

James Bornholt

📍 2317 Speedway, Stop D9500, Austin, TX 78712-1757

✉ bornholt@cs.utexas.edu

🌐 <https://www.cs.utexas.edu/~bornholt/>

Employment

| | |
|---|-------------------------------|
| University of Texas at Austin Assistant Professor, Department of Computer Science | Austin, TX, USA 2021– |
| Amazon Web Services Visiting Academic | Austin, TX, USA 2021– |
| Amazon Web Services Senior Applied Scientist, S3 | Seattle, WA, USA 2019–2021 |
| Amazon Web Services Research Intern, Automated Reasoning Group | Seattle, WA, USA 2018 |
| Microsoft Research Software Engineer, Research in Software Engineering (RiSE) group | Canberra, Australia 2014 |
| Microsoft Research Research Intern, Research in Software Engineering (RiSE) group | Seattle, WA, USA 2012–2013 |
| Microsoft Research Research Intern, Research in Software Engineering (RiSE) group | Seattle, WA, USA 2011–2012 |

Education

| | |
|---|----------------------------------|
| University of Washington Ph.D., Computer Science & Engineering <ul style="list-style-type: none">• Advisors: Emina Torlak, Dan Grossman, Luis Ceze• Thesis: Optimizing the Automated Programming Stack | Seattle, WA, USA 2014–2019 |
| University of Washington M.S., Computer Science & Engineering | Seattle, WA, USA 2016 |
| Australian National University Bachelor of Philosophy (Honours) <ul style="list-style-type: none">• Advisor: Steve Blackburn• First Class Honours and the University Medal in Computer Science | Canberra, Australia 2010–2013 |

Publications

Conference Papers

Vectorization for Digital Signal Processors via Equality Saturation. Alexa VanHattum, Rachit Nigam, Vincent T. Lee, James Bornholt, and Adrian Sampson. ASPLOS 2021.

A Synthesis-Aided Compiler for DSP Architectures. Alexa VanHattum, Rachit Nigam, Vincent T. Lee, James Bornholt, and Adrian Sampson. LCTES 2020.

Automatic Generation of High-Performance Quantized Machine Learning Kernels. Meghan Cowan, Thierry Moreau, Tianqi Chen, James Bornholt, and Luis Ceze. CGO 2020.

Fixing Code That Explodes Under Symbolic Evaluation. Sorawee Porncharoenwase, James Bornholt, and Emina Torlak. VMCAI 2020.

Scaling Symbolic Evaluation for Automated Verification of Systems Code with Serval. Luke Nelson, James Bornholt, Ronghui Gu, Andrew Baumann, Emina Torlak, and Xi Wang. SOSP 2019. **Best Paper Award. Distinguished Artifact Award.**

Finding Code That Explodes Under Symbolic Evaluation. James Bornholt and Emina Torlak. OOPSLA 2018. **Distinguished Artifact Award.**

Nickel: A Framework for Design and Verification of Information Flow Control Systems. Helgi Sigurbjarnarson, Luke Nelson, Bruno Castro-Karney, James Bornholt, Emina Torlak, and Xi Wang. OSDI 2018.

Hyperkernel: Push-Button Verification of an OS Kernel. Luke Nelson, Helgi Sigurbjarnarson, Kaiyuan Zhang, Dylan Johnson, James Bornholt, Emina Torlak, and Xi Wang. SOSP 2017.

Synthesizing Memory Models from Framework Sketches and Litmus Tests. James Bornholt and Emina Torlak. PLDI 2017.

Push-Button Verification of File Systems via Crash Refinement. Helgi Sigurbjarnarson, James Bornholt, Emina Torlak, and Xi Wang. OSDI 2016. **Best Paper Award.**

Disciplined Inconsistency with Consistency Types. Brandon Holt, James Bornholt, Irene Zhang, Dan R. K. Ports, Mark Oskin, and Luis Ceze. SoCC 2016.

Specifying and Checking File System Crash-Consistency Models. James Bornholt, Antoine Kaufmann, Jialin Li, Arvind Krishnamurthy, Emina Torlak, and Xi Wang. ASPLOS 2016.

A DNA-Based Archival Storage System. James Bornholt, Randolph Lopez, Douglas M. Carmean, Luis Ceze, Georg Seelig, and Karin Strauss. ASPLOS 2016. **IEEE Micro Top Picks.**

Optimizing Synthesis with Metasketches. James Bornholt, Emina Torlak, Dan Grossman, and Luis Ceze. POPL 2016.

Hardware-Software Co-Design: Not Just a Cliché. Adrian Sampson, James Bornholt, and Luis Ceze. SNAPL 2015.

Uncertain<T>: A First-Order Type for Uncertain Data. James Bornholt, Todd Mytkowicz, and Kathryn S. McKinley. ASPLOS 2014. **SIGPLAN Research Highlight. IEEE Micro Top Picks.**

Journal Papers

Noninterference Specifications for Secure Systems. Luke Nelson, James Bornholt, Arvind Krishnamurthy, Emina Torlak, and Xi Wang. ACM SIGOPS Operating Systems Review, vol. 54, no. 1, pp. 31–39, July 2020.

A Taxonomy of General Purpose Approximate Computing Techniques. Thierry Moreau, Joshua San Miguel, Mark Wyse, James Bornholt, Armin Alaghi, Luis Ceze, Natalie Enright Jerger, and Adrian Sampson. IEEE Embedded Systems Letters, vol. 10, no. 1, pp. 2–5, March 2018.

Toward a DNA-Based Archival Storage System. James Bornholt, Randolph Lopez, Douglas M. Carmean, Luis Ceze, Georg Seelig, and Karin Strauss. IEEE Micro, vol. 37, no. 3, pp. 98–104, May–June 2017.

Uncertain<T>: Abstractions for Uncertain Hardware and Software. James Bornholt, Todd Mytkowicz, and Kathryn S. McKinley. IEEE Micro, vol. 35, no. 3, pp. 132–143, May–June 2015.

Workshop Papers

Scaling Program Synthesis by Exploiting Existing Code. James Bornholt and Emina Torlak. ML4PL 2015 (colocated with ECOOP 2015).

Approximate Program Synthesis. James Bornholt, Emina Torlak, Luis Ceze, and Dan Grossman. WAX 2015 (colocated with PLDI 2015).

REACT: A Framework for Rapid Exploration of Approximate Computing Techniques. Mark Wyse, Andre Baixo, Thierry Moreau, Bill Zorn, James Bornholt, Adrian Sampson, Luis Ceze, and Mark Oskin. WAX 2015 (colocated with PLDI 2015).

Programming the Internet of Uncertain <T>hings. James Bornholt, Na Meng, Todd Mytkowicz, and Kathryn S. McKinley. SCAW 2015 (colocated with HPCA 2015).

There's Something About Bayes: Effective Probabilistic Programming for the Rest of Us. James Bornholt, Todd Mytkowicz, and Kathryn S. McKinley. APPROX 2014 (colocated with PLDI 2014).

Posters & Talks

Uncertain<T>: A First-Order Type for Uncertain Data. James Bornholt. PLDI 2013 Student Research Competition. **First Place, PLDI Student Research Competition. Second Place, ACM Student Research Competition Grand Final.**

The Model Is Not Enough: Understanding Energy Consumption in Mobile Devices. James Bornholt, Todd Mytkowicz, and Kathryn S. McKinley. Hot Chips 24, 2012.

Theses

Optimizing the Automated Programming Stack. James Bornholt. PhD thesis, University of Washington, 2019.

Abstractions and Techniques for Programming with Uncertain Data. James Bornholt. Honours thesis, Bachelor of Philosophy (Honours), Australian National University, 2013.

Awards

| | |
|---|-----------|
| SOSP Best Paper Award | 2019 |
| OOPSLA Distinguished Artifact Award | 2018 |
| Facebook Ph.D. Fellowship | 2018–2020 |
| IEEE Micro Top Picks from the Computer Architecture Conferences, for DNA storage | 2017 |
| OSDI Jay Lepreau Best Paper Award | 2016 |
| IEEE Micro Top Picks from the Computer Architecture Conferences, for Uncertain<T> | 2015 |
| ACM SIGPLAN Research Highlight, for Uncertain<T> | 2014 |
| David Notkin Endowed Graduate Fellowship, University of Washington | 2014–2015 |
| Second Place, ACM Student Research Competition Grand Finals (undergraduate) | 2014 |
| First Place, ACM PLDI Student Research Competition (undergraduate) | 2013 |
| ANU University Medal for Computer Science | 2013 |

Teaching

| | |
|---|--|
| CS 395T, Systems Verification and Synthesis Instructor | University of Texas at Austin Spring 2021 |
| CSE 507, Computer Aided Reasoning for Software Teaching Assistant | University of Washington Winter 2017 |
| CSE 507, Computer Aided Reasoning for Software Teaching Assistant | University of Washington Spring 2016 |

Service

Review Committees

| | |
|--|------|
| Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA) — External Review Committee | 2021 |
| Operating Systems Design and Implementation (OSDI) — Program Committee | 2021 |

| | |
|--|------|
| Architectural Support for Programming Languages and Operating Systems (ASPLOS) — Program Committee | 2021 |
| Programming Languages Design and Implementation (PLDI) — Program Committee | 2020 |
| Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA) — Review Committee | 2020 |
| Architectural Support for Programming Languages and Operating Systems (ASPLOS) — External Review Committee | 2020 |
| Principles and Practice of Parallel Programming (PPoPP) — Brief Announcements Program Committee | 2020 |
| Formal Techniques for Java-Like Programs (FTfJP) — Program Committee | 2019 |
| Programming Languages Design and Implementation (PLDI) — External Review Committee | 2017 |
| Computer-Aided Verification (CAV) — Artifact Evaluation Committee | 2017 |
| Principles of Programming Languages (POPL) — Artifact Evaluation Committee | 2016 |
| Programming Languages Design and Implementation (PLDI) — Artifact Evaluation Committee | 2015 |

External Reviewing

| | |
|--|------|
| Architectural Support for Programming Languages and Operating Systems (ASPLOS) | 2018 |
| IEEE Transactions on Emerging Topics in Computing | 2017 |
| Computer-Aided Verification (CAV) | 2015 |
| ACM Transactions on Embedded Computing | 2015 |
| Architectural Support for Programming Languages and Operating Systems (ASPLOS) | 2015 |

Department Service

| | |
|--|-----------|
| Graduate Admissions Committee, University of Washington | 2017–2019 |
| Prospective Student Committee Co-Chair, University of Washington | 2016 |
| Prospective Student Committee, University of Washington | 2015–2019 |

Presentations and Seminars

Program Synthesis in the Small

| | |
|---------------------------------|----------|
| Purdue University, Invited Talk | Oct 2020 |
|---------------------------------|----------|

Synthesizing System Specifications

| | |
|---|----------|
| State of the Art in Program Synthesis, Invited Talk | Sep 2019 |
|---|----------|

Optimizing the Automated Programming Stack

| | |
|---|----------|
| Australian National University, Invited Talk | Jul 2019 |
| University of Toronto, Invited Seminar | Apr 2019 |
| Princeton University, Invited Seminar | Apr 2019 |
| University of British Columbia, Invited Seminar | Apr 2019 |
| École Polytechnique Fédérale de Lausanne, Invited Seminar | Apr 2019 |
| University of Massachusetts Amherst, Invited Seminar | Apr 2019 |
| Northeastern University, Invited Seminar | Mar 2019 |
| Microsoft Research, Invited Seminar | Mar 2019 |
| Georgia Institute of Technology, Invited Seminar | Mar 2019 |
| University of California, Berkeley, Invited Seminar | Feb 2019 |
| Brown University, Invited Seminar | Feb 2019 |
| Carnegie Mellon University, Invited Seminar | Feb 2019 |
| University of Maryland, College Park, Invited Seminar | Feb 2019 |

| | |
|--|----------|
| University of Texas at Austin, Invited Seminar | Feb 2019 |
| Cornell University, Invited Seminar | Jan 2019 |
| <i>Finding Code That Explodes Under Symbolic Evaluation</i> | |
| University of California, Santa Cruz, Guest Lecture | Nov 2019 |
| OOPSLA, Conference Talk | Nov 2018 |
| Galois, Invited Talk | Jun 2018 |
| <i>Ocelot: Relational Logic in a Solver-Aided Language</i> | |
| Future of Alloy Workshop, Invited Talk | Apr 2018 |
| <i>Synthesizing Memory Models from Framework Sketches and Litmus Tests</i> | |
| PLDI, Conference Talk | Jun 2017 |
| <i>Programming with Estimates</i> | |
| Programming Languages Mentoring Workshop, Invited Talk | Jun 2016 |
| <i>Specifying and Checking File-System Crash Consistency Models</i> | |
| DARPA BRASS PI Meeting, Invited Talk | Jul 2016 |
| ASPLOS, Conference Talk | Apr 2016 |
| <i>A DNA-Based Archival Storage System</i> | |
| ASPLOS, Conference Talk | Apr 2016 |
| <i>Optimizing Synthesis with Metasketches</i> | |
| POPL, Conference Talk | Jan 2016 |
| Dagstuhl Seminar 15491 (Approximate and Probabilistic Computing), Invited Talk | Dec 2015 |