PhD Thesis Proposal

Title: **Next-generation interactive analysis frameworks for the HL-LHC: combining user experience with scalable in-memory computing**

Background & Motivation
The HL-LHC is expected to deliver an order of magnitude more data than its predecessor, which will push the boundaries of both data storage and analysis infrastructure. As traditional ROOT-based workflows reach limits of scalability and usability, the HEP community is increasingly exploring in-memory column analyses, interactive notebooks and heterogeneous computing resources (HPCs, clouds, GPUs).

Despite promising prototypes (e.g. Coffea, Dask-based workflows and JupyterHub integrations), there is a gap between these tools and their broad, sustained adoption at all user levels and across institutions. This project seeks to close this gap by developing and evaluating next-generation tools for large-scale analyses.

Objectives

- **Seamless analyses scaling**: Development of an infrastructure for the transition from notebook-based local prototyping to large-scale distributed computing environments (e.g. Dask-on-HPC, Spark-on-cloud)
- **Intelligent storage integration**: Rethinking data storage models by combining conventional file-based access (ROOT) with object storage (e.g. Ceph, S3) and on-demand data conversion pipelines
- **Notebook-native interfaces**: Create intuitive Jupyter-based interfaces that integrate:
    1. Resource-aware scheduling
    2. Automatic scaling of clusters
    3. Version-controlled workflows with reproducibility in mind
- **Dynamic data reduction and intelligent caching**: Exploration of intelligent caching mechanisms and on-the-fly reduction layers (inspired by query engines such as DuckDB or Polars) for interactive exploration of large amounts of data
- **Federated analytics and ML readiness**: Prototype workflows that support federated data access and real-time ML model integration to enable low-latency inference on large, remote datasets

Methodology & Work Plan

Year 1: Foundations & User-Centric Design

- Collaboration with analysis teams (e.g. Belle II, CMS) to map typical workflows
- Benchmarking of existing tools (Coffea, AnalysisX, Dask, RNTuple) in realistic scenarios
- Prototyping of a Jupyter-based interface that integrates Dask/Spark with token-based authorisation and live monitoring

Year 2: Smart Storage & Scalable Execution

- Develop a hybrid storage model that combines XRootD and object stores
- Implement strategies for lazy loading and partial deserialization using Parquet, Awkward

Array, or RNTuple
- Integrate with real analysis pipelines (e.g., nanoAOD) and validate with physics-driven benchmarks

Year 3: Training, Community Adoption & ML

- Creation of training materials, tutorials and reproducible sample analyses
- Collaboration with the IRIS-HEP and Analysis Grand Challenge groups
- Expansion of the framework to include support for ML inference in real time (ONNX, Triton Inference Server)

Expected Contributions

- A reproducible, user-friendly analysis framework tailored to HL-LHC datasets
- A prototype hybrid memory analysis pipeline with columnar, on-demand computation
- Recommendations and best practices for the use of HPC/cloud in HEP analysis
- Contributions to open source HEP analysis software (e.g. PRs on Coffea, Awkward, RNTuple)

Expected Impact

- Lower entry barrier for newcomers thanks to interactive, notebook-like interfaces
- Improved reproducibility and transferability of analyses in different computing environments
- Improved resource efficiency through more intelligent data access and caching strategies
- Closer integration with modern data science ecosystems, making HEP analysis more attractive and collaborative