Optimizing Java on the EU processor design

AERO

An open source cloud software ecosystem for the EPI hardware

Christos Kotselidis, Technical Coordinator
Associate Professor, The University of Manchester
Chief Engineer, KTM Innovation
Brno, 16/06/2023

Funded by the European Union

Motivation: EU Chips Act

European Commission

English Search

Shaping Europe's digital future

Home | Policies | Activities | News | Library | Funding | Calendar | Consultations

Home > Policies > European Chips Act

European Chips Act

The European Chips Act will boost Europe's technological sovereignty, competitiveness, resilience and contribute to the digital and green transitions.

Why do we need a European Chips Act?

Chips — also known as semiconductors — are the building block of all electronic products. They play a central role in our modern economies and our daily lives. Chips underpin the digital transformation and are essential to all industries, such as the car industry, communications, data processing, space, defence, smart devices and gaming, to name a few.

The recent global chips shortage has disrupted supply chains, caused product shortages ranging from cars to medical devices, and in some cases even forced factories to close.

The European Chips Act, having reached a political agreement on 18 April 2023, seeks to strengthen the semiconductor ecosystem. It is composed of a Communication, which spells out the European Strategy and rationale behind the Chips Act, a proposal for a Regulation, and a Recommendation to Member States.

Communication, Regulation, Recommendation and Joint Undertaking >

Staff Working document >

https://digital-strategy.ec.europa.eu/en/policies/european-chips-act

Funded by the European Union

AERO

Accelerated EuRopean clOud

Vision

Enable the future heterogeneous EU cloud infrastructure

AERO will **upbring** and **optimise** software for the heterogeneous cloud ecosystem on the European processor:

- **compilers**, **runtime systems**
- **operating systems**, **system software**
- auxiliary **software deployment services**.

Accelerate the adoption of the EU cloud ecosystem

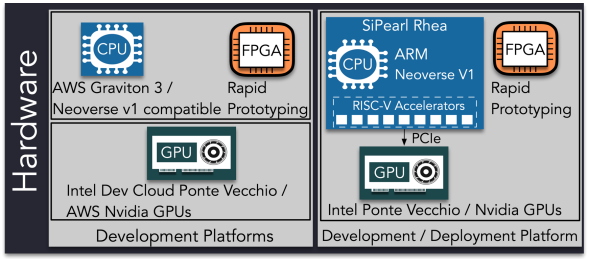
- Accelerate the adoption of the EU cloud ecosystem
- Open source ecosystem
- Communication and dissemination of results to industry, academia, and standardization bodies.

Funded by the European Union

AERO

HW Platforms

- Numerous processor designs derived by the European Processor Initiative [1] and other projects
- AERO focuses on SIPEARL's Rhea processor
  - ARM Neoverse V1 processors
  - PCIe support for GPUs
  - RISC-V accelerators
- RISC-V platforms
- FPGA boards for rapid prototyping
- GPUs and other accelerators



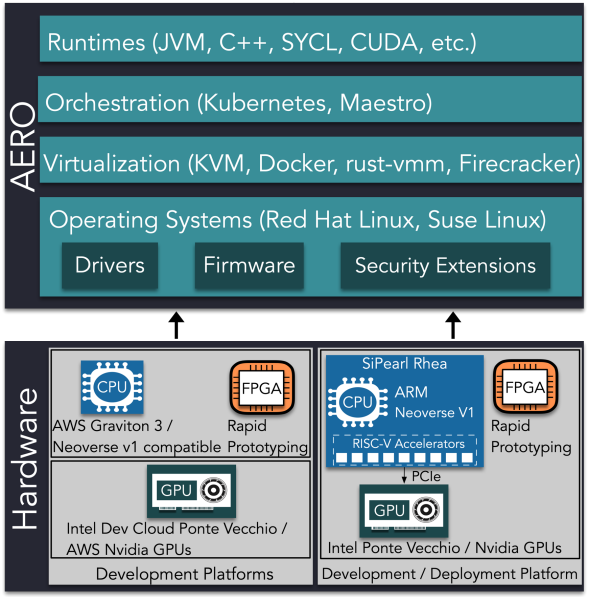
Funded by the European Union

AERO

System Software | Execution Runtimes

Optimized execution of programming languages & runtime systems primed for **cloud:**

- ***OpenJDK, GraalVM*** → *managed programming languages (Java, Python, Scala, R, etc.)*
- ***TornadoVM*** → *GPU HW acceleration of managed programming languages*
- ***Quarkus*** → *Cloud Microservices*
- SYCL & DPC++/OneAPI → HW acceleration of non-managed applications running in C/C++



Funded by the European Union

Java on the EU Processor

- Focus of The University of Manchester and Red Hat
- Optimize production JVMs on the heterogeneous EU platforms
  - AArch64, RISC-V, GPU/FPGA Acceleration
- OpenJDK, Mandrel, GraalVM, **Quarkus** and **TornadoVM**
- Aim for *stability*, *compatibility* and *performance*

TORNADO VM

QUARKUS

Funded by the European Union

TornadoVM

A JVM plugin that accelerates Java methods on heterogeneous hardware!

Features:

- Lightweight API
- Platform agnostic
- Automatic code specialization

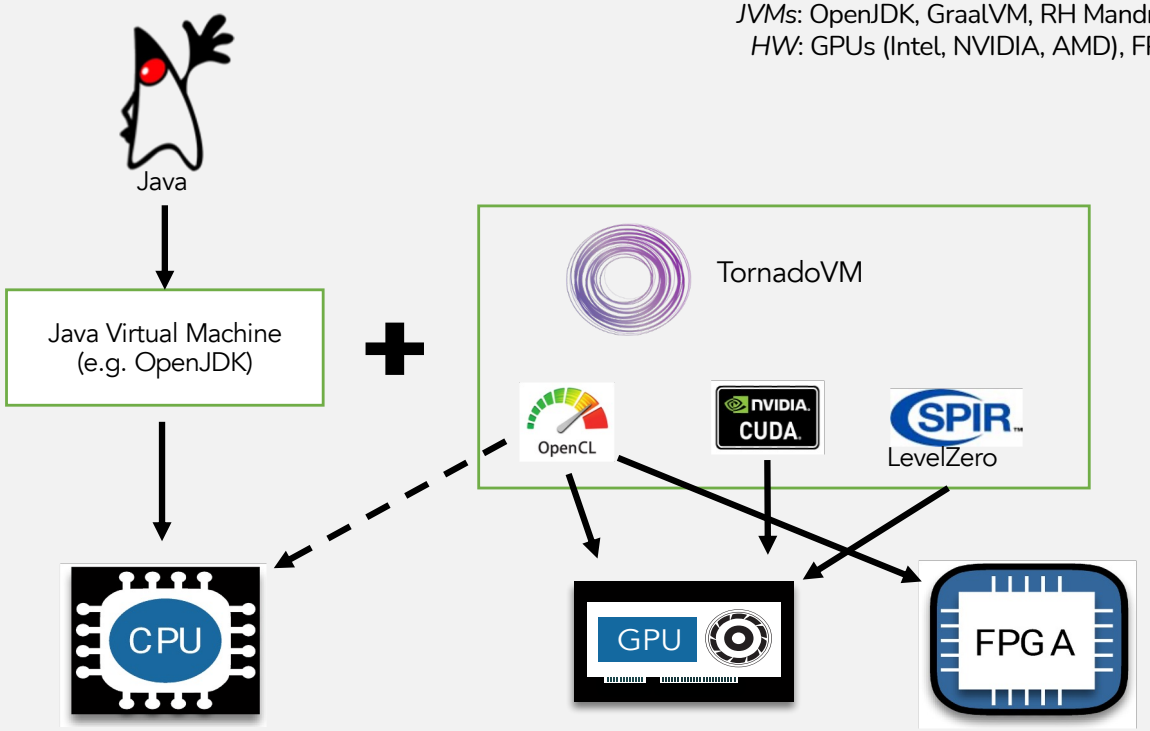

https://tornadovm.org


@tornadovm

The University of Manchester

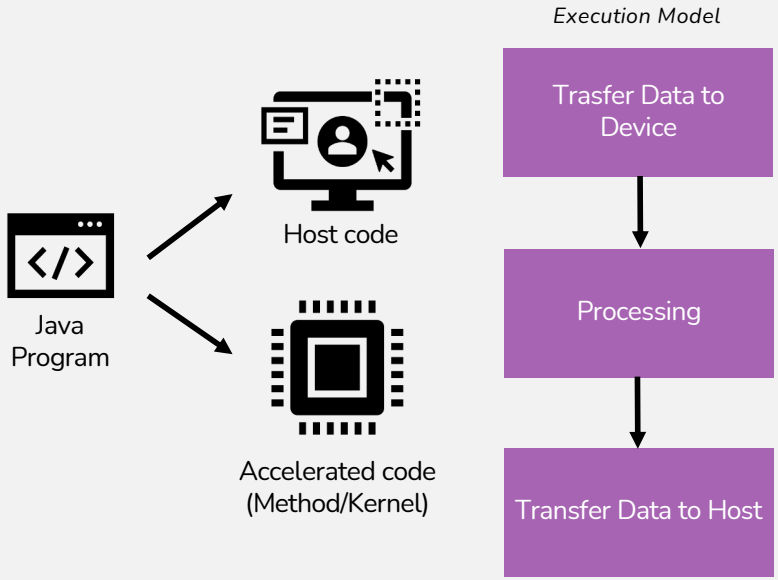
TornadoVM

JVMs: OpenJDK, GraalVM, RH Mandrel, AWS Corretto, MS JDK
HW: GPUs (Intel, NVIDIA, AMD), FPGAs (Intel, Xilinx), CPUs (x86, Aarch64)



Funded by the European Union

TornadoVM | Execution Model

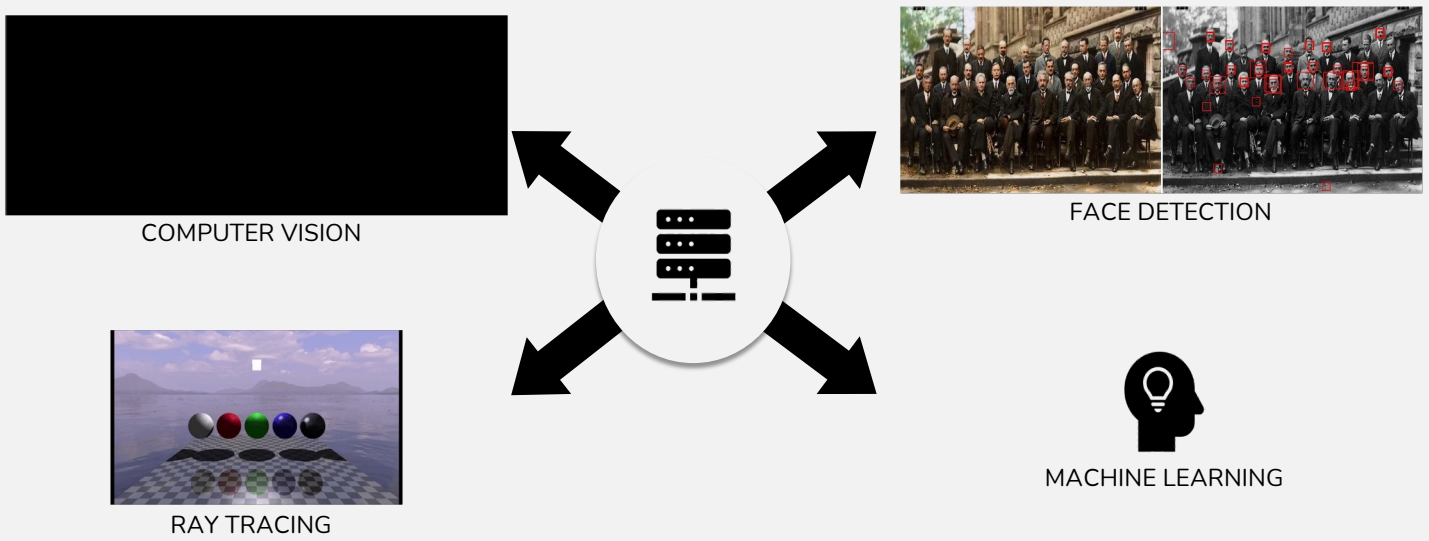



```
TaskGraph taskGraph = new TaskGraph("tg")
        .transferToDevice(DataTransferMode.FIRST_EXECUTION, input)
        .task("methodA", Class::MethodA, input, output, ...)
        .transferToHost(DataTransferMode.EVERY_EXECUTION, output);
```

A) Loop Parallel API for Java programmers | Easy to use, lightweight

B) Kernel API for programmers with OpenCL/CUDA expertise, or not (e.g. porting existing kernels) | Advanced features and capabilities

TornadoVM | Use Cases



https://www.tornadovm.org/use-cases

Thank you!

Contact us:

aero-project.eu

@AERO_Project_EU

github.com/AERO-Project-EU

zenodo.org/communities/aero

Funded by the European Union. Views and opinions are however those of the author(s) only and do not necessarily reflect those of the European Union or the HaDEA. Neither the European Union nor the granting authority can be held responsible for them. Project number: 101092850.

AERO has also received funding from UKRI under grants no. 10048318 and 10048915, and the Swiss State Secretariat for Education, Research, and Innovation.