CHAIR FOR COMPILER CONSTRUCTION

cfaed CENTER FOR ADVANCING ELECTRONICS DRESDEN

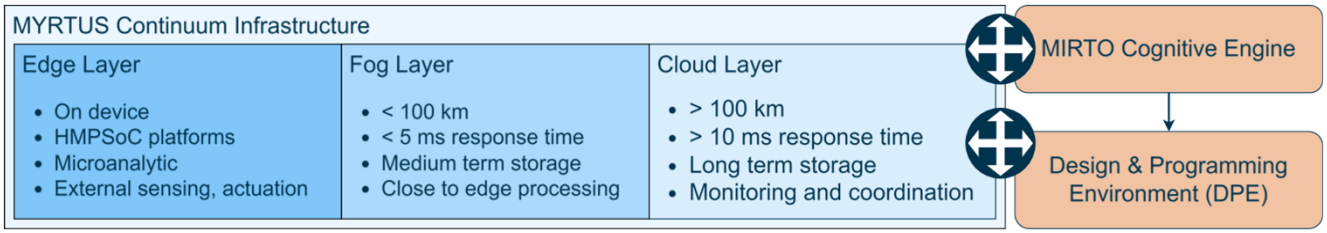
Leveraging the MLIR infrastructure for the computing continuum

Contact: jiahong.bi, guilherme.dos_santos_korol, jeronimo.castrillon@tu-dresden.de

MYRTUS Computing Continuum Infrastructure

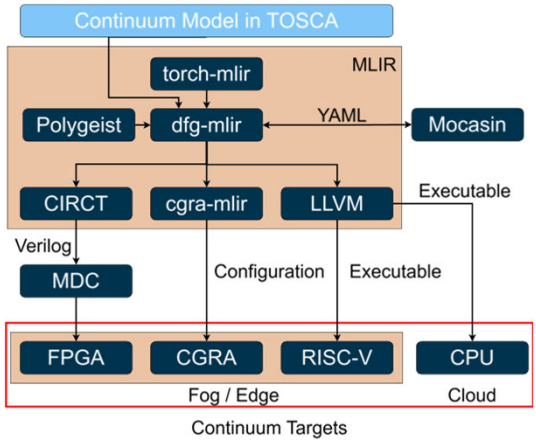
- ❑ Cross-layer modelling
- ❑ Multi-layer simulation / analysis
- ❑ Heterogeneous computing nodes
- ❑ Modularity, composability, security

- ❑ MYRTUS[1] is built as a layered cloud-fog-edge continuum
- ❑ MIRTO engine manages resources, connections and workloads across the continuum
- ❑ Design & Programming Environment (DPE) supports definition, implementation and deployment



Node-Level Optimization and Deployment (NLOP)

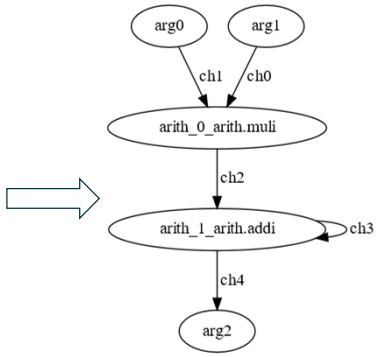
- ❑ TOSCA[2] files will be leveraged for orchestration in the continuum
- ❑ NLOP supports different inputs, such as C and PyTorch
- ❑ Transformations to be implemented for Dataflow Graph generation
- ❑ Mocasin Design Space Exploration (DSE) to find mappings for CGRA
- ❑ RISC-V processors will manage FPGA/CGRA acceleration



Dataflow Graph Dialect

- ❑ Multiple backends for different hardware architectures
  - ❑ A general FPGA backend[3]
  - ❑ Cloud FPGA backend[4]
  - ❑ General CPU backend based on OpenMP
- ❑ Ability to describe graphs using Operator/Process
- ❑ Support for iteration arguments for specific tasks
- ❑ Generate the inner dataflow inside an operator

```
dfg.operator @mac
        inputs(%in0: i32, %in1: i32)
        outputs(%out: i32)
        iter_args(%sum: i32)
    initialize
    {
        %0 = arith.constant 0 : i32
        dfg.yield %0 : i32
    }
{
    %0 = arith.muli %in0, %in1 : i32
    %1 = arith.addi %0, %sum : i32
    dfg.output %1 : i32
    dfg.yield %1 : i32
}
```



Future Plans and Possibilities

Middle-end and Backend

- ❑ Integration with Mocasin[5] for DSE
- ❑ Integration with MDC[6] tools
- ❑ New MLIR dialects for CGRA[7], etc…

dfg-mlir atop CIRCT[8]

- ❑ Similar semantics to existing dialects
- ❑ Introduce multiple pull/push for channels
- ❑ Customizable FIFO channel implementation

Time and Adaptivity

- ❑ Reactor model in Lingua Franca[9]
- ❑ Hybrid mapping technology and adaptive execution on hardware[10]

1. Palumbo, Francesca, et al. "MYRTUS: Multi-layer 360 dYnamic orchestration and interopeRable design environmenT for compute-continUum Systems." Proceedings of the 21st ACM International Conference on Computing Frontiers: Workshops and Special Sessions. 2024.
2. OASIS TOSCA Standard. https://docs.oasis-open.org/tosca/TOSCA-Simple-Profile-YAML/v1.3/os/TOSCA-Simple-Profile-YAML-v1.3-os.html
3. Bi, Jiahong. "A Lowering for High-Level Data Flows to Reconfigurable Hardware." (2024).
4. Soldavini, Stephanie, et al. "Etna: MLIR-Based System-Level Design and Optimization for Transparent Application Execution on CPU-FPGA Nodes." 2024 IEEE 32nd Annual International Symposium on Field-Programmable Custom Computing Machines (FCCM). IEEE, 2024.
5. Menard, Christian, et al. "Mocasin—rapid prototyping of rapid prototyping tools: A framework for exploring new approaches in mapping software to heterogeneous multi-cores." Proceedings of the 2021 Drone Systems Engineering and Rapid Simulation and Performance Evaluation: Methods and Tools Proceedings. 2021. 66-73.
6. Manca, Federico, Francesco Ratto, and Francesca Palumbo. "ONNX-to-Hardware Design Flow for Adaptive Neural-Network Inference on FPGAs." arXiv preprint arXiv:2406.09078 (2024).
7. Vazquez, Daniel, et al. "STRELA: STReaming ELAstic CGRA Accelerator for Embedded Systems." arXiv preprint arXiv:2404.12503 (2024).
8. CIRCT Project. https://circt.llvm.org/
9. Lohstroh, Marten, et al. "Toward a lingua franca for deterministic concurrent systems." ACM Transactions on Embedded Computing Systems (TECS) 20.4 (2021): 1-27.
10. Smejkal, Till, et al. "E-Mapper: Energy-Efficient Resource Allocation for Traditional Operating Systems on Heterogeneous Processors." arXiv preprint arXiv:2406.18980 (2024).