

Emerging Heterogeneous Processing for Future HEP Experiments

US Higgs Factory Planning

Collaborators: Z. Demiragli (Boston University), D. Sperka (Boston University), A. Paramonov (Argonne National Laboratory), Rainer Bartoldus (SLAC), Nhan Tran (Fermi National Accelerator Laboratory), Michael Begel (Brookhaven National Laboratory), Antonio Boveia (The Ohio State University)

Project Abstract

Goal: Evaluate modern and emerging computing architectures for data processing systems.

Context:

- Higgs Factory experiments will benefit from triggerless readout, requiring more off-detector data processing to filter experimental data before storage.
- Modern platforms rely heavily on accelerators (e.g., IPU, GPU, FPGA, ASIC) for efficiency and speed.

Key Areas of Exploration:

- Industry driven proprietary accelerators (e.g., Google TPU).
- Experiment-developed solutions (LHC experience and beyond).
- Emerging accelerators and computation platforms.

Towards development of community-endorsed benchmarks for reproducible and standardized evaluation

Usage of heterogeneous processing aligns with DOE's emphasis on energy-efficient and scalable technologies! 2

Why Heterogeneous Architectures?

Commercial and high-performance computing increasingly leverage heterogeneous accelerator architectures for improved efficiency compared to CPUs.

- Examples include GPUs, Google's Tensor Processing Unit (TPU), FPGAs, and various ASICs (e.g., compute in memory, neuromorphic, AI/ML accelerators).

Key Characteristics that are needed:

- Simplicity: Low-effort portability to various computing platforms.
- Diversity: Capture a wide range of workloads (e.g., processing, sorting, memory access).

Evaluation Metrics:

- Efficiency: Power consumption for specific workloads.
- Ease of Use: Effort required for software development and support.

Proposed Work Plan and Funding Needs

Request:

- 1 postdoc and 1 student across 3 years (shared between collaborators).
- Access to various heterogeneous accelerator architectures.

Year 1: Industry and HEP survey to compile viable computation platforms for the next decade.

Year 2: Evaluate programming models and ease of use for identified platforms.

Year 3: Formalize and run HEP-relevant benchmarks to assess performance and efficiency.

Output: Release a community report to share the findings