

# Refactoring AwkwardForth Generation in Uproot

## IRIS-HEP Fellows Program Project Proposal

Seth Bendigo

June 2023

## 1 Introduction

Uproot is an I/O library for reading and writing ROOT files for use in Python [1]. When reading in non-columnar data types, iteration is required, and Python loops are slow compared to compiled languages. A compiled language cannot be used directly since the code needed to deserialize a ROOT file with complex data structures is not known at compile time. Just-in-time compilation could be used, but to maintain portability, Uproot instead implements the use of AwkwardForth, an internal domain specific language [2]. By generating AwkwardForth code to read in the incoming complex data types, Uproot is significantly optimized. In the case of `std::vector<std::vector<float>>`, AwkwardForth is faster than Python by a factor of about 400.

## 2 Project Description

The current implementation of AwkwardForth generation poses some problems that makes it difficult to maintain. One of the more notable problems is excessive mutability, which makes it particularly difficult to debug. For instance, there is Python code that generates Python code which further generates AwkwardForth code, and this generation changes as data is read. Another problem is the readability of the code-base. There are many attributes which do not have descriptive names, and there is a lot of unused code that needs to be cleaned up.

The main goal of this project is to refactor AwkwardForth generation, utilizing test-driven development, to avoid these issues. One of the first steps taken will be to locate and remove the unused code. Then, the code will be restructured to align more with functional programming. In doing this, mutability will be reigned in, and as a result, the code will be easier to read, debug, and maintain. Once the refactor is complete, adequate documentation will also be provided. Furthermore, a CONTRIBUTING.md file for the entirety of Uproot will be written as Uproot does not currently have one.

The project will be completed under the supervision of Jim Pivarski and Ioana Ifrim.

### 3 Timeline

- June 5 - June 17: Familiarize myself with the Uproot code-base and how AwkwardForth code is currently generated.
- June 18 - July 8: Remove unused code, and begin work on refactoring AwkwardForth generation to align more with functional programming, utilizing current tests.
- July 9 - July 29: Project placed on a temporary hold for conferences and moving.
- July 30 - Aug 19: Finish up work on refactoring AwkwardForth generation, along with completing further testing, and writing documentation.
- Aug 20 - Sep 2: Finish writing documentation, and create any presentation materials.

### References

- [1] Jim Pivarski, Henry Schreiner, Angus Hollands, Pratyush Das, Kush Kothari, Aryan Roy, Jerry Ling, Nicholas Smith, Chris Burr, and Giordon Stark. Uproot, September 2017.
- [2] Jim Pivarski, Ianna Osborne, Pratyush Das, David Lange, and Peter Elmer. AwkwardForth: accelerating uproot with an internal DSL. *EPJ Web of Conferences*, 251:03002, 2021.