

Eric Bond | Resume

1100 University St – Seattle Washington – US

+1 (317) 667 7195 • bond.eric.23@gmail.com

[linkedin.com/in/ericbond1](https://www.linkedin.com/in/ericbond1) - github.com/bond15 - twitter.com/EricBond10

Education

Purdue University

M.S. Computer Science

Incomplete 8-2019

West Lafayette, IN

2018-2019

Purdue University

B.S. Computer Science

Minor in Mathematics

West Lafayette, IN

2013-2017

Research Interests

Type Theory, Formalized Mathematics, Applied Category Theory, Programming Language Semantics

Research Experience

Two Six Technologies

Lead Research Scientist

Seattle, WA

April 2021 - Current

- Worked as a member of the Applied Mathematics team on several DARPA research programs including DPRIVE, SafeDocs, and VSPELLS.
- Verified optimizations for high bit width arithmetic primitives for FHE accelerators in Isabelle/HOL.
- Formalized colored operads in Coq to reason about composition of DSLs.
- Identified research opportunities and assisted principal researchers with proposal writing.
- Languages/Technologies: Isabelle/HOL, Coq, Agda, Haskell.

AMS Mathematical Research Communities - Applied Category Theory

Participant

Buffalo, NY

May 2022

- Member of Valeria de Paiva's group investigating applications of Dialectica Categories.
- Leading a subgroup interested in formalizations of Dialectica Categories in Agda and computational implementations in Catlab.

Masters Thesis (Incomplete)

Purdue University

November 2018 – August 2019

- Interactive proof assistants represent inductive data and induction principals differently depending on the underlying type theory. Coq utilizes a complicated implementation of an intrinsic calculus of inductive constructions. Cedille is a proof assistant based on a pure extrinsic type theory with a notably small core implementation in which induction principles are generically derivable for data and computation represented as lambda encoded F-algebras.
- We are investigating transformation of proof developments between these two proof assistants to study trustworthiness of formal verifications in either type theory.
- Collaborators/Advisers: Prof. Aaron Stump (Iowa), Prof. Benjamin Delaware (Purdue)

Galois

Graduate Research Intern

Portland, OR

May 2019 – August 2019

- Contributed to projects aimed at delivering provably secure software via formal methods.
- Designed and implemented an automata-based API usage checker for the Crucible symbolic simulator.
- Languages/Technologies: Haskell, LLVM, Crucible

Lawrence Livermore National Lab*Graduate Research Intern***Livermore, CA***May 2018 – August 2018*

- Member of the 2018 Data Science Summer Institute.
- Participated in a quantum computing working group.
- Built a pipeline to generate simulation data for quantum chemical systems using VASP.
- Prototyped an extension of TopoMS, a tool for topological data analysis of molecular systems, which allowed for the analysis of potential fields.
- Languages/Technologies:: C++, Python, Keras, TopoMS, VASP, MPI

Quantum Information and Computation Theory Group*Undergraduate Researcher***Purdue University***August 2017 – October 2018*

- Investigated mathematical formalizations for quantum programming languages for the quantum gate/circuit model.
- Surveyed various existing languages and frameworks such as Quantomatic, Quipper, QASM, QWire, Qiskit and others.

Collaborative Robotics Laboratory*Undergraduate Researcher NSF REU***Purdue University***January 2017 – June 2017*

- Researched and implemented methods for task based self-adaptation reasoning using an RDF ontology (Apache Jena) and semantic networks.
- Publications
 - A Software Architecture Supporting Self-Adaptation of Wireless Control Networks, IEEE CASE 2017

Work Experience

47 Degrees*Software Engineer 2 - Functional Programming Consultant***Seattle WA***January 2020 – April 2021*

- Redesigned existing services and drove development of new functional services while teaching engineers new functional programming patterns.
- Helped multiple clients adopt Scala (a la Haskell via the Typelevel stack) for cleaner and safer development via strong typing and functional abstractions.
- Languages/Technologies:: Scala, Cats (Haskell Typeclasses in Scala), Typelevel stack, OpenAPI, AWS, GCP, Docker.

Amazon - (Alexa Machine Learning Platform)*Software Development Engineer Intern***Cambridge, MA***May 2017 – August 2017*

- Designed and implemented a data access and visualization tool for scientists and engineers to view critical data for machine learning applications/training.
- Languages/Technologies:: Java, Javascript, AWS (S3, DynamoDB, Lambda), Mockito

Amazon - (Fulfillment by Amazon Technologies)*Software Development Engineer Intern***Seattle, WA***June 2016 – August 2016*

- Extended an FBA seller central web applications with a widget for aggregating and displaying shipping details.
- Languages/Technologies:: Java, Javascript, Spring Framework, Mockito

Teaching Experience

CS 252 - Systems Programming*Graduate Teaching Assistant***Purdue University***January 2019 – May 2019*

- Taught students entry-level systems programming in C/C++ through labs on creating a web server, memory allocator, and shell.