)***
Krakow, Poland. February 22, 2018

The Dramatic Consequences of the Open Source Revolution: Unrecognized Challenges & Some Modest Attempts at Solutions in Scala — *Devoxx 2017* ***(invited)***
Paris, France. April 7, 2017

The Dramatic Consequences of the Open Source Revolution & How the Scala Center Hopes to Help — *Scala Exchange 2016* ***(keynote)***
London, UK. December 9, 2016

Function Passing: A Model for Typed, Distributed Functional Programming — *SPLASH 2016*
Amsterdam, The Netherlands. November 2, 2016

Introducing the Scala Center — *Scala Days 2016* ***(keynote)***
New York, NY, US. May 10, 2016 & Berlin, Germany. June 16, 2016
(total ~1700 attendees)

Function Passing Style: Typed, Distributed Functional Programming — *Strange Loop 2014*
St. Louis, MO, USA. September 19, 2014

Spores: A Type-Based Foundation for Closures in the Age of Concurrency and Distribution — *ECOOP 2014*
Uppsala, Sweden. August 1, 2014

Functional Programming For All! Scaling a MOOC for Students and Professionals Alike — *ICSE 2014*
Hyderabad, India. June 4, 2014

Academese to English: Scala's Type System, Dependent Types — *NEScala 2014*

and What It Means To You
New York, NY, USA. March 1, 2014

Instant Pickles: Generating Object-Oriented Pickler Combinators for Fast and Extensible Serialization — *OOPSLA 2013*
Indianapolis, IN, USA. October 30, 2013

PL Abstractions for Distributed Programming: Pickle Your Spores! — *Indiana University* ***(invited)***
Bloomington, IN, USA. October 25, 2013

Spores: Distributable Functions in Scala — *Strange Loop 2013*
St. Louis, MO, USA. September 19, 2013

Open Issues in Dataflow Programming — *LaME 2013* ***(invited)***
Montpellier, France. July 1, 2013

Scala as a Research Tool — *ECOOP 2013 Tutorial*
Montpellier, France. July 1, 2013

On Pickles & Spores: Improving Scala's Support for Distributed Programming — *ScalaDays 2013*
New York, NY, USA. June 12, 2013

Futures & Promises in Scala 2.10 — *PhillyETE 2013* ***(invited)***
Philadelphia, PA, USA. April 2, 2013

I am also a frequent speaker in industry, at industrial conferences, developer "meet-ups", and everything in between. Some such events include:

Scala Italy (9/2018, Florence, Italy), **LxScala** (6/2018, Lisbon, Portugal), **Open Source Summit** (12/2017, Paris, France), **Scala World** (9/2017, Lake District, UK), **LxScala** (5/2017, Lisbon, Portugal), **Lambda Days** (2/2017, Krakow, Poland), **PhillyETE** (4/2016, Philadelphia, USA), **Code Mesh** (11/2015, London, UK), **Scalar** (4/2015, Warsaw, Poland), **f(by)** (11/2014, Minsk, Belarus), **SF Scala** (11/2014, SF, USA), **Scalapeño** (9/2014, Tel Aviv, Israel), **SoundCloud TechTalks** (7/2014, Berlin, Germany), **Scala Days** (6/2014, Berlin, Germany), **NEScala** (3/2014, NYC, USA), amongst others.

External Activities

Scalawags Monthly Podcast, co-host — *2014 – 2016*

Students Supervised

Zeeshan Lakhani, *TBD* — *2018 –*
PhD thesis — *Carnegie Mellon*

Christopher Meiklejohn, *TBD* — *2018 –*
PhD thesis — *Carnegie Mellon*

Kevin Clancy, *Eventual Consistency via Types* — *2017 –*
PhD thesis — *Northeastern*

Joceyln Boullier, *Evaluating the Efficiacy of the Function Passing Model* — *2/2016 – 8/2016*
M.Sc. thesis — *EPFL*

Jorge Vicente Cantero, *Implementing the Function Passing Model* — *2/2016 – 6/2016*
B.Sc. thesis — *EPFL*

Thaddée Yann Tyl, *Learning Scala Style* — *2/2013 – 6/2013*
M.Sc. thesis — *EPFL*

References

Martin Odersky, Professor
École Polytechnique Fédérale de Lausanne
☎ +41 21 693 68 63
✉ martin.odersky@epfl.ch

Matthias Felleisen, Trustee Professor
Northeastern University
☎ +1-617-373-2085
✉ matthias@ccs.neu.edu

Jan Vitek, Professor
Northeastern University
☎ +1-617-749-8148
✉ j.vitek@northeastern.edu

Matei Zaharia, Assistant Professor
Stanford University
☎ +1-510-610-0001
✉ matei@cs.stanford.edu

Philipp Haller, Assistant Professor
KTH Royal Institute of Technology
☎ +46 70 738 28 43
✉ phaller@kth.se