CHAIR FOR COMPILER CONSTRUCTION

cfaed CENTER FOR ADVANCING ELECTRONICS DRESDEN

Domain specific languages and compilation flows in the EVEREST project

Jeronimo Castrillon
Chair for Compiler Construction (CCC)
TU Dresden, Germany

Workshop on dEsign enVironmEnt foR Extreme-Scale big data analyTics on heterogeneous platforms (EVEREST)
Budapest, Hungary
June 22, 2022

EVEREST EU Project

- EVEREST: dEsign enVironmEnt foR Extreme-Scale big data analyTics on heterogeneous platforms

IBM Reseach Lab, Zurich (Switzerland)
Project Administration, Prototype of the target system

Università della Svizzera italiana (Switzerland)
Data security requirements and protection techniques

Centro Internazionale di Monitoraggio Ambientale (Italy) Weather prediction models

Virtual Open Systems (France)
Virtualization techniques, runtime extensions to manage heterogeneous resources

Numtech (France)
Application for monitoring the air quality of industrial sites

Politecnico di Milano (Italy)
Project Administration, High-Level System, Flexbile Memory Manager, Autotuning

TU Dresden (Germany)
Domain-specific extensions, code optimizations and variants

IT4Innovations (Czech Republic)
Exploitation leaders, Large HPC infrastructure, Workflow libraries

Duferco Energia (Italy)
Application for prediction of renewable energies

Sygic A/S (Slovakia)
Application for intelligent transportation

EVEREST

https://everest-h2020.eu

EVEREST use cases

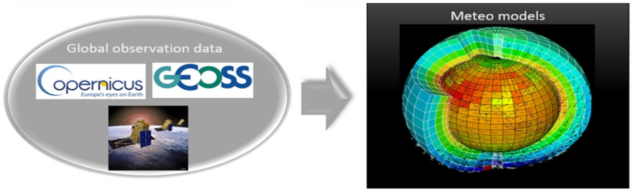
- ❑ Challenging use cases
  - ❑ Air-quality modelling in industrial sites
  - ❑ Weather-based prediction of renewable energy production
  - ❑ Traffic modelling for intelligent transportation

➔ Data driven applications: Industrial and societal impact!

- ❑ HW acceleration as a constant request
- ❑ All have ML-based components
- ❑ All have complex pipelines not just a single task processing

Air quality: Example

1. WRF Deterministic weather forecast



Global observation

WRF execution

Improve speed to produce forecast

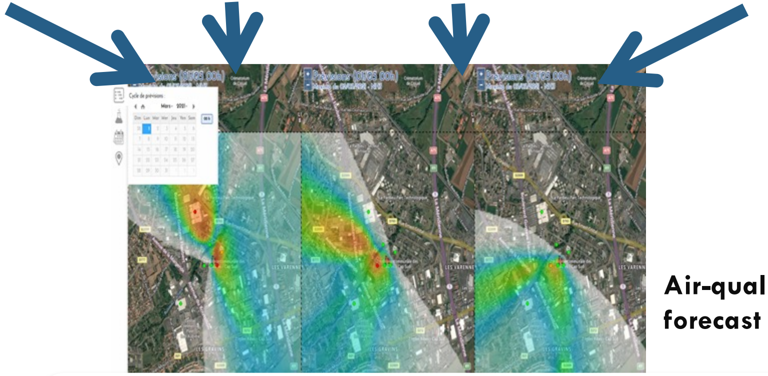
3. Air-quality dispersion forecast

Landuse, Topography

Industrial site data

Emission forecast

Local weather forecast

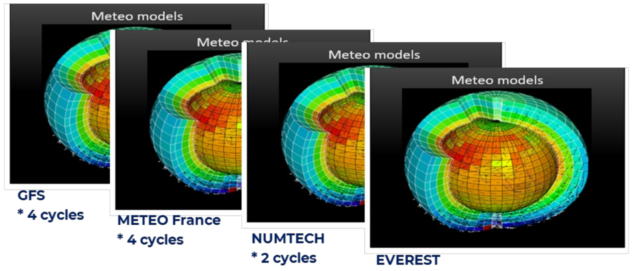


Air-quality forecast

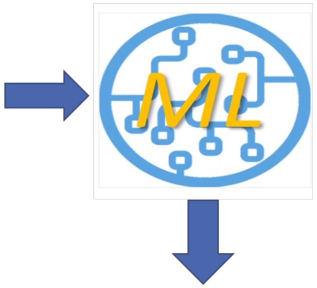
Improve speed to produce air-quality forecast and its quality

2. Ensemble prediction

N x deterministic weather forecast



Local weather observation on-site



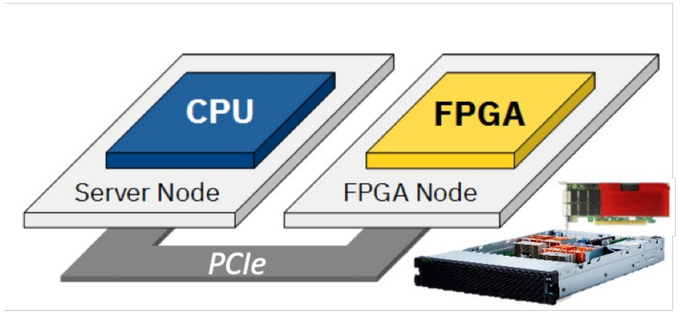
One aggregated weather forecast forced by observation

Improve quality of local weather forecast

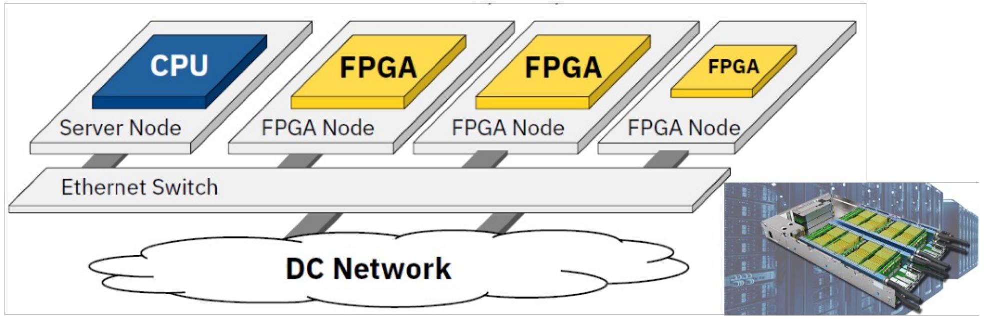
EVEREST Target system

- ❑ Network-attached and PCIe-attached FPGA nodes
  - ❑ Off-the-shelf FPGA devices
  - ❑ User logic designed and customized with HLS tools
- ❑ DC infrastructure and Supercomputers
  - ❑ workflow orchestration
  - ❑ reference implementation

FPGA as a Co-Processor

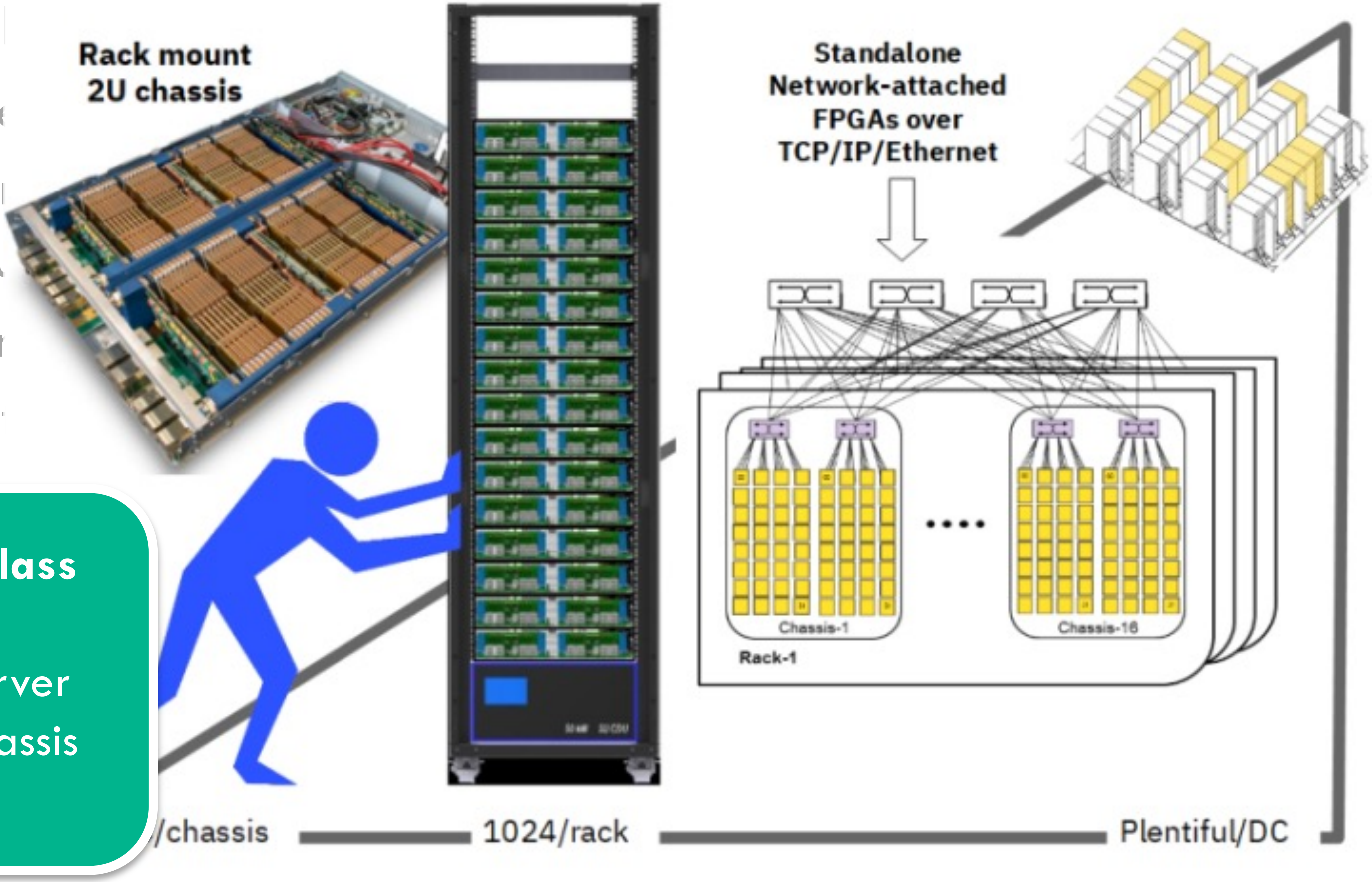


FPGA as a Peer-Processor



EVEREST Target system

- ❑ Network-attached and
  - ❑ Off-the-shelf FPGA de
  - ❑ User logic designed a
- ❑ DC infrastructure and Su
  - ❑ workflow orchestratio
  - ❑ reference implementa



Cloud-FPGA: FPGS as 1st-class citizens within a DC
(disaggregated from the server nodes, densely packed in chassis and racks)

EVEREST "SDK"

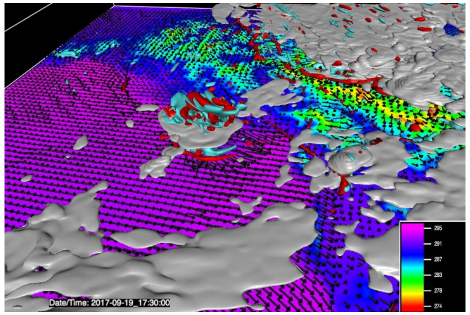
- SDK as in "**System** Development Kit"
  - Languages and high-level compilers
  - HLS tools and HW generation flows
  - Runtime adaptation and auto-tuning

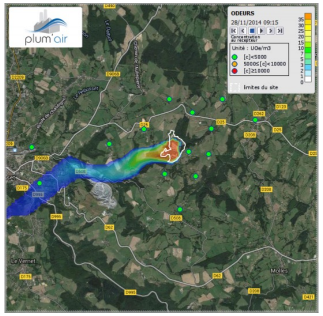
- Focus: Seamless acceleration of critical use case kernels and data-driven execution

Dataflow/task graphs (traffic routing)

HPC kernels (weather simulation)

ML (predictive models and decision making)

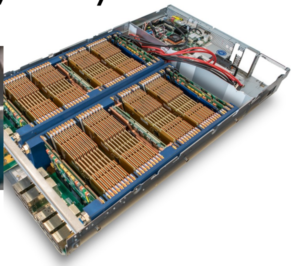




Heterogeneous HPC, BigData, ML systems

HBM-FPGA

HPC system



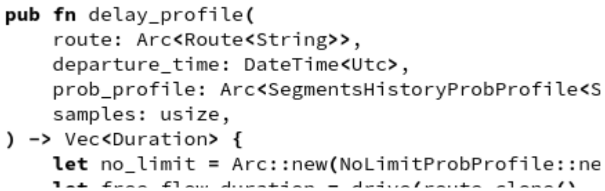
Cloud FPGA

C. Pilato, et al. "EVEREST: A design environment for extreme-scale big data analytics on heterogeneous platforms", DATE 2021

Entry points

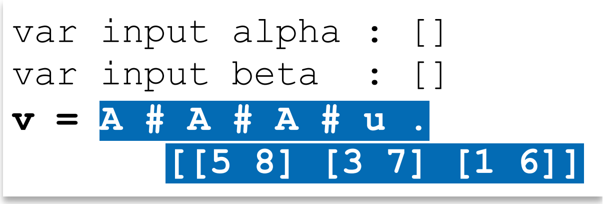
- ❑ Python-like and Rust-like syntax for implicit dataflow
  - ❑ Sound dataflow extraction and optimization
  - ❑ Guaranteed deterministic execution with shared state
- ❑ Domain-specific abstractions for numerics
  - ❑ Tensor, stencils, linear algebra
  - ❑ Algebraic transformations and polyhedral optimization
- ❑ Interoperable machine learning models (TVM, ONNX)
  - ❑ Leverage rich ML infrastructure
  - ❑ Work on top of libraries to optimize and distribute exec.

Dataflow/task graphs



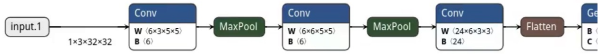
```
pub fn delay_profile(
    route: Arc<Route<String>>,
    departure_time: DateTime<Utc>,
    prob_profile: Arc<SegmentsHistoryProbProfile<S
    samples: usize,
) -> Vec<Duration> {
    let no_limit = Arc::new(NoLimitProbProfile::ne
    let free_flow_duration = drive(route.clone()
```

HPC kernels



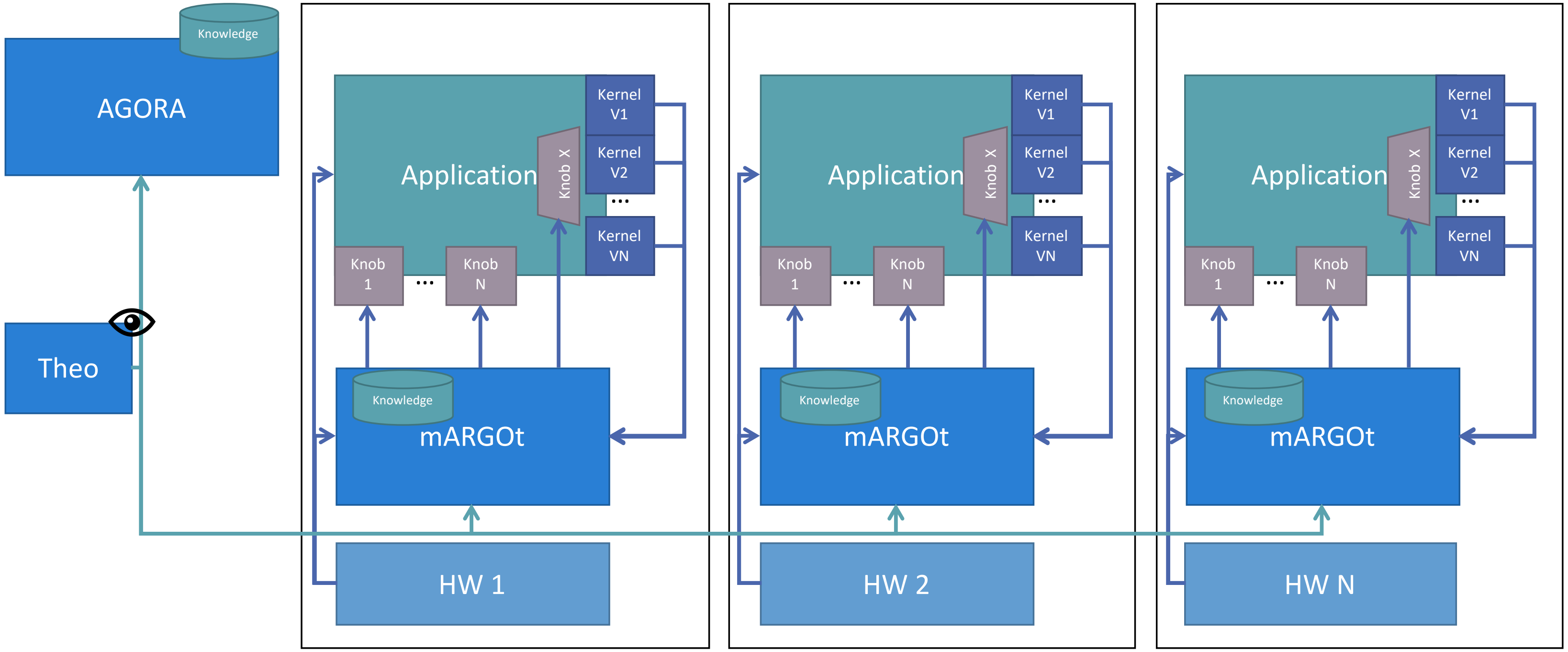
```
var input alpha : []
var input beta  : []
v = A # A # A # u .
      [[5 8] [3 7] [1 6]]
```

ML



EVEREST Runtime Environment

- ❑ FPGA systems with monitoring and infrastructure for dynamic autotuning
- ❑ Application variants (HW/SW) from compiler adapted to workload conditions



Ohua: Implicit dataflow

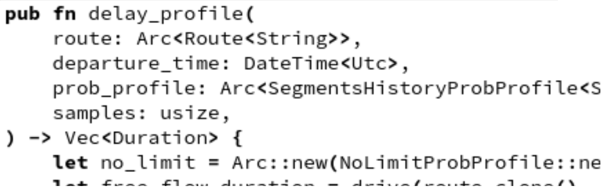
- ❑ Flexible compiler/language framework
  - ❑ Different entry languages (Rust--, Python--, Go--, ...)
  - ❑ Coordination lang.: Source-to-Source compiler

S. Ertel, et al. "STCLang: State Thread Composition as a Foundation for Monadic Dataflow Parallelism", Proceedings of the Symposium on Haskell, Aug 2019

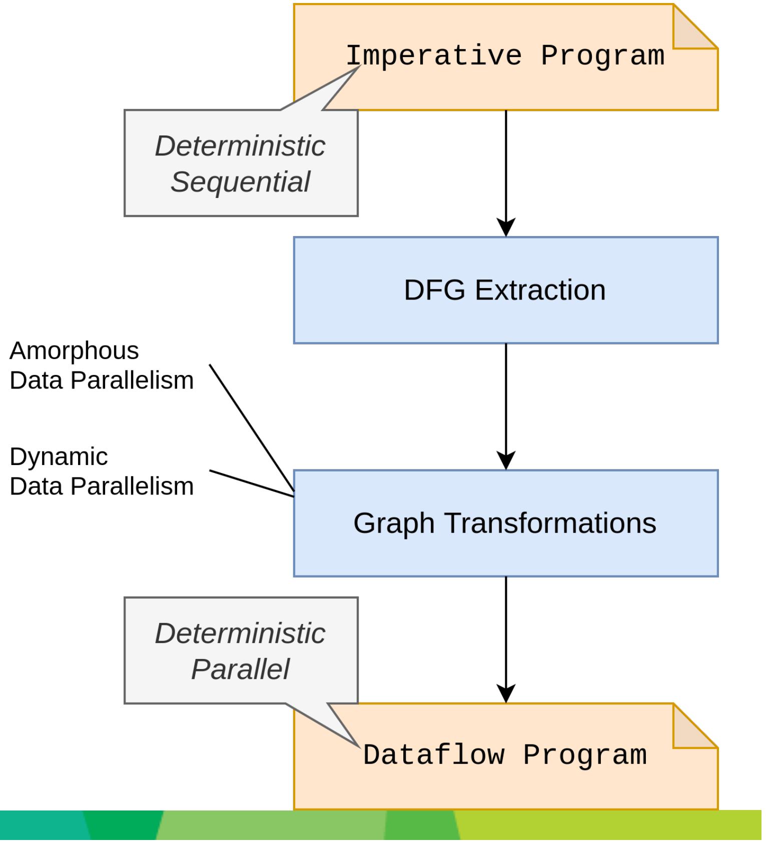
- ❑ Optimizations for Big-Data and micro-services

S. Ertel, et al. "Compiling for Concise Code and Efficient I/O", Proceedings of the Conference on Compiler Construction (CC 2018) , Feb 2018
S. Ertel, et al. "Supporting Fine-grained Dataflow Parallelism in Big Data Systems", Proceedings of PMAM, Feb 2018.

- ❑ Working on
  - ❑ Data parallelism (even for stateful functions)
  - ❑ Extended backend for FPGAs in EVEREST platform



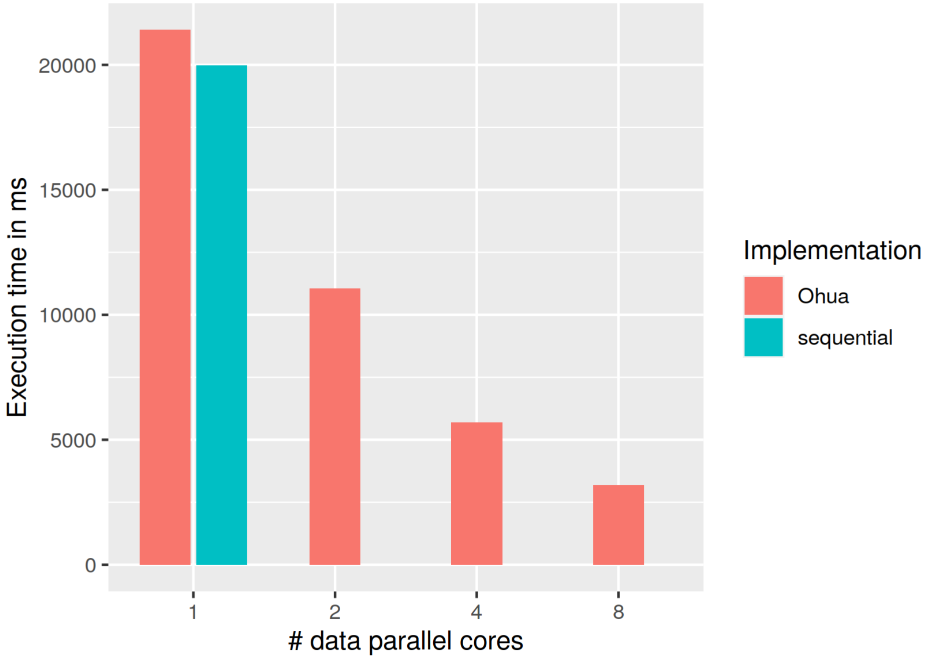
```
pub fn delay_profile(
    route: Arc<Route<String>>,
    departure_time: DateTime<Utc>,
    prob_profile: Arc<SegmentsHistoryProbProfile<S
    samples: usize,
) -> Vec<Duration> {
    let no_limit = Arc::new(NoLimitProbProfile::ne
    let free_flow_duration = drive(route.clone()
```

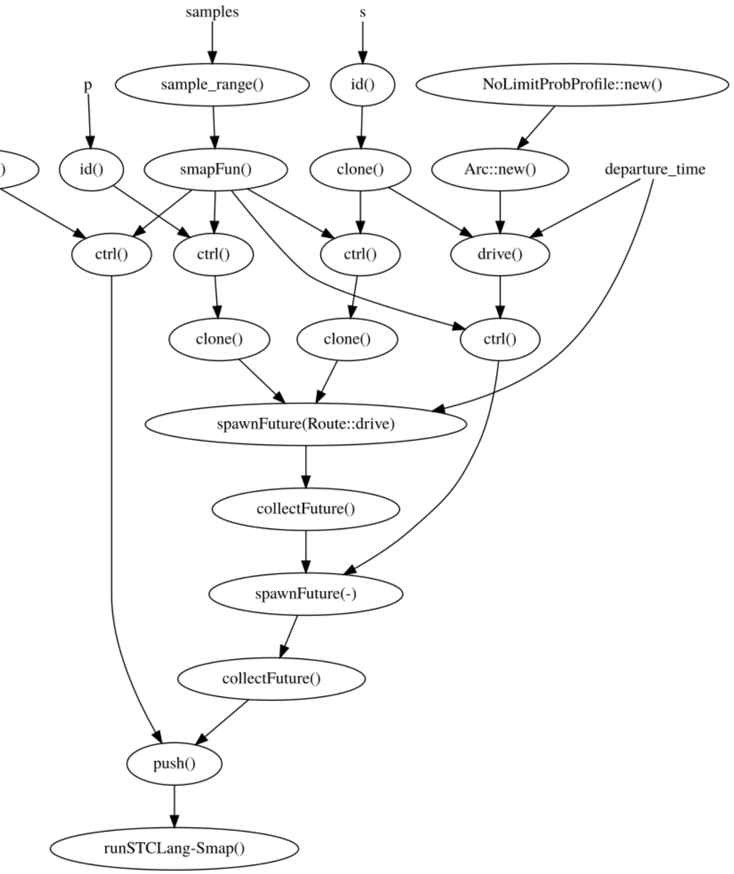


Ohua: Traffic routing algorithm

- Graph extraction and performance analysis

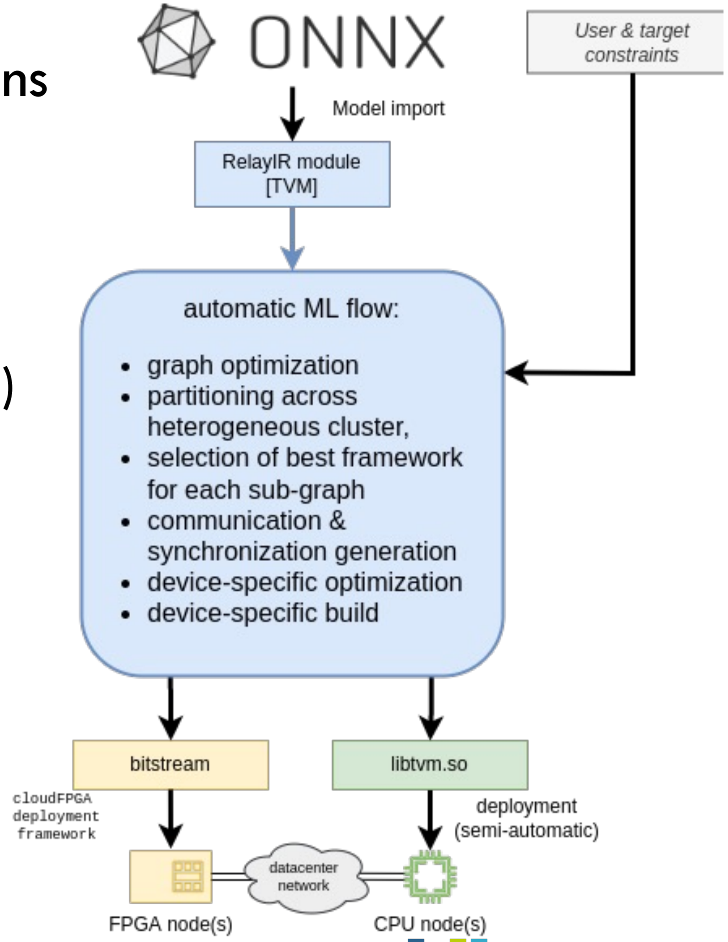
```
pub fn delay_profile(
    route: Arc<Route<String>>,
    departure_time: DateTime<Utc>,
    prob_profile: Arc<SegmentsHistoryProbProfile<String, Quartiles>>,
    samples: usize,
) -> Vec<Duration> {
    let no_limit = Arc::new(NoLimitProbProfile::new());
    let free_flow_duration =

    let mut res = Vec::new()
    for _ in helpers::sample_
        let dur = drive(rout
        let delta = dur - fr

        res.push(delta);
    }

    res
}
```





(Distributed) DNNs on FPGAs

- DNN of FPGA: Select among efficient implementations
  - haddoc2, FINN, hls4ml, VTA, VitisAI, …
  - Standardized way to include 3rd-party libraries
  - Automatic DSE of best available framework (depending on: operation, precision, target device, …)
- Currently based on TVM, MLIR integration WIP



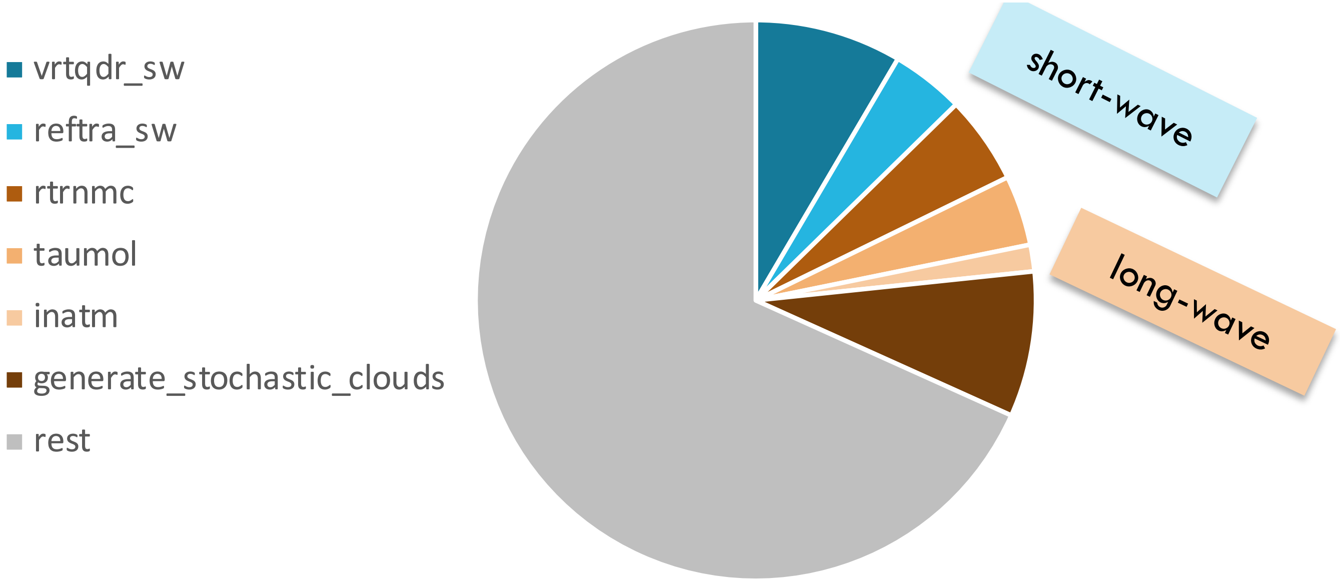
ONNX
User & target constraints
Model import
RelayIR module [TVM]
automatic ML flow:
graph optimization
partitioning across heterogeneous cluster,
selection of best framework for each sub-graph
communication & synchronization generation
device-specific optimization
device-specific build
bitstream
libtvm.so
cloudFPGA deployment framework
deployment (semi-automatic)
datacenter network
FPGA node(s)
CPU node(s)

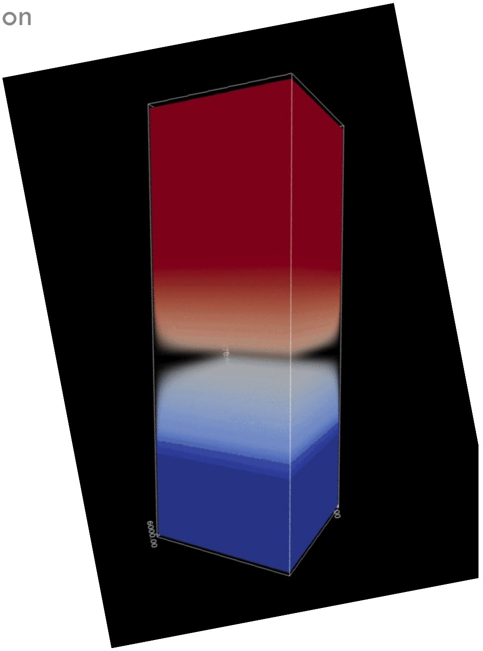
Weather modeling (WiP)

- ❑ Integration into WRF framework and complex build system
- ❑ Focus on radiation driver: Enable finer stepping
- ❑ Leveraging novel RRTMG-Parallel

https://everest-h2020.eu

Pincus, Robert, Eli J. Mlawer, and Jennifer S. Delamere. "Balancing accuracy, efficiency, and flexibility in radiation calculations for dynamical models." Journal of Advances in Modeling Earth Systems 11.10 (2019): 3074-3089.





© Prof. J. Castrillon. EVEREST Workshop @ HiPEAC 2022

Tensors expressions (numerics in CFD, ML)

- Expression-language for tensor operations and optimizations
  - Originally for spectral element methods in computational fluid dynamics

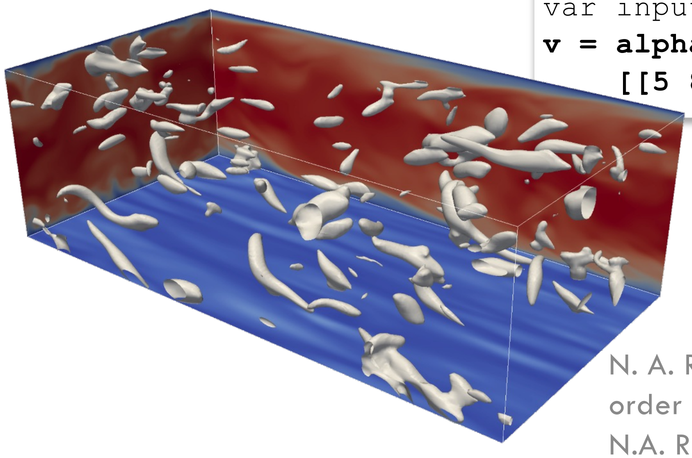
$$\mathbf{v}_e = (\mathbf{A} \otimes \mathbf{A} \otimes \mathbf{A})\, \mathbf{u}_e$$

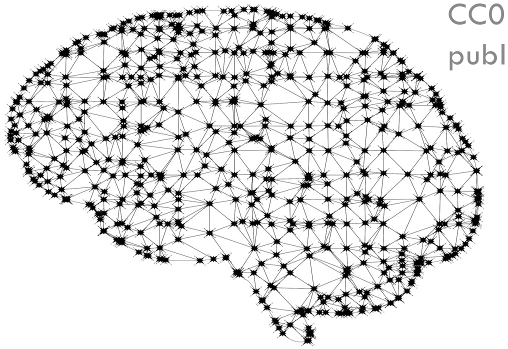
Interpolation kernel

```
source =  ...
var input A    : matrix            &
var input u    : tensorIN          &
var input output v   : tensorOUT &
var input alpha : []               &
var input beta  : []               &
v = alpha * (A # A # A # u .
    [[5 8] [3 7] [1 6]]) + beta * v
```

```
auto A = Matrix(m, n), B = Matrix(m, n),
     C = Matrix(m, n);
auto u = Tensor<3>(n, n, n);
auto v = (A*B*C)(u);
```

Fortran and C++ integration





CC0
public domain

N. A. Rink, et al. “CFDlang: High-level code generation for high-order methods in fluid dynamics”. RWDSL’18.

N.A. Rink, N. A. and J. Castrillon. “TeIL: a type-safe imperative Tensor Intermediate Language”, ARRAY’19, pp. 57-68

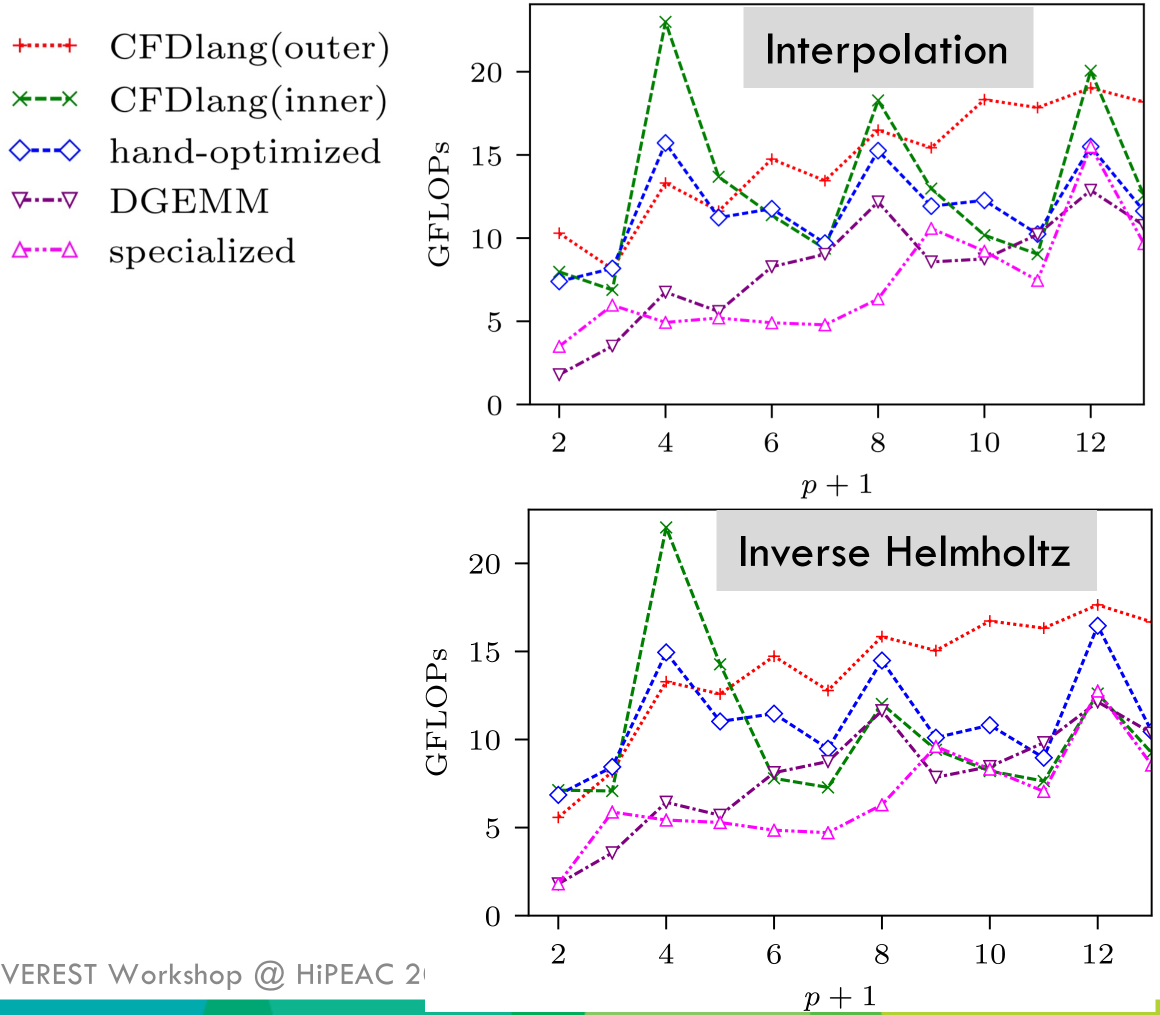
Closing the performance gap

- ❑ Not really optimization magic
  - ❑ Leverage expert knowledge
  - ❑ Algebraic identities

$$v_{ijk} = \sum_{l,m,n} (A_{kn} \cdot (A_{jm} \cdot (A_{il} \cdot u_{lmn}))$$

$$v_{ijk} = \sum_{l,m,n} (A_{kn} \cdot A_{jm}) \cdot (A_{il} \cdot u_{lmn})$$

$$v_{ijk} = \sum_{l,m,n} (A_{kn} \cdot ((A_{jm} \cdot A_{il}) \cdot u_{lmn}))$$



N. A. Rink, et al. "CFDlang: High-level code generation for high-order methods in fluid dynamics". RWDSL'18.
A. Susungi, et al., "Meta-programming for Cross-Domain Tensor Optimizations", GPCE'18 pp. 79-92.

Closing the performance gap

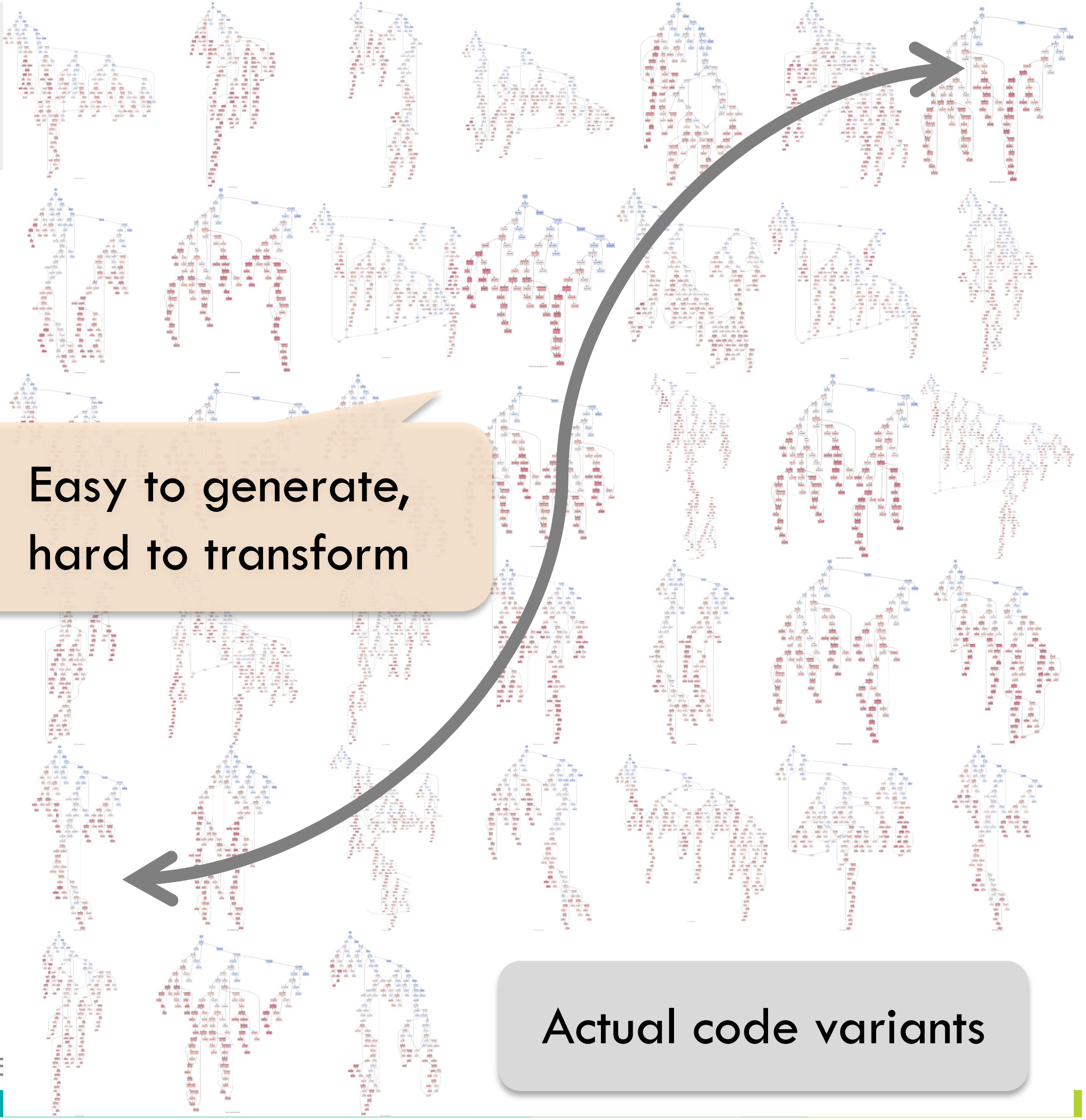
- ❑ Not really optimization magic
  - ❑ Leverage expert knowledge
  - ❑ Algebraic identities

$$v_{ijk} = \sum_{l,m,n} (A_{kn} \cdot (A_{jm} \cdot (A_{il} \cdot u_{lmn})))$$

$$v_{ijk} = \sum_{l,m,n} (A_{kn} \cdot A_{jm}) \cdot (A_{il} \cdot u_{lmn})$$

$$v_{ijk} = \sum_{l,m,n} (A_{kn} \cdot ((A_{jm} \cdot A_{il}) \cdot u_{lmn}))$$

N. A. Rink, et al. “CFDlang: High-level code generation for high-order methods in fluid dynamics”. RWDSL’18.
A. Susungi, et al., "Meta-programming for Cross-Domain Tensor Optimizations", GPCE’18 pp. 79-92.

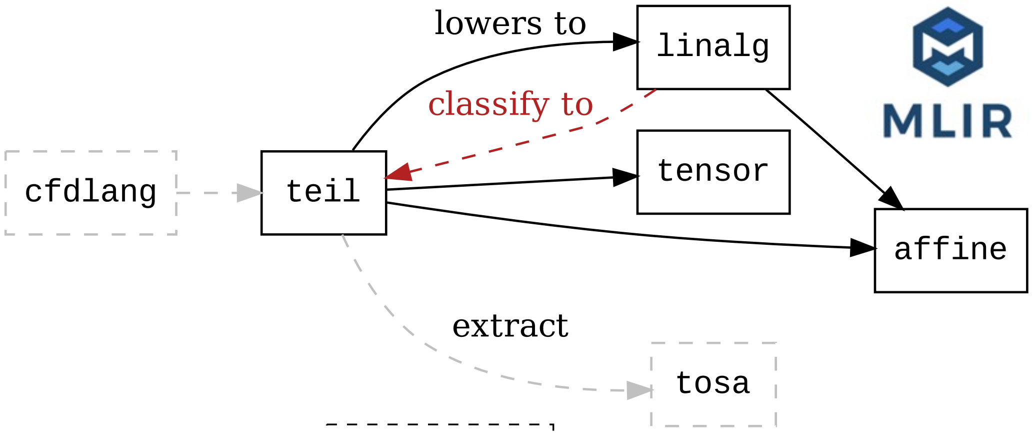


Easy to generate, hard to transform

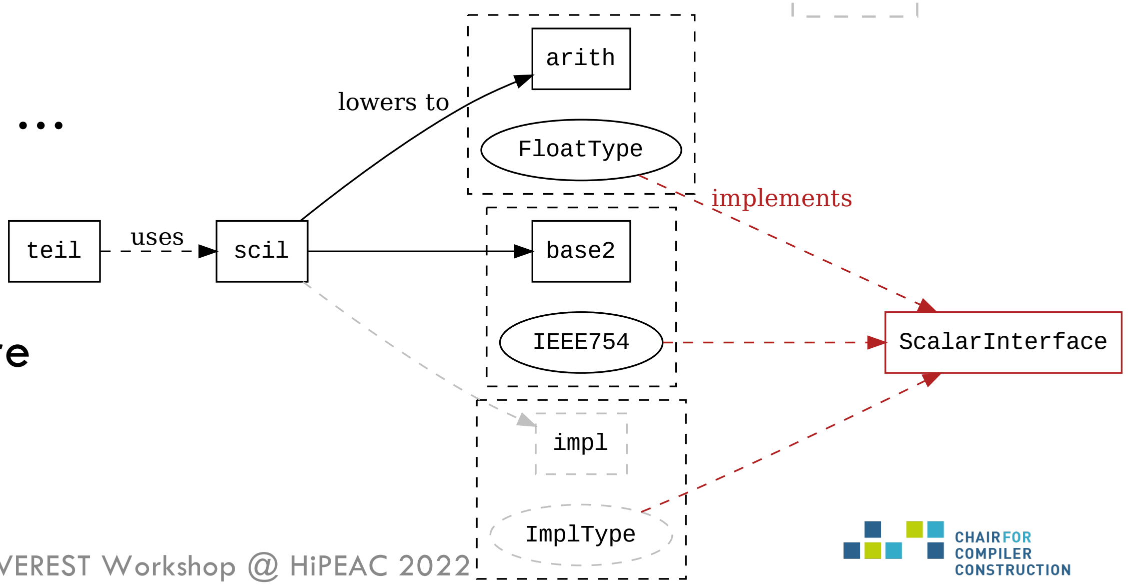
Actual code variants

TeIL in MLIR

- ❑ Primitive ops instead of index maps
- ❑ Easier to express identities (big-O trfs)
- ❑ Uses symbolic math, infinite precision



- ❑ Scalar types
  - ❑ ScIL provides scalar operators
  - ❑ ScIL provides Rationals, Neutrals, …
  - ❑ Base2 provides parametric binary number types
  - ❑ Based2 models (custom) hardware

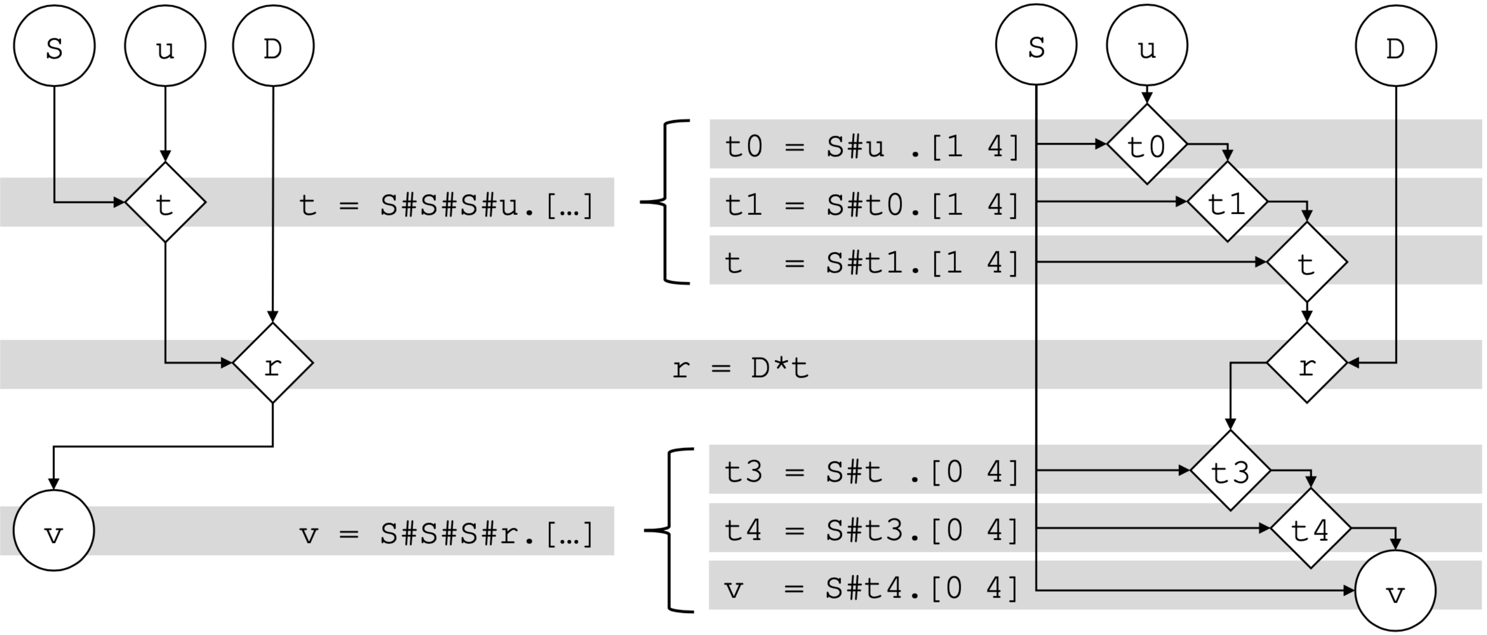


Domain-specific optimization

- ❑ Encode algebraic transformations (Interpolation as example)
- ❑ Direct feedback to expert via DSL export

$$t = (S \otimes S \otimes S \otimes u)_{axbycz}{}^{xyz}$$

$$t = \left(S \otimes \left(S \otimes (S \otimes u)_{cz}{}^{xyz}\right)_{by}{}^{cxy}\right)_{ax}{}^{bcx}$$



```
t  = S#S#S#u.[…]
r  = D*t
v  = S#S#S#r.[…]
```

```
t0 = S#u .[1 4]
t1 = S#t0.[1 4]
t  = S#t1.[1 4]
r = D*t
t3 = S#t .[0 4]
t4 = S#t3.[0 4]
v  = S#t4.[0 4]
```

FPGA code generation: Bus-attached FPGAs

- ❑ H2020 EU Project: Convergence HPC, Big Data and ML

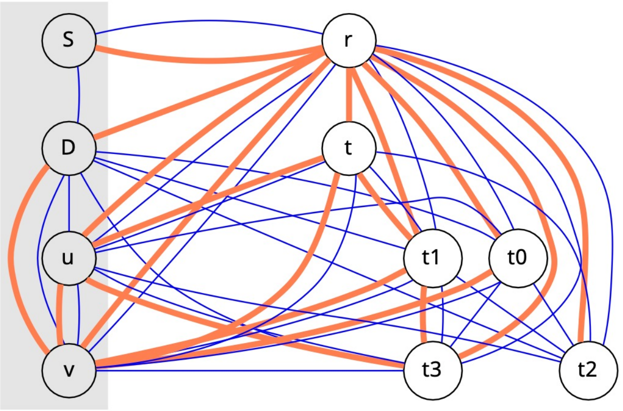
  C. Pilato, et al. "EVEREST: A design environment for extreme-scale big data analytics on heterogeneous platforms", DATE 2021

- ❑ Inverse Helmholtz Kernel

$$\mathbf{v}_e = (\mathrm{S} \otimes \mathrm{S} \otimes \mathrm{S})\mathrm{D}_e^{-1}(\mathrm{S}^T \otimes \mathrm{S}^T \otimes \mathrm{S}^T)\,\mathbf{u}_e$$

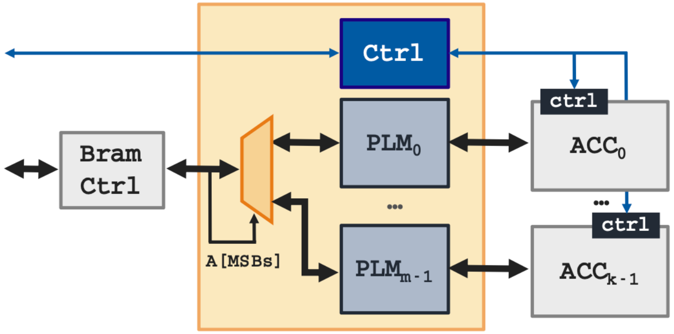
```
t = S # S # S # u . [[1
r = D * t
v = S # S # S # r . [[0
```

Lifetime analysis (polyhedral analysis)



Menosyne

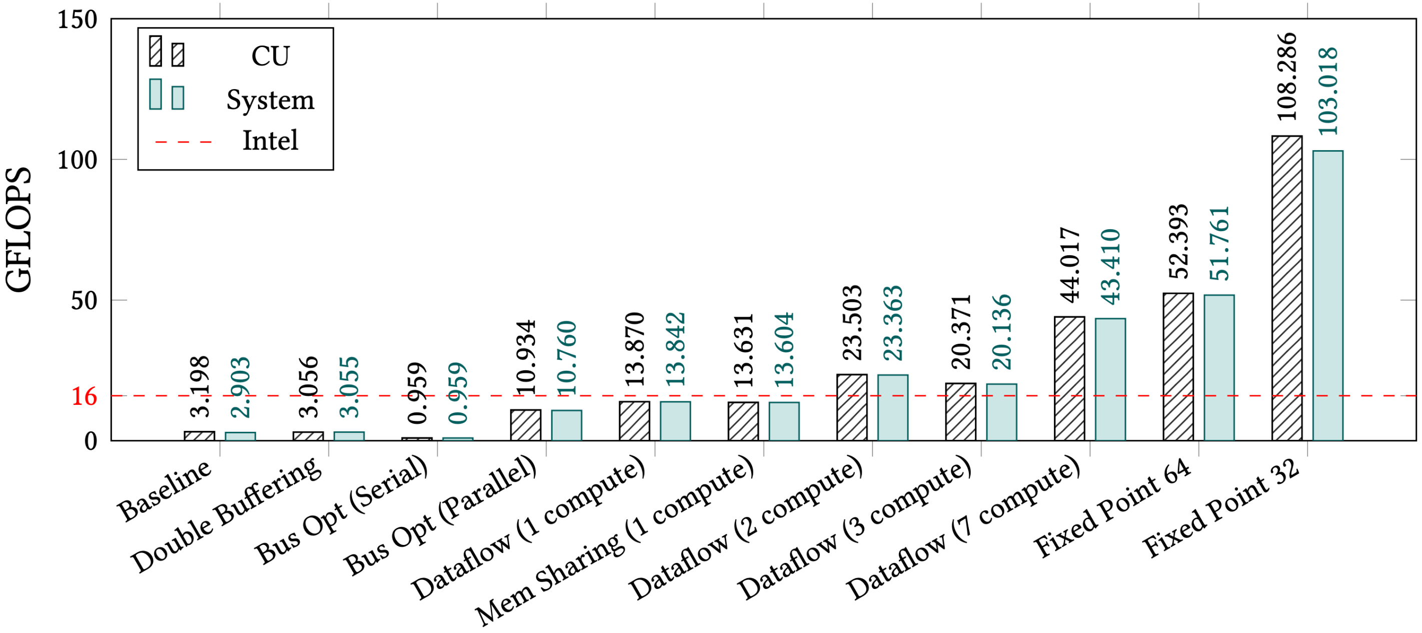
mem-subsystem gen (buffer sharing)



K. F. A. Friebel, S. Soldavini, G. Hempel, C. Pilato, J. Castrillon, "From Domain-Specific Languages to Memory-Optimized Accelerators for Fluid Dynamics", Proceedings of the FPGA for HPC Workshop, held in conjunction with IEEE Cluster 2021, Sep 2021

FPGA code generation: HBM FPGA

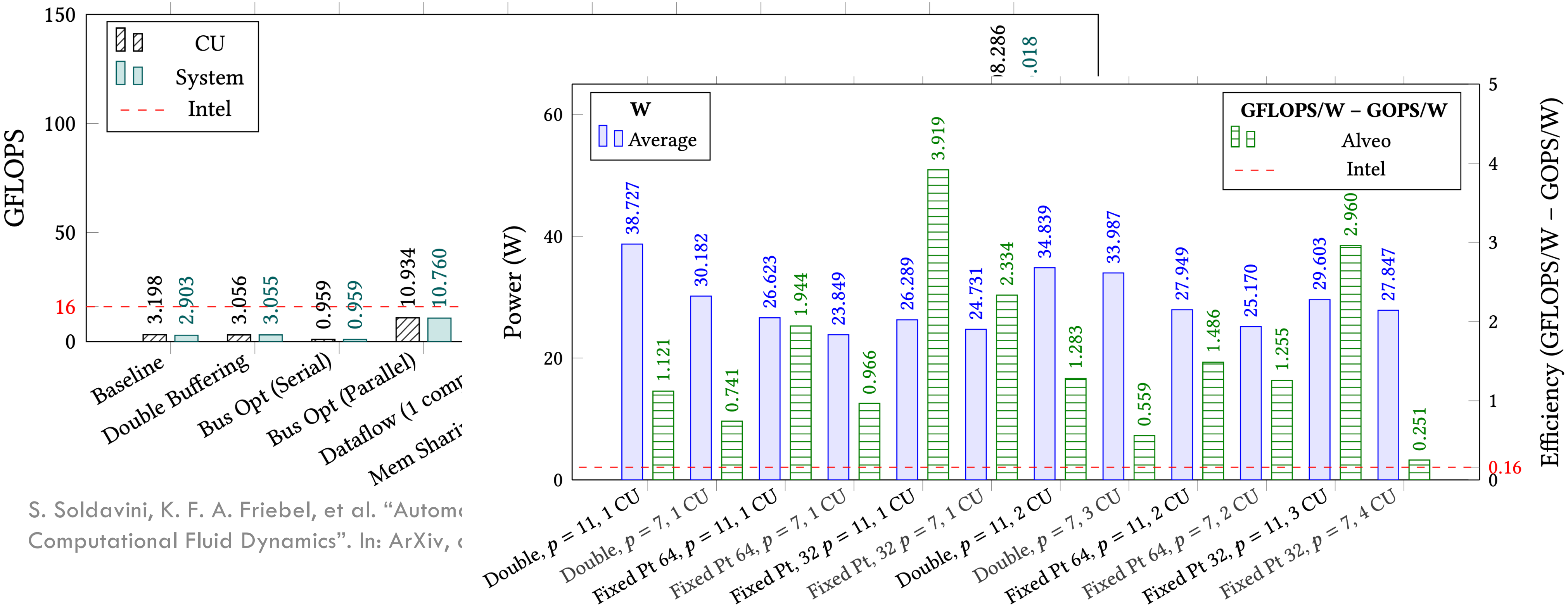
- H2020 EU Project: Convergence HPC, Big Data and ML
- HBM-FPGA and Cloud FPGA (ongoing)



S. Soldavini, K. F. A. Friebel, et al. “Automatic Creation of High-Bandwidth Memory Architectures from Domain-Specific Languages: The Case of Computational Fluid Dynamics”. In: ArXiv, arXiv:2203.10850 (Mar. 2022)

FPGA code generation: HBM FPGA

- H2020 EU Project: Convergence HPC, Big Data and ML
- HBM-FPGA and Cloud FPGA (ongoing)

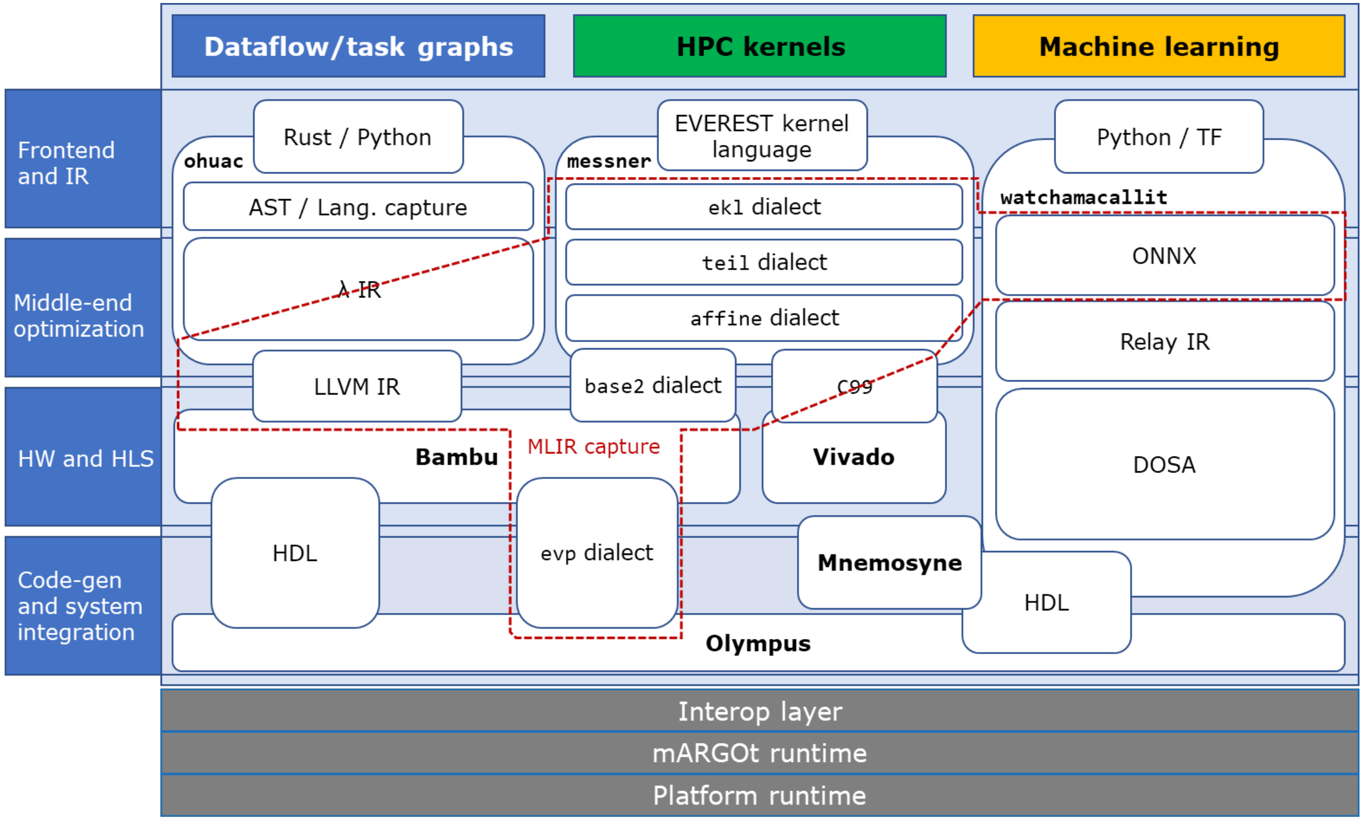


S. Soldavini, K. F. A. Friebel, et al. "Autom
Computational Fluid Dynamics". In: ArXiv,

Towards IR-level convergence

- ❑ Cross-domain analysis and optimizations
- ❑ Working on integration around **MLIR** (multi-level intermediate representation)

MLIR

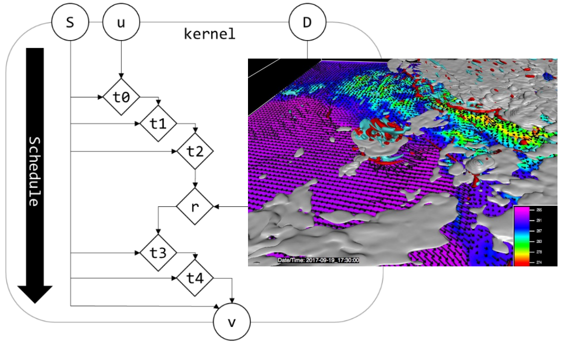


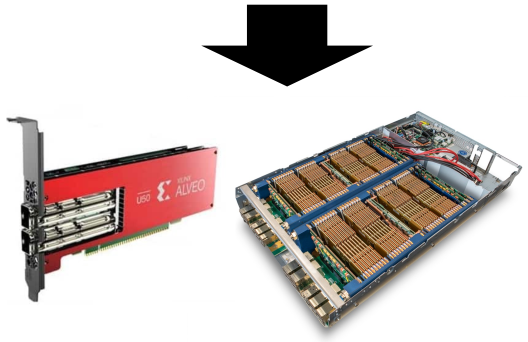
Summary

- dEsign enVironmEnt foR Extreme-Scale big data analyTics heterogeneous platforms
  - Use cases and target systems
  - SDK for programming, system generation and adaptivity
  - Provided examples of the current status of tool flows

- Further challenges
  - Modernization of code (e.g. WRF), better integration
  - More convergence at IR-level, need for community!
  - Optimization/DSE: More intelligence, cost modeling, …

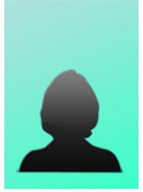
$$t = \left(S \otimes \left(S \otimes \left(S \otimes u\right)_{cz}^{xyz}\right)_{by}^{cxy}\right)_{ax}^{bcx}$$

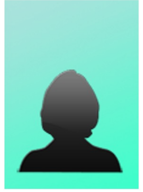


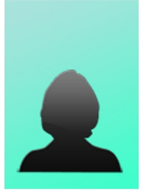


Thanks! & Acknowledgements

Hasna Bouraoui


João Cardoso


Hamid Farzaneh


Clément Fournier

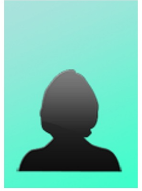
Karl Friebel

Dr. Asif Khan

Robert Khasanov

Alexander Brauckmann

Nesrine Khouzami


Dr. Steffen Köhler

Christian Menard

Julian Robledo

Lars Schütze

Felix Wittwer

Dr. Fazal Hameed

…, and previous members of the group (**Norman Rink**, Sven Karol, Sebastian Ertel, **Andres Goens**), and collaborators (**J. Fröhlich**, I. Sbalzarini, **A. Cohen**, **T. Grosser**, T. Hoefler, H. Härtig, **H. Corporaal**, **C. Pilato**, S. Parkin, P. Jääskeläinen, J-J. Chen, A. Jones)

EVEREST

https://everest-h2020.eu

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957269

STAATSMINISTERIUM
FÜR WISSENSCHAFT
KULTUR UND TOURISMUS

Freistaat
SACHSEN