A common compiler ecosystem for domain specific languages

Citation for published version:
Brown, N 2023, 'A common compiler ecosystem for domain specific languages', ISC High Performance 2023, Hamburg, Germany, 21/05/23 - 25/05/23.

Link:
Link to publication record in Edinburgh Research Explorer

Document Version:
Peer reviewed version

General rights
Copyright for the publications made accessible via the Edinburgh Research Explorer is retained by the author(s) and / or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy
The University of Edinburgh has made every reasonable effort to ensure that Edinburgh Research Explorer content complies with UK legislation. If you believe that the public display of this file breaches copyright please contact openaccess@ed.ac.uk providing details, and we will remove access to the work immediately and investigate your claim.

Download date: 05. May. 2024
What are Domain Specific Languages (DSLs)?

There is an explosion of hardware architectures for HPC, and as we move into the exascale era a key challenge is how to fully exploit such complex, highly parallel heterogeneous supercomputers.

Why are DSLs not ubiquitous in HPC?

A major disadvantage is in the siloing of compiler infrastructure, where DSLs tend to share very little or no infrastructure between them at the toolchain level. This siloing of toolchains results in:

- Uncertainty around long term maintenance
- Reinventing the wheel
- Limited opportunities for DSLs to target multiple domains
- Considerable effort needed to develop a DSL
- Considerable effort for new architectures

We must solve the challenges around siloing in order for DSLs to become more widespread. Hence our vision: A DSL being a thin layer atop an existing, mature ecosystem with a wealth of third party tools.

What is our xDSL ecosystem?

A Python toolbox based upon MLIR, integration with MLIR & series of HPC dialects

Many application domains can benefit from using DSLs on HPC

Many dialects are provided which can be mixed and manipulated

There are numerous third party tools for LLVM -IR, such as profilers and debuggers that DSL developers get for free

Our Python toolbox exposes MLIR dialects and utilities in a high productivity manner

MLIR dialects are translated into LLVM-IR which provides backends for many hardware architectures, or alternatively C or Fortran

Our ecosystem provides:

- Composability where DSL owners choose what parts of our ecosystem to leverage
- Interoperability between DSLs
- Code reuse of toolchain infrastructure
- Longevity of DSL compiler technology as we build upon LLVM and MLIR

We have selected two existing DSLs to act as evaluators

- Devito is a Python based DSL for symbolic finite difference computation. Especially popular for seismological simulations, there are a variety of users from oil & gas exploration to glaciologists
- PSyclone is a Fortran based DSL for developing weather and climate codes. It is used extensively by the Met Office

Project running August 2021 until July 2024

- Initial PSyclone toolchain
- Initial Devito toolchain
- Portability demonstrated for both application and DSL developers due to code reuse
- Performance because dialects and backends are developed by experts
- Productivity for both the application and DSL developers due to code reuse
- Productivity for application codes and DSLs across architectures.

Above illustrates translating the PSyclone DSL’s Fortran code into LLVM-IR using our approach

Building on LLVM and MLIR

LLVM is a collection of common compilation tools and infrastructure. There are numerous LLVM backends available for different hardware, targeted via LLVM-IR.

MLIR enables representing and mixing dialects of intermediate representations and abstractions, thus providing easier integration, reuse, and optimisation. However, currently written in C++ there is a fairly steep learning curve that our Python toolbox looks to address.

Evaluating the xDSL framework

We have selected two existing DSLs to act as evaluators

- PSyclone is a Fortran based DSL for developing weather and climate codes. It is used extensively by the Met Office
- Devito is a Python based DSL for symbolic finite difference computation. Especially popular for seismological simulations, there are a variety of users from oil & gas exploration to glaciologists

Funded by ExCALIBUR

The UK ExCALIBUR program address the challenges and opportunities offered by computing at the exascale and aims to deliver the next generation of HPC simulation software and tooling