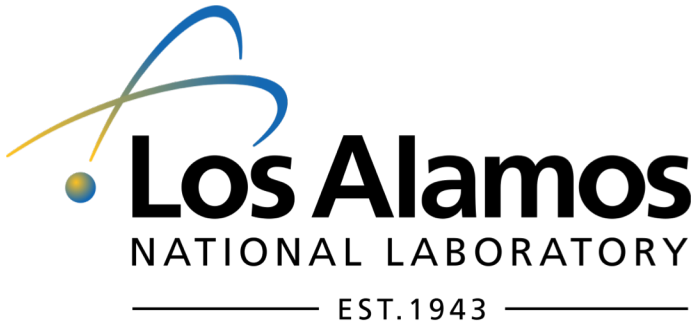
< AIDE | QC >

Advancing Integrated Development Environments for Quantum Computing through Fundamental Research

Team director: Bert de Jong (LBNL)

Team deputy director: Travis Humble (ORNL)

Team



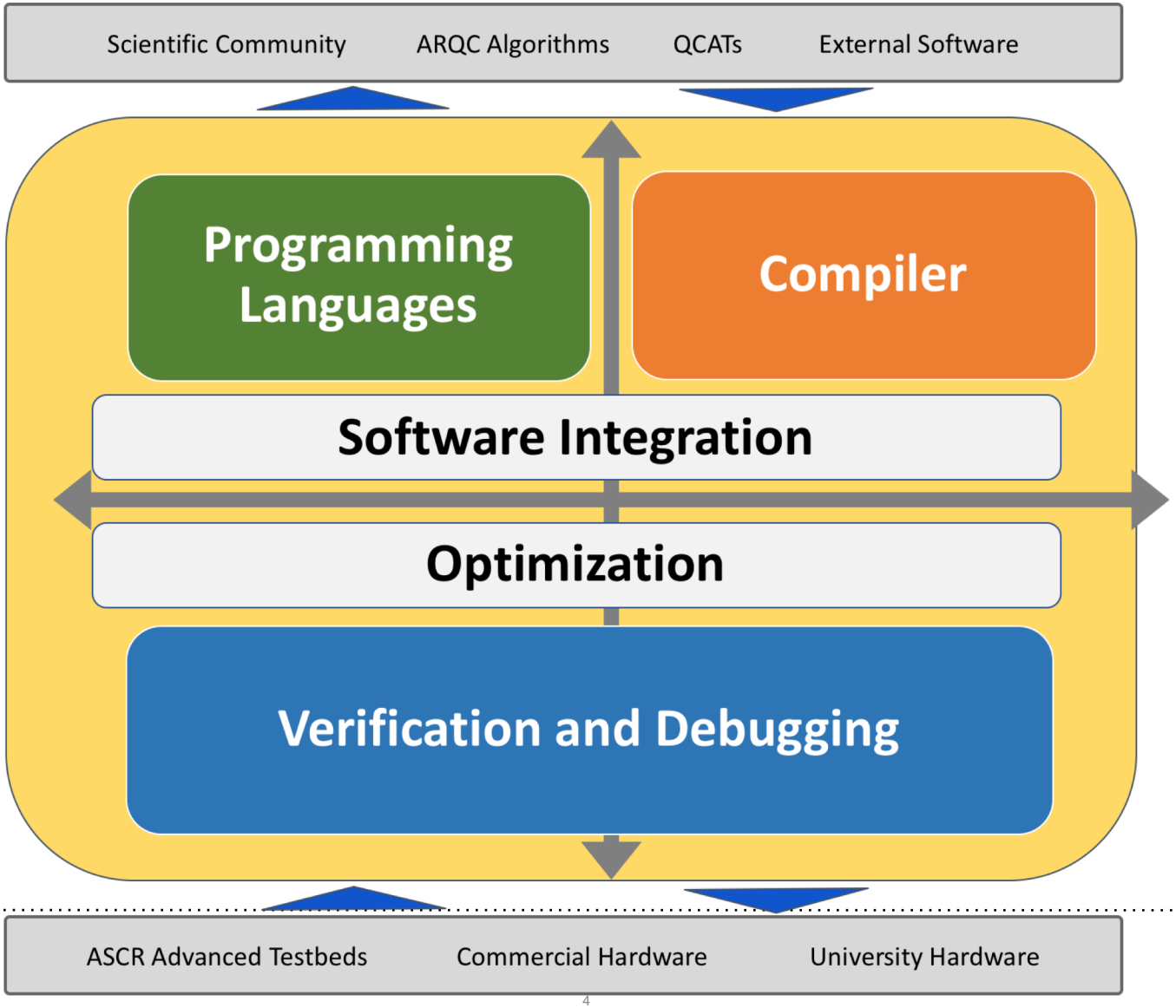
THE UNIVERSITY OF
CHICAGO

Our mission

Our goal is to develop and deliver a robust open-source software environment that supports scientific breakthroughs by the DOE quantum computing (QC) community.

By addressing key aspects in computer science research that accelerate the programming of near-term, intermediate-scale quantum (NISQ) devices for scientific exploration.

Overview of AIDE-QC project



Research thrust areas

Programming Languages. Advancing high-level programming models, languages and libraries for heterogeneous quantum/classical, digital/analog computation

Compiler. Deploying platform agnostic compilers for NISQ devices that deliver optimal program synthesis while incorporating error mitigation strategies.

Verification and Debugging. Developing techniques and tools for verification and debugging of quantum and hybrid quantum-classical programs.

Cross-cutting thrust areas

Optimization. Developing critically important optimization algorithms needed across the thrust areas to fully exploit the limited resources afforded by NISQ hardware.

Software Integration. Establishing an integrated software infrastructure for programming, error mitigation, executing, classical simulation and analyzing quantum computations.

Our organization

DOE/ASCR
Point of Contact
Ceren Susut

Project Director
Wibe Albert de Jong (LBNL)

Deputy Director
Travis Humble (ORNL)

LBNL Stewardship
Kathy Yelick (ALD)
David Brown (CRD Director)

Programming Models
Lead: Tiffany Mintz (ORNL)
de Jong, Lavrijsen (ORNL)
Humble, Pooser, Elwasif (ORNL)
Finkel, Hovland (ANL)

Compiler
Lead: Costin Iancu (LBNL)
Lavrijsen, Yang, Carter (LBNL)
McCaskey, Mintz, Moore (ORNL)
Suchara, Finkel (ANL)
Cincio, Coles (LANL)
Chong (UChicago)

Verification and Debugging
Lead: Mohan Sarovar (SNL)
Yao, Lavrijsen, Mueller (LBNL)
Bennink, Dumitrescu, Poorser (ORNL)
Witzel, Grace, Hulette, Aytac (SNL)
Otten, Hovland (ANL)

Optimization
Cross-cut Lead: Stefan Wild (ANL)
Mueller, Yang, Lin (LBNL); Peters (ORNL); Chong (UChicago)
Ridzal, von Winckel (SNL))

Software Integration
Cross-cut Lead: Alex McCaskey (ORNL)
Austin, Carter (LBNL), Liakh, Bennink, Elwasif, Humble (ORNL)
Finkel, Hovland (ANL), Chong (UChicago)

BERKELEY LAB

Programming Languages Thrust

Goal:

- To enable high-level quantum programming across multiple scientific domains through the the development of library based programming language extensions for single source implementations of quantum-classical applications

Proposed Work:

- Investigate programming patterns and data usage of common numerical and computational motifs that may be relevant to quantum problems
- Develop libraries that implement quantum algorithms across a hierarchy of programming abstractions
- Design a progressive interface that enables the expression of known quantum algorithms as well as imminent algorithms across multiple scientific domains
- Enable extensible data capture enabling program diagnostics and postmortem debugging & analysis

Compiler Thrust

Circuit Synthesis • Scalable synthesis -> numerical optimization, tensor decomposition) • Global circuit optimization using synthesis -> V&V, mapping, algs • Hardware design exploration (gates, topology) -> apps • Error aware synthesis/optimization -> V&V, mapping, ML, optimization	**Pulse Level Optimization** • Incorporating noise models into optimization -> V&V, apps • Hierarchical circuit optimization -> algs, V&V, mapping
Routing and Mapping • Hybrid automated reasoning -> V&V,algs • Hierarchical circuit optimization -> algs	**C++ Compiler** • Language design, programming API • Integration of other technologies -> synthesis, mapping, error mitigation, pulse optimization

BERKELEY LAB

Verification and Debugging Thrust

Goal: develop techniques and tools for verification of quantum computer performance and real-time debugging of quantum programs.

Verification

- Verification of translation from algorithm to circuit (compilation)
- Verification of algorithm/circuit performance on hardware
- Verification of large-scale hardware (e.g., >1000 qubits)

Debugging

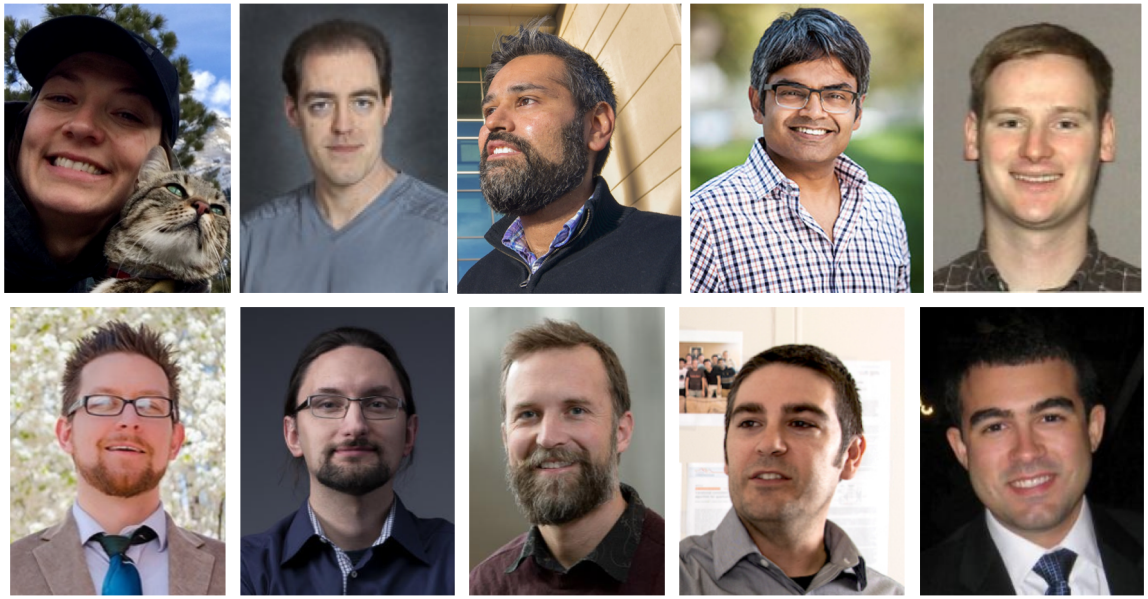
- Exploiting symmetry is quantum programs to monitor “parity” operators
- Variational quantum compiling with symmetry in mind

Optimization Crosscutting Thrust

- Developing and applying numerical optimization algorithms throughout the quantum computer science stack
- Designed to maximize impact of limited resources afforded by NISQ devices

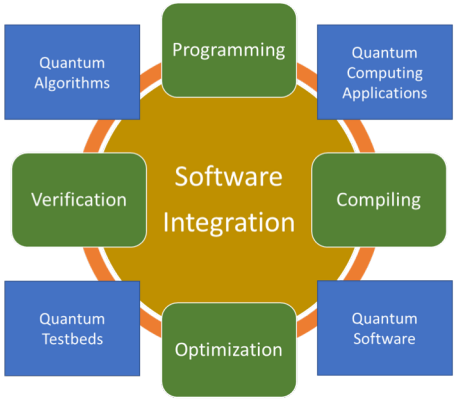
Team members:

Julie Mueller
Wim Lavrijsen
Ojas Parekh
Mohan Sarovar
Denis Ridzal
Greg von Winckel
Lukasz Cincio
Stefan Wild
Patrick Coles (liaison)
Alex McCaskey (liaison)



BERKELEY LAB

Software Integration Thrust



Goal:

- Deliver an integrated, wholistic programming environment that enables future domain scientists to leverage quantum computers for scientific discovery

Proposed Work:

- Single-source C++ approach, extend to high-level application languages through appropriate language bindings
- Modular, service-oriented architecture. Treat workflow steps as black-box.
- Quantum hardware-agnostic
- Define standard interfaces across programming, compilation, and execution workflow
- State-of-the-art integration framework, promote interoperability with existing approaches

Summary

Thrust areas:

- Programming languages
- Compilers
- Verification and debugging
- Optimization
- Software integration

We will not be reinventing the wheel, we will focus on integrating the best available and expand/improve!