

CHEP 2015 Poster Session - Okinawa Japan

Dual-use tools and systematics-aware analysis workflows in the ATLAS Run-II analysis model

Dual-use tool requirements

The **Athena Framework** is used for all official production in ATLAS, including reconstruction, simulation, etc.

Most (~75%) ATLAS analysis code is written in *alternate* frameworks or in a basic *ROOT* environment.

Dual-use tools need to support the appropriate components depending on the environment:

	Athena	ROOT
Build system	CMT	RootCore
Event store	Storegate	TEvent, TStore
Tool Retrieval	ToolSvc	Simple ToolStore
Messaging	Gaudi messaging	Simple MsgStream

Combined performance tools

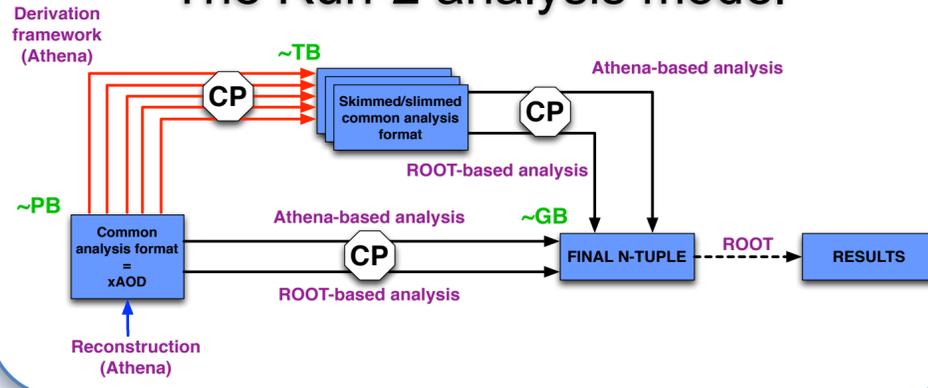
Tools that implement the recommendations and calibrations of the combined performance groups of ATLAS, including

- Object corrections
 - Energy calibrations
 - Resolution smearing
- Object selections
 - Cleaning bad objects
 - Identification criteria
- Object weights
 - Reconstruction/identification efficiencies

Anatomy of a dual-use tool

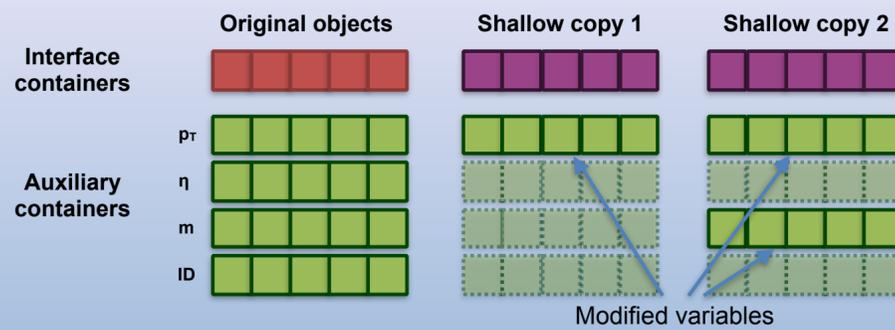
- Interface class inheriting from `IAsgTool`
 - Supports the component design pattern
- Implementation class inheriting from `AsgTool`
 - Provides event store access through common interface
 - Provides messaging functionality
- Compiler directives switch underlying framework machinery

The Run-2 analysis model

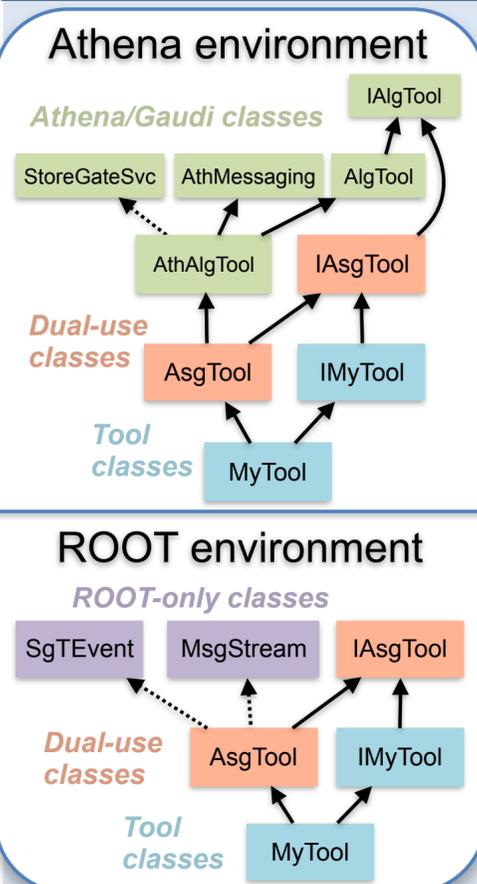


Shallow copies in the new EDM

Shallow copies are a powerful new EDM feature for analysis workflows. When an object container is copied, auxiliary variables are only duplicated when modified. Otherwise, they are taken from the original container.



Dual-use tool class structure



Systematics: tool interface

Systematics-aware tools implement the following methods

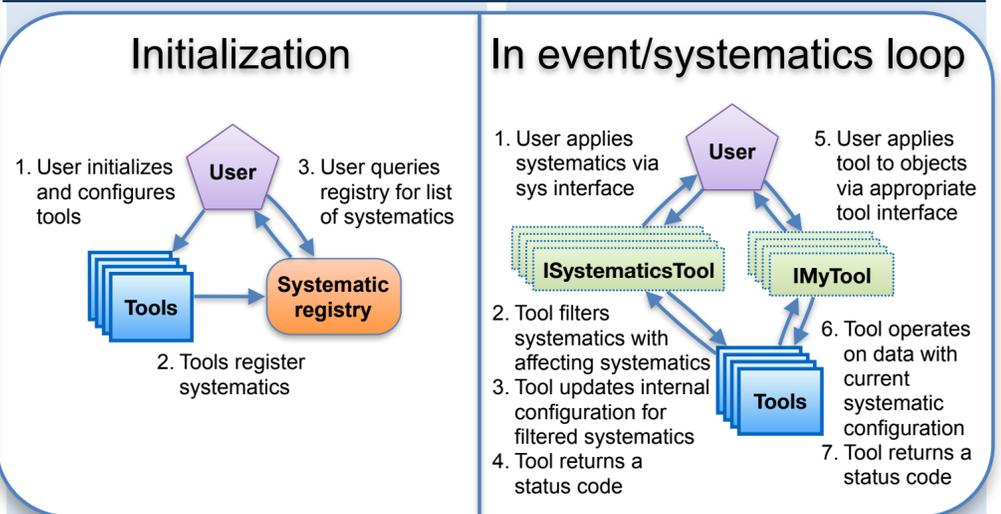
```
// Am I affected by this systematic?
bool isAffectedBySystematic(sys)

// Which systematics affect me?
SystematicSet affectingSystematics()

// Which systematics do I recommend to apply?
SystematicSet recommendedSystematics()

// Update configuration for systematic set
SysCode applySystematicVariation(sysSet)
```

Systematics: usage in analysis application



Components of the new analysis model

- xAOD event data model**
 - Readable in Athena and in ROOT
 - Simple interface objects and containers
 - Variable payloads stored in a dynamic auxiliary store
- Derivation framework**
 - Centralized reduction of input data for analyzers
 - Applies fixes, corrections, and preliminary selections to go from PB to TB
- Dual-use combined performance (CP) tools**
 - Tools for the user to apply all recommended prescriptions of the combined performance groups

Systematics: representation

A **SystematicVariation** object describes a single nuisance parameter and a variation

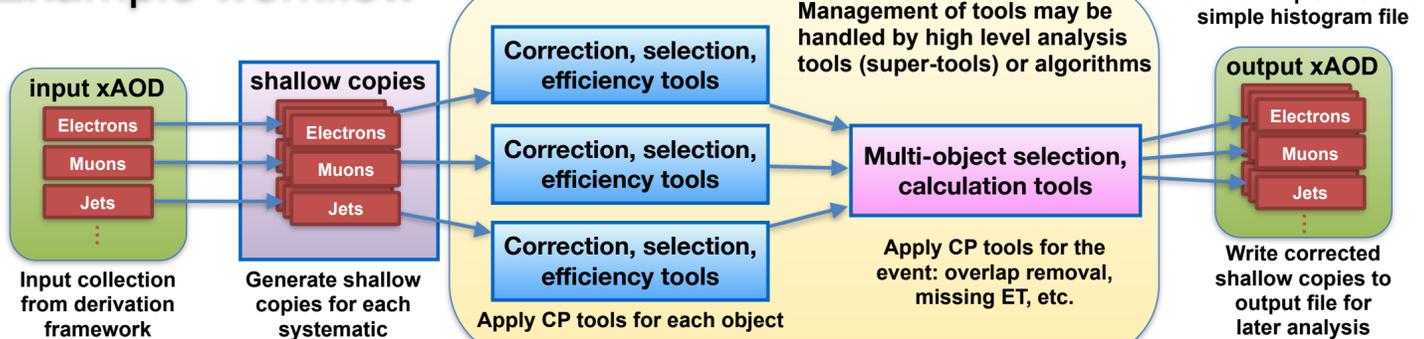
- e.g., jet energy scale at +1 sigma

A **SystematicSet** object describes a collection of systematic variations defining a point in nuisance parameter space

- allows to evaluate correlations
- caches hash value for efficient lookups in hash tables

A **SystematicRegistry** object holds information about all possible and recommended systematics in a user's application

Example workflow



Outlook and future work

The dual-use tool design has transformed the way analysis code is written

- Harmonized interfaces and behaviors make life easier for the user
- Dual-use concept gives users flexibility in choice of analysis framework
- Underlying tool machinery eases burden on tool developers to support multiple frameworks

- Possible future developments
- Improved handling of meta-data in ROOT
 - Additional dual-use framework components
 - e.g. algorithms, tool service