Project Proposal: Extending the Alpaka performance portability library with CUDA Cooperative Groups for the CMS pixel reconstruction

Description

This project aims to enhance the Alpaka API[1] to support launching Alpaka kernels utilizing CUDA Cooperative Groups[2]. This will involve a detailed examination of Alpaka's current accelerator and kernel launch mechanisms to identify integration points for cooperative group functionalities. The project encompasses setting up a development environment, designing API extensions, implementing these extensions, and conducting basic performance comparisons using simple benchmarks. The result of the project will be a prototype of Alpaka with support for CUDA Cooperative Groups. Analogous mechanisms for other accelerator types beyond CUDA will not be implemented within the scope of this project. However, such existing APIs will be explored, and the Alpaka code will be written in a way that will provide a clear path to their future implementation. Should the final prototype be feature-ready, it will be tested within the CMS Pixeltrack[3] project.

Deliverables

- **API Design Document**: A github issue in Alpaka repository—proposal for extending Alpaka's API to include support for CUDA Cooperative Groups.
- **Standalone mock-ups**: Conceptual prototypes to test API extension ideas.
- **Enhanced Alpaka API**: A prototype version of Alpaka with integrated support for CUDA Cooperative Groups.
- **Pixeltrack integration**: A version of Pixeltrack with support for CUDA Cooperative Groups (optional, depending on the final state of the prototype)
- **Report**: Document covering the implementation details, initial performance observations, and usage examples. Based on results, further decisions on design changes and change of directions (if needed)

[1] https://github.com/alpaka-group/alpaka
[2] https://developer.nvidia.com/blog/cooperative-groups
[3] https://github.com/cms-patatrack/pixeltrack-standalone

Timeline

Month 1: Learning and Initial Design

- Gain a thorough understanding of CUDA Cooperative Groups and the current architecture of the Alpaka library.
- Set up a development environment with CUDA Toolkit and Alpaka.
- Get acquainted with the Alpaka group's software development process workflows.
- Develop initial API extension proposals and create conceptual mock-ups for integrating CUDA Cooperative Groups.

Month 2: API Development and Prototyping

- Iterate on the conceptual mock-ups, and refine the API proposal.
- Clarify and visualize changes in Alpaka data structures and design needed to implement the API proposal.
- Begin coding the new features into the Alpaka API, focusing on enabling the use of cooperative groups in user-defined kernels.
- Implement basic functionality and iterate based on simple test cases.

Month 3: Application Integration, and Performance Testing

- Refine the API based on feedback from preliminary tests using example kernels.
- Integrate with Pixeltrack to conduct performance comparisons.
- Compile a report detailing the development process, API and implementation changes, and initial performance results.