Recent advances in ADL, CutLang and adl2tnm

G. Unel / UCI - USA

S. Sekmen / KNU - Korea

H. Prosper / FSU - USA

A. Paul / ICTP - Italy

vCHEP May 2021

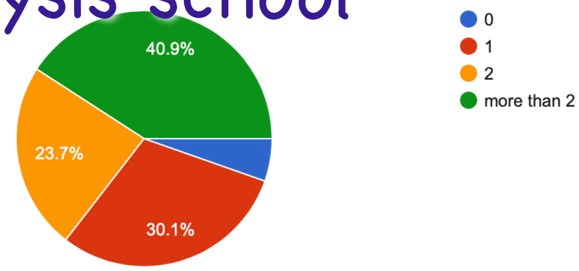
the problem and the solution

- Problem
  - ➡ collider data analysis is necessary but tedious and hard.
    - Analysis provenance difficult to follow
    - Complexities of the frameworks
    - Not easy for the advisor to follow (who doesn't want to read the code)
    - Becomes boring in time: same loops, same deletions...
- How-to solve
  - ➡ Experiments provide recipes for best objects
  - ➡ clean objects, apply some thresholds, make histograms
  - ➡ select interesting events in different regions
- Traditional solution
  - ➡ Write an analysis code using a General Purpose Language
- Could there be a more effective alternative?

vox populi

How many physics analysis frameworks have you worked with?

93 responses



- A survey was conducted in CMS Data Analysis school
  - 75% students, 25% facilitators
  - fairly competent audience
- source of analyst inefficiency is the complexity of analysis frameworks
  - 61%
- community should devote effort to create more accessible analysis ecosystems
  - 82%
- useful to have an analysis ecosystem where the physics information is decoupled from the software framework
  - 62%

What can be done?

a more effective(?) solution

- Higher level language
  - ➡Describe the whole analysis in an unambigous way
  - ➡English-like human understandable simple text file
  - ➡HEP related functions predefined + user additions
- Framework independent
  - ➡focus on physics, not programming technicalities
  - ➡adaptability between exp. & pheno work
  - ➡better communication inter & intra groups / experiments
- Algorithm organizer
  - ➡reflect the conceptual reasoning of particle physicists
  - ➡sections for object and variable definitions
  - ➡sections for event selection in signal and control regions
- We baptized this solution as Analysis Description Language

is ADL idea feasible?

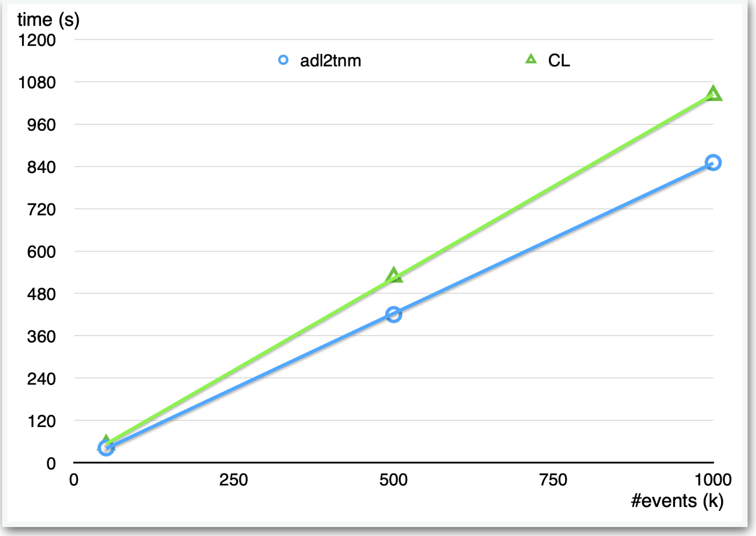
- Discussions
  - ➡ Les Houches workshops 2015-2016
    - LH, Grenoble, CERN
  - ➡ Analysis Description Languages for LHC workshop @FermiLab 2019
  - ➡ CMS OpenData
  - ➡ CERN Analysis Preservation Group
- Exploratory Prototyping
  - ➡ 3+5 seniors + 2 PhD students+ 3 CERN summer students
  - ➡ simple TEXT files for ADL
  - ➡ "Keyword - expression" structure
  - ➡ functions availability
    - arith, logic, trig, optimization
    - HEP related, LVs, histogramming
- Interpreted or Compiled?

ADL analysis example: CMS-SUS-16-037

```
 object jets
  take Jet
  select pT(Jet) > 30
  select abs(Eta(Jet)) < 2.4

object bjets
  take jets
  select BTag(Jet) == 1

object largejets
  take jets
  select pT(jets) > 50

object muons
  take Muon
  select pT(Muon) > 20
  select abs(eta(Muon)) < 2.4

object electrons
  take Electron
  select pT(Electron) > 20
  select abs(eta(Electron)) < 2.5

object leptons : Union(electrons, muons)

define MJ = Sum( m(largejets) )
define MTl = Sqrt( 2*{leptons[0]}pT * MET*(1-
cos({METLV_0}phi - {leptons[0]}phi )))

# Baseline selection
region baseline
  select Size(electrons) >= 0
  select Size(muons     ) >= 0
  select Size(leptons   ) == 1
  select HT(jets) + pT(leptons[0]) > 500
  select MET > 200
  select Size(jets) >= 6
  select Size(bjets) >= 1

# signal region
region SR
  baseline
  select MTl > 140
  select Size(largejets) >= 1
  select MJ > 400
   bin MET 200 350 500
```

interpreted: CutLang

- Run Time Interpreter
  - ➡ No need for compilation after playing with cuts
  - ➡ Quick prototyping & algorithm development
  - ➡ Written in C++, based on ROOT classes
    - about 20% slower than regular C++
    - function pointer tree evaluation
  - ➡ Lex/Yacc for parsing the ADL file

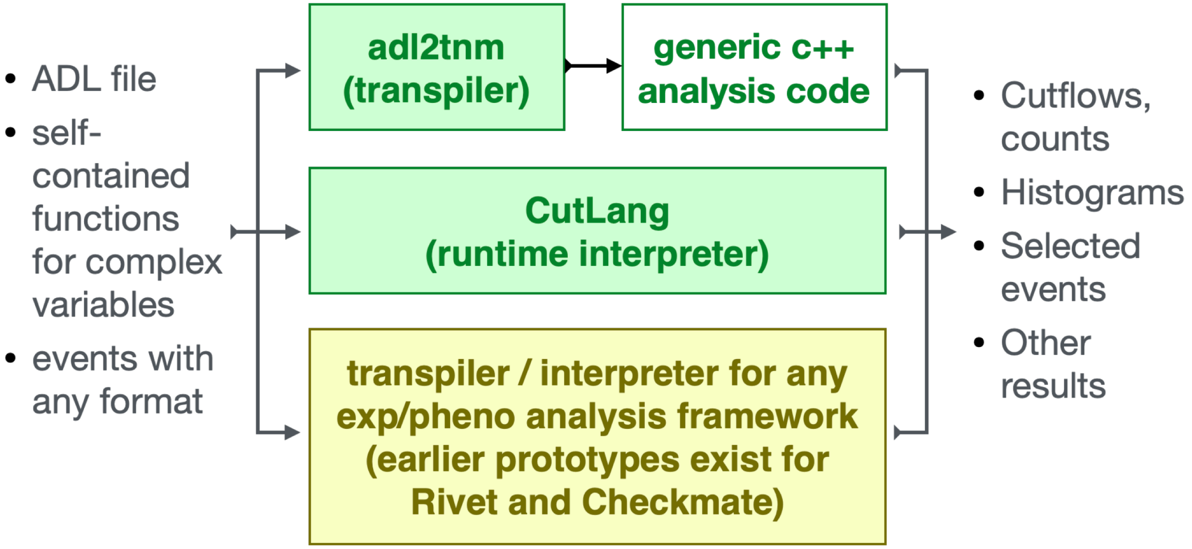


- Environment
  - ➡ Plug-ins for input data types
    - ATLAS, CMS, FCC, OpenData, DELPHES, and more can be added (not automatic)
  - ➡ tools for parallel processing
    - distribute events to cores
    - multiple regions: preselection, sr, cr1, cr2...
  - ➡ Output in ROOT format & contains information for analysis provenance

I want to compile !?

transpiled: adl2tnm

- transpiler
  - ➡Python to "read" ADL and root files
  - ➡convert to C++
  - ➡compile and run


• ADL file
• self-contained functions for complex variables
• events with any format
adl2tnm (transpiler)
generic c++ analysis code
CutLang (runtime interpreter)
transpiler / interpreter for any exp/pheno analysis framework (earlier prototypes exist for Rivet and Checkmate)
• Cutflows, counts
• Histograms
• Selected events
• Other results

- Input type
  - ➡Can work with any simple ntuple format
  - ➡Automatically incorporates input event format into C++

- Status
  - ➡Result compatibility with CutLang and other similar sw.
  - ➡Currently moving from POP to formal grammar for building and parsing.

Are these ever used in real life?

so far

- Availability
  - ➡ ADL : http://adl.web.cern.ch/index.html
  - ➡ CutLang : https://github.com/unelg/CutLang
  - ➡ adl2tnm : https://github.com/hbprosper/adl2tnm
  - ➡ analysis repository : https://github.com/ADL4HEP/ADLLHCanalyses (~15)
- Collaborations with
  - ➡ from various universites across the globe as developers
  - ➡ SModelS group as users
- Phenomenology
  - ➡ HL-LHC & FCC to E6 isosinglet quarks, in Eur. Phys. J. C **81**, 214 (2021)
- Education & Training
  - ➡ Analysis School in Istanbul (Feb.2020) for undergrads, and follow-up evaluation paper, Eur. J. Phys. 42 035802 (2021)
    - "The utilization of an Analysis Description Language greatly lowers the barrier of entry, making actual data analysis accessible even to students with no prior programming experience"
    - "
  - ➡ 26th Vietnam School of Physics (VSOP) in Dec. 2020

what is next?

future prospects

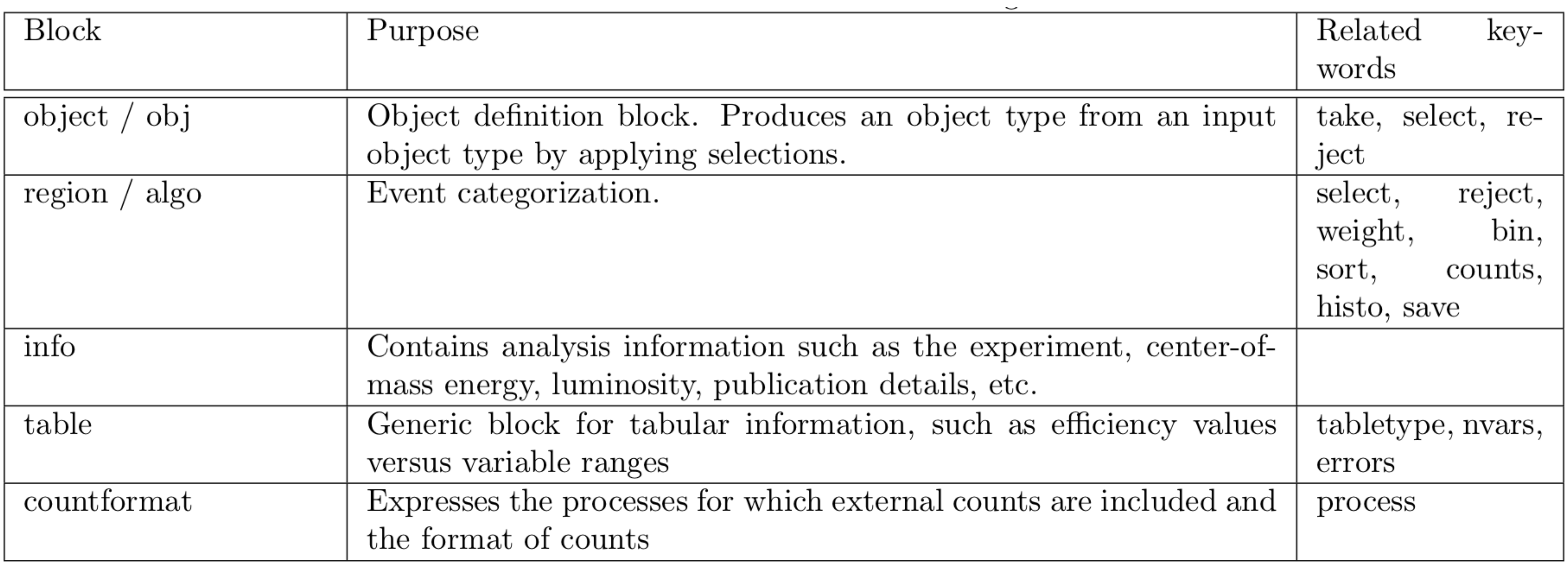
- Planned Development
  - ➡develop ADL into a domain complete language
    - improve functionality & robustness
  - ➡Further unification of interpreter and transpiler through a modern set of tools
  - ➡Automatization of external function and new input type plugin addition
  - ➡Static analysis & differentiable programming
- Physics Studies
  - ➡Enlarge the existing analysis database
    - enlarge external function database
  - ➡Feedback from Experimental & Pheno communities
- New features
  - ➡Inclusion of simple systematics
  - ➡Interfacing to a statistical analysis tool (RooStats?)

Thank You

for your attention

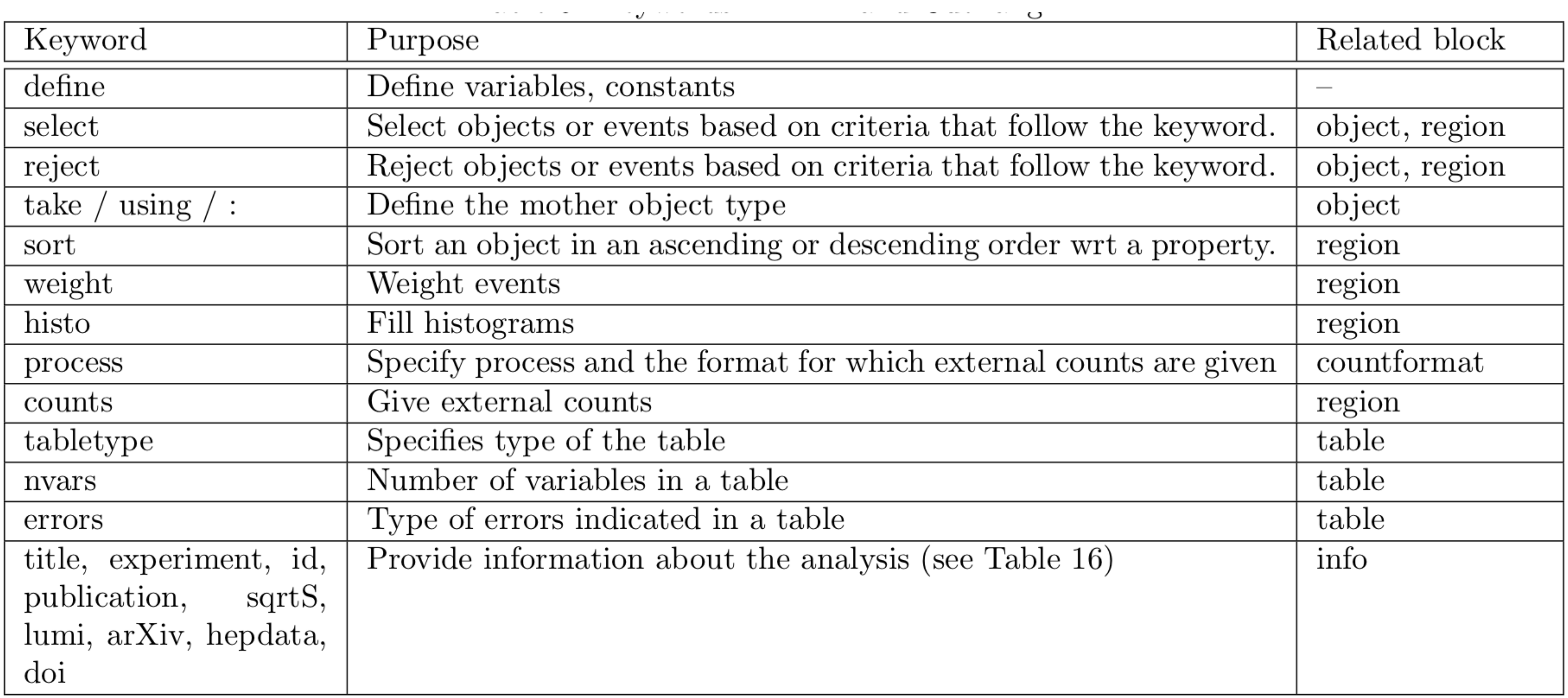
backup slides

blocks



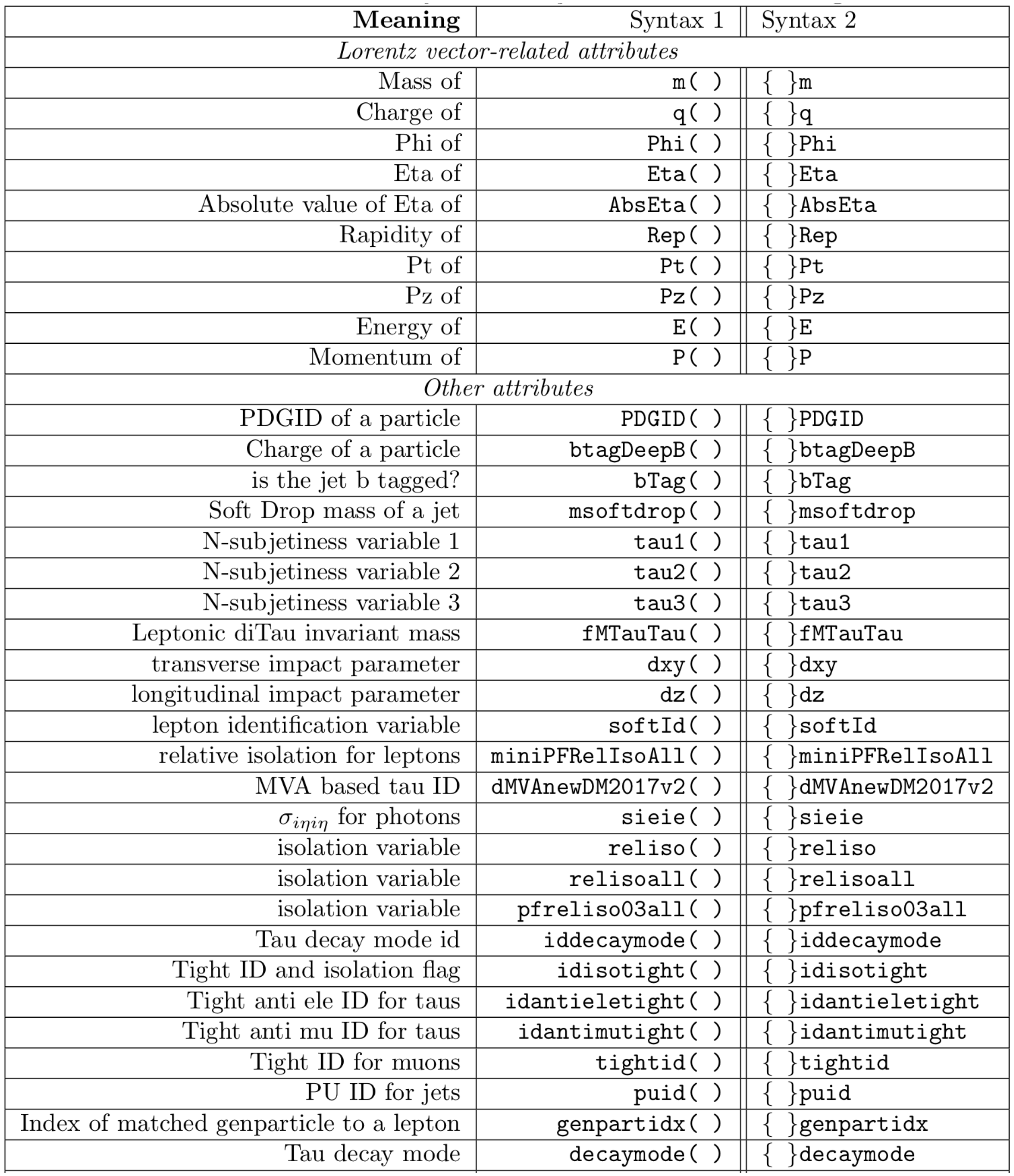
Block	Purpose	Related keywords
object / obj	Object definition block. Produces an object type from an input object type by applying selections.	take, select, reject
region / algo	Event categorization.	select, reject, weight, bin, sort, counts, histo, save
info	Contains analysis information such as the experiment, center-of-mass energy, luminosity, publication details, etc.	
table	Generic block for tabular information, such as efficiency values versus variable ranges	tabletype, nvars, errors
countformat	Expresses the processes for which external counts are included and the format of counts	process

keywords

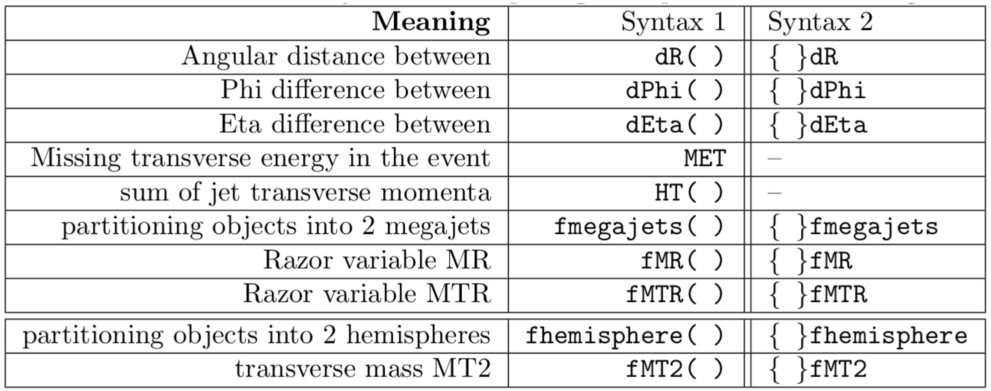


Keyword	Purpose	Related block
define	Define variables, constants	–
select	Select objects or events based on criteria that follow the keyword.	object, region
reject	Reject objects or events based on criteria that follow the keyword.	object, region
take / using / :	Define the mother object type	object
sort	Sort an object in an ascending or descending order wrt a property.	region
weight	Weight events	region
histo	Fill histograms	region
process	Specify process and the format for which external counts are given	countformat
counts	Give external counts	region
tabletype	Specifies type of the table	table
nvars	Number of variables in a table	table
errors	Type of errors indicated in a table	table
title, experiment, id, publication, sqrtS, lumi, arXiv, hepdata, doi	Provide information about the analysis (see Table 16)	info

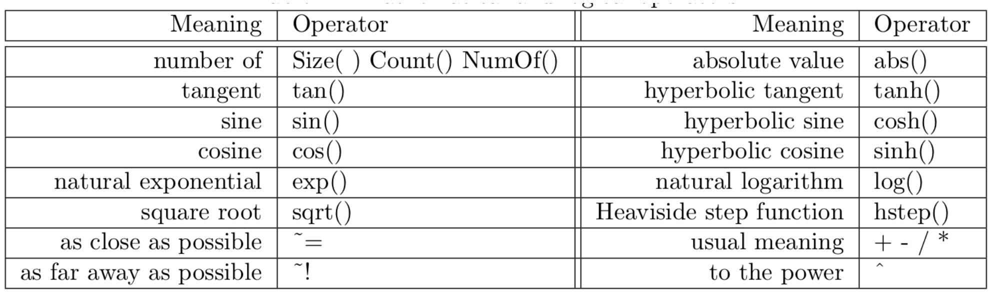
functions



Meaning	Syntax 1	Syntax 2
Lorentz vector-related attributes		
Mass of	m( )	{ }m
Charge of	q( )	{ }q
Phi of	Phi( )	{ }Phi
Eta of	Eta( )	{ }Eta
Absolute value of Eta of	AbsEta( )	{ }AbsEta
Rapidity of	Rep( )	{ }Rep
Pt of	Pt( )	{ }Pt
Pz of	Pz( )	{ }Pz
Energy of	E( )	{ }E
Momentum of	P( )	{ }P
Other attributes		
PDGID of a particle	PDGID( )	{ }PDGID
Charge of a particle	btagDeepB( )	{ }btagDeepB
is the jet b tagged?	bTag( )	{ }bTag
Soft Drop mass of a jet	msoftdrop( )	{ }msoftdrop
N-subjetiness variable 1	tau1( )	{ }tau1
N-subjetiness variable 2	tau2( )	{ }tau2
N-subjetiness variable 3	tau3( )	{ }tau3
Leptonic diTau invariant mass	fMTauTau( )	{ }fMTauTau
transverse impact parameter	dxy( )	{ }dxy
longitudinal impact parameter	dz( )	{ }dz
lepton identification variable	softId( )	{ }softId
relative isolation for leptons	miniPFRelIsoAll( )	{ }miniPFRelIsoAll
MVA based tau ID	dMVAnewDM2017v2( )	{ }dMVAnewDM2017v2
$\sigma_{i\eta i\eta}$ for photons	sieie( )	{ }sieie
isolation variable	reliso( )	{ }reliso
isolation variable	relisoall( )	{ }relisoall
isolation variable	pfreliso03all( )	{ }pfreliso03all
Tau decay mode id	iddecaymode( )	{ }iddecaymode
Tight ID and isolation flag	idisotight( )	{ }idisotight
Tight anti ele ID for taus	idantieletight( )	{ }idantieletight
Tight anti mu ID for taus	idantimutight( )	{ }idantimutight
Tight ID for muons	tightid( )	{ }tightid
PU ID for jets	puid( )	{ }puid
Index of matched genparticle to a lepton	genpartidx( )	{ }genpartidx
Tau decay mode	decaymode( )	{ }decaymode

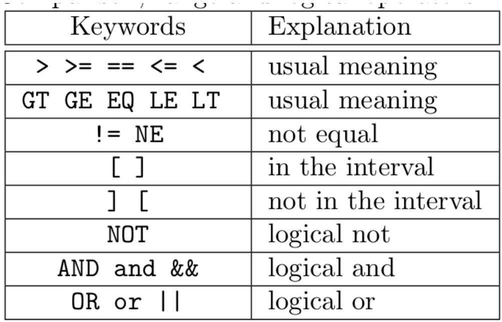


Meaning	Syntax 1	Syntax 2
Angular distance between	dR( )	{ }dR
Phi difference between	dPhi( )	{ }dPhi
Eta difference between	dEta( )	{ }dEta
Missing transverse energy in the event	MET	–
sum of jet transverse momenta	HT( )	–
partitioning objects into 2 megajets	fmegajets( )	{ }fmegajets
Razor variable MR	fMR( )	{ }fMR
Razor variable MTR	fMTR( )	{ }fMTR
partitioning objects into 2 hemispheres	fhemisphere( )	{ }fhemisphere
transverse mass MT2	fMT2( )	{ }fMT2



Meaning	Operator	Meaning	Operator
number of	Size( ) Count() NumOf()	absolute value	abs()
tangent	tan()	hyperbolic tangent	tanh()
sine	sin()	hyperbolic sine	cosh()
cosine	cos()	hyperbolic cosine	sinh()
natural exponential	exp()	natural logarithm	log()
square root	sqrt()	Heaviside step function	hstep()
as close as possible	˜=	usual meaning	+ - / *
as far away as possible	˜!	to the power	ˆ

Operation	Operator
Comparison operators	> < => =< == != [] (include)][(exclude)
Mathematical operators	+ - * / ^
Logical operators	and or not
Ternary operator	condition ? truecase : falsecase
Optimization operators	~= (closest to) ~! (furthest from)
Lorentz vector addition	LV1 + LV2 LV1 LV2

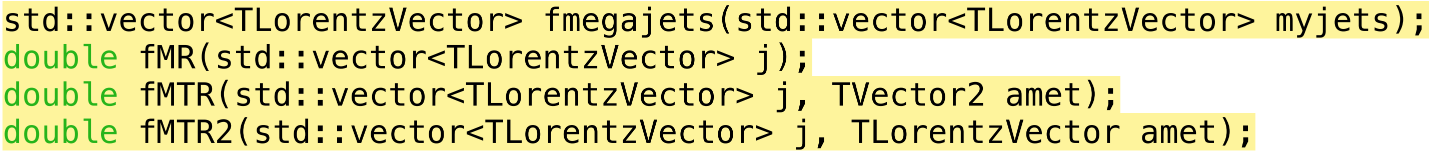


Keywords	Explanation
> >= == <= <	usual meaning
GT GE EQ LE LT	usual meaning
!= NE	not equal
[ ]	in the interval
] [	not in the interval
NOT	logical not
AND and &&	logical and
OR or ||	logical or

User (external) functions

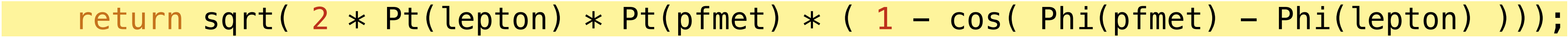
- User defined selection functions are somewhat difficult to incorporate into an interpreter

- Currently we define a user function type and compile it in.
    - ➡ CutLang provides the means to do this automatically
    - ➡ Some functions are pre-integrated:

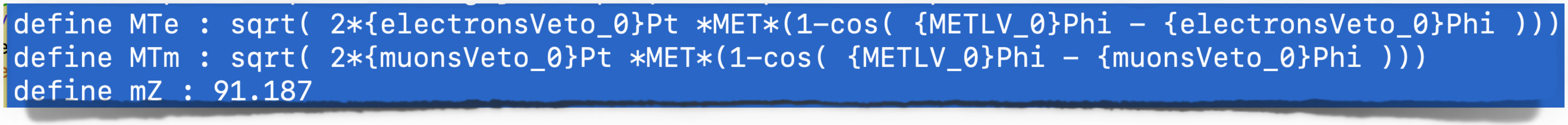


```
std::vector<TLorentzVector> fmegajets(std::vector<TLorentzVector> myjets);
double fMR(std::vector<TLorentzVector> j);
double fMTR(std::vector<TLorentzVector> j, TVector2 amet);
double fMTR2(std::vector<TLorentzVector> j, TLorentzVector amet);
```

- Simple functions can be added using math functions:

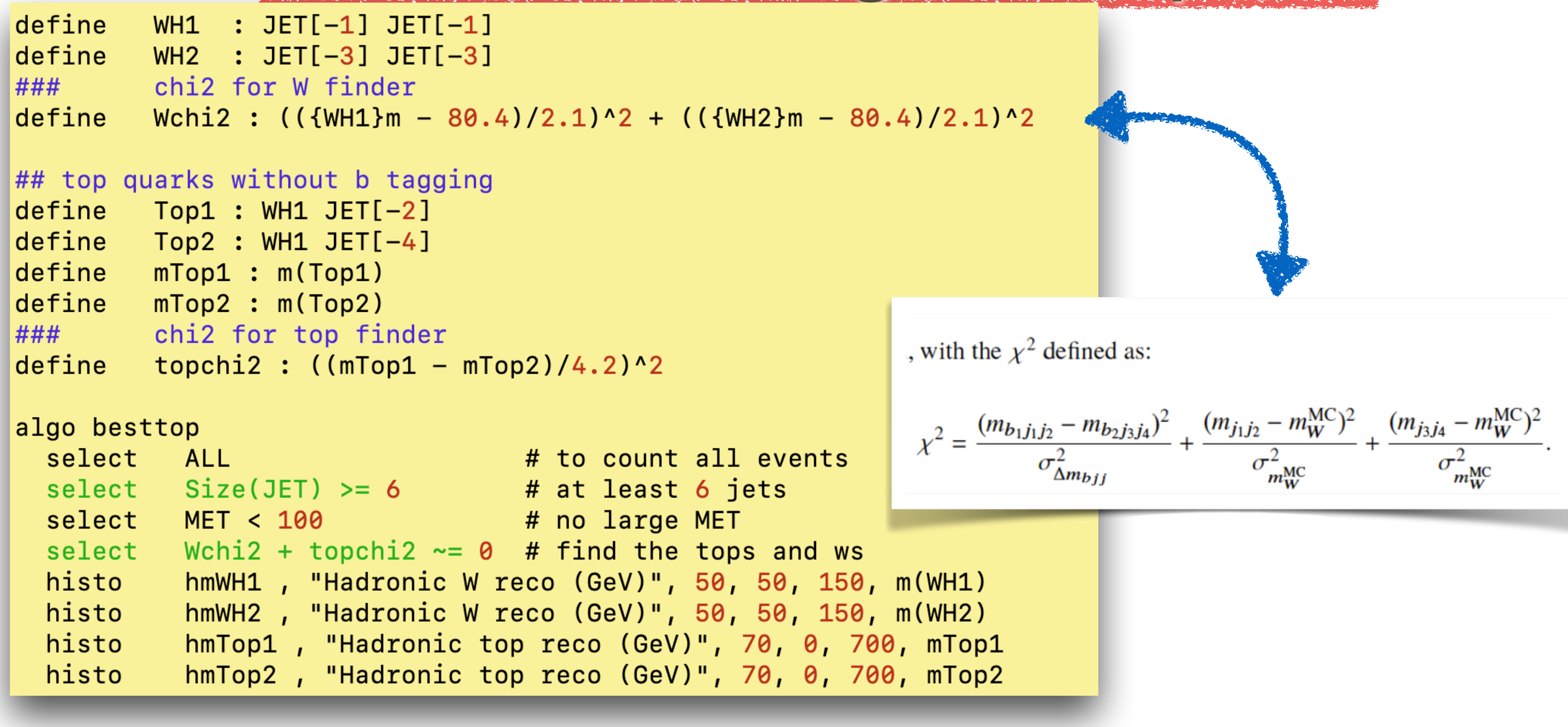


```
    return sqrt( 2 * Pt(lepton) * Pt(pfmet) * ( 1 - cos( Phi(pfmet) - Phi(lepton) )));
```



```
define MTe : sqrt( 2*{electronsVeto_0}Pt *MET*(1-cos( {METLV_0}Phi - {electronsVeto_0}Phi )))
define MTm : sqrt( 2*{muonsVeto_0}Pt *MET*(1-cos( {METLV_0}Phi - {muonsVeto_0}Phi )))
define mZ : 91.187
```

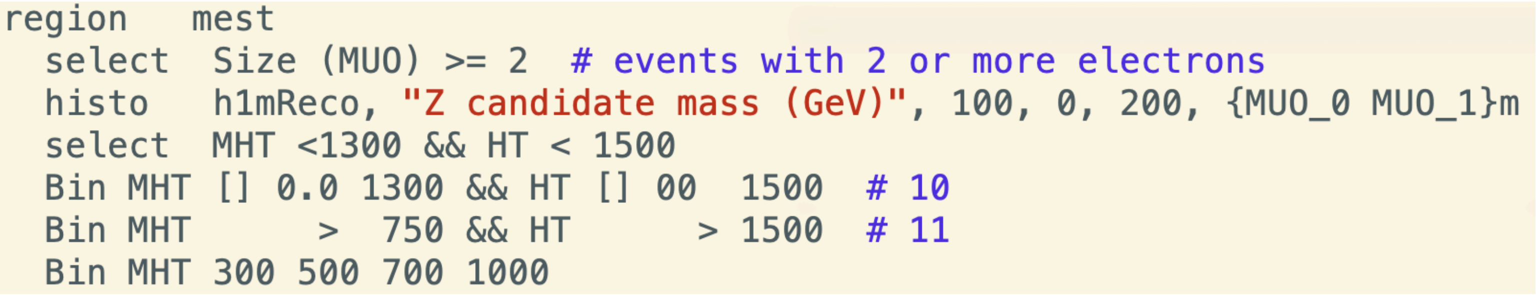
ttbar & binning example



```
define   WH1  : JET[-1] JET[-1]
define   WH2  : JET[-3] JET[-3]
###      chi2 for W finder
define   Wchi2 : (({WH1}m - 80.4)/2.1)^2 + (({WH2}m - 80.4)/2.1)^2

## top quarks without b tagging
define   Top1 : WH1 JET[-2]
define   Top2 : WH1 JET[-4]
define   mTop1 : m(Top1)
define   mTop2 : m(Top2)
###      chi2 for top finder
define   topchi2 : ((mTop1 - mTop2)/4.2)^2

algo besttop
  select   ALL                   # to count all events
  select   Size(JET) >= 6        # at least 6 jets
  select   MET < 100             # no large MET
  select   Wchi2 + topchi2 ~= 0  # find the tops and ws
  histo    hmWH1 , "Hadronic W reco (GeV)", 50, 50, 150, m(WH1)
  histo    hmWH2 , "Hadronic W reco (GeV)", 50, 50, 150, m(WH2)
  histo    hmTop1 , "Hadronic top reco (GeV)", 70, 0, 700, mTop1
  histo    hmTop2 , "Hadronic top reco (GeV)", 70, 0, 700, mTop2
```

, with the χ^2 defined as:

$$\chi^2 = \frac{(m_{b_1j_1j_2} - m_{b_2j_3j_4})^2}{\sigma^2_{\Delta m_{bjj}}} + \frac{(m_{j_1j_2} - m_W^{\mathrm{MC}})^2}{\sigma^2_{m_W^{\mathrm{MC}}}} + \frac{(m_{j_3j_4} - m_W^{\mathrm{MC}})^2}{\sigma^2_{m_W^{\mathrm{MC}}}}.$$



```
region   mest
  select   Size (MUO) >= 2  # events with 2 or more electrons
  histo    h1mReco, "Z candidate mass (GeV)", 100, 0, 200, {MUO_0 MUO_1}m
  select   MHT <1300 && HT < 1500
  Bin MHT [] 0.0 1300 && HT [] 00  1500  # 10
  Bin MHT        >  750 && HT       > 1500  # 11
  Bin MHT 300 500 700 1000
```

continuity of the effort

- Funding Agencies
  - ➡Korea, National Research Foundation : awarded 360k $ for 4 years
  - ➡USA, National Science Foundation : proposal submitted, expecting results this summer
  - ➡TR, Reseach and Scientific Council : proposal submitted, expecting results this summer
- Collaboration with computing professionals
- Usage
  - ➡already used in 2 Atlas exotics analyses