



**UF** UNIVERSITY of  
FLORIDA

# Introduction and Overview

**Sergei Gleyzer, Lorenzo Moneta**

**TMVA Developers Meeting**

**September 15, 2016**



- **Status and Overview**
- **Release plans**
- **Latest developments**

**Today's meeting aims to:**

- **Bring together TMVA developers**
  - Past, present and future
  - Hopefully make it a regular occurrence
  - More open for anyone interested to attend or contribute

## **Today's topic: latest developments in TMVA**

- A lot of recent development (2015-2016)**
- Design improvements, new features and functionality**
- Significant involvement of S.G, L.M., Ph.D. students, GSoC, other contributors, IML**



**One year ago: Future of TMVA document drafted**

**[Future of TMVA 16\\_09\\_2015.pdf](#)**

**In the beginning of IML**

- **HEP Community inter-experimental review and consensus to what was missing and needed to improve**
  - **Addresses important points about design, performance and desired features**

# Excerpt 1



## The future of TMVA (Draft version 16/09/2015)

### Introduction:

- **TMVA is the ROOT-integrated package for Machine Learning (ML)**
- **Commonly used in many published HEP analyses**
- **In production software of the two major LHC experiments (ATLAS/CMS)**
- **Provides the first point of contact for people in HEP trying to use ML**
- **Has basic neural networks, boosted decision trees, etc**
- **Provides a common interface and associated support - very useful to HEP**
- **Written about 10 years ago, and ML has evolved significantly in that time**
  - **Originally written to introduce ML techniques to the HEP community**
  - **It has now fulfilled that purpose - time to move to the next stage**

### TMVA use cases:

- **Standard analysis users (exotics, SUSY, etc)**
  - **Happy with standard BDT, etc**
  - **Would benefit from integration of modern algorithms and full TMVA support**
- **Precision analysis or performance users (for example: b-tagging)**
  - **Want the best performance possible**
  - **Willing to invest significant time into cutting edge algorithms**
  - **Already willing to work with ML algorithms which are not yet in TMVA**
  - **Would benefit from having a generic TMVA interface to make the process more straightforward that includes a consistent toolkit for evaluation/monitoring**
- **Advanced users and other interested parties**
  - **Are willing to probe the limits of what is possible in TMVA**
  - **Are potential future TMVA developers or at least contributors**

# Excerpt 2



## Core requirements:

- **The core TMVA package should provide a set of competitive and simple algorithms for standard HEP analysis usage**
  - **XGBoost is a promising C++ package for integration as the core BDT algorithm**
  - **Other core algorithms should also be updated**
- **TMVA interfaces for R and python (with support libraries) for high-performance use**
  - **Allows usage of modern ML packages for performance users**
- **Provide full and straightforward separation of training, testing and application**
  - **Packages which are not simple to integrate with TMVA can then be trained externally and the results can be applied through TMVA**

## Modernising TMVA:

- **Flexibility**
  - **The code should be made more modular, such that adding interfaces is straightforward**
    - **Significant progress has already been made by the RTMVA group**
  - **The core should be more flexible, allowing for decoupling for datasets/methods/variables in contrast to the current approach**
    - **The new re-design by the RTMVA group addresses this issue**
- **Computational Performance**
  - **The core code should be redesigned for improved computational performance**
    - **The C++ standard and programming techniques have been substantially updated in the past decade**
    - **Change matrix algebra to something like eigen which links to kernel level packages (ATLAS, BLAS)**

# Excerpt 3



## Desired Features:

- **Cross-validation**
  - **Standard in ML**
  - **New redesign by the RTMVA team allows easy implementation due to feature/method/dataset decoupling**
- **Additional information for Analyzer:**
  - **Variable importance, accurate feature ranking**
    - **FAST algorithm for feature importance currently being integrated by the RTMVA team with the new redesign.**
- **Parallelization**
  - **Many places where it applies, the RTMVA team is currently working on a general prototype**
  - **Thread safety for multi-threading (important for production)**
- **GPU support for the most computationally intensive algorithms**
- **Alternative input file types to more easily work with ML community (example: HDF5)**
- **A high-statistics sample for testing purposes: the current sample within TMVA is not adequate for studying modern algorithm performance**
- **Ability to output a standalone c++ code/executable without additional dependencies**
- **Expert users should be able to pause and resume training after tweaking hyperparameters as is done in the ML community**
- **Make it easier for the ML community to contribute directly to TMVA such as through a GitHub repository which is open to pull requests**

**Most  
addressed  
this  
year**

## How a TMVA redesign affects the three use cases:

- **All users: improved computational performance and dataset flexibility**





## Stay true to what made TMVA successful

- **By HEP and for HEP**
- **Method neutrality**
- **Competitive performance**
- **Ease of use**
  - Introduction of new users
- **Continuously improve**
  - Address modern challenges with latest ideas

# New Features



**Modular Design and Interfaces**

**Deep Learning (CPU + GPU)**

**Parallelism**

- **Multi-threading**
- **Multi-processing**
- **Spark**
- **GPUs**

**Cross-Validation**

**HyperParameter Tuning**

**Interactive Training**

**Unsupervised Learning**

**Greater integration with Jupyter**

**2015**

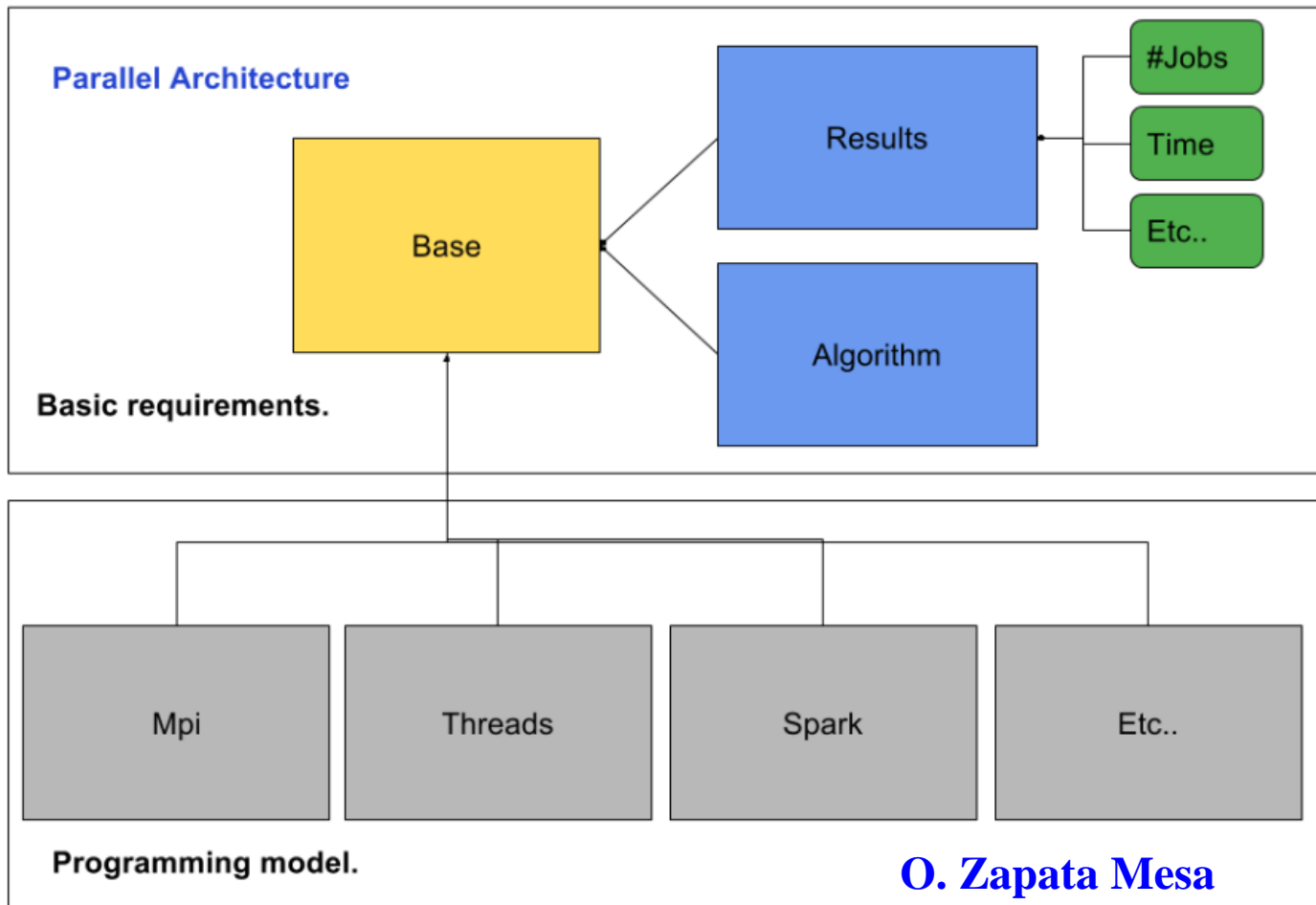
**2016**



## Next Release:

- **With next ROOT release at the end of September**
  - **Will include some of the features you will see today**
  - **Busy with integration**
  - **Some of the features will follow shortly**

- **Many recent developments in TMVA**
  - **Addressing community-driven review and desired features**
  - **People across experiments joining and contributing to this effort (IML)**
    - **Especially students**
      - **You will hear about some of their contributions today**





- Open source **web-based** application blends **code** with elements such as text, figures, links

- Excellent integration of instructions and executable code

- Perfect for interactive analysis and teaching demonstrations of anything that involves code

- Can be run locally, on a server, laptop or smartphone

