第四届新物理数值计算讲习班

探测器快速模拟

朱 睿

2026 年 7 月 8 日

李政道研究所
TSUNG-DAO LEE INSTITUTE

目录

Content

对撞机探测器快速模拟

PART 01

为什么？

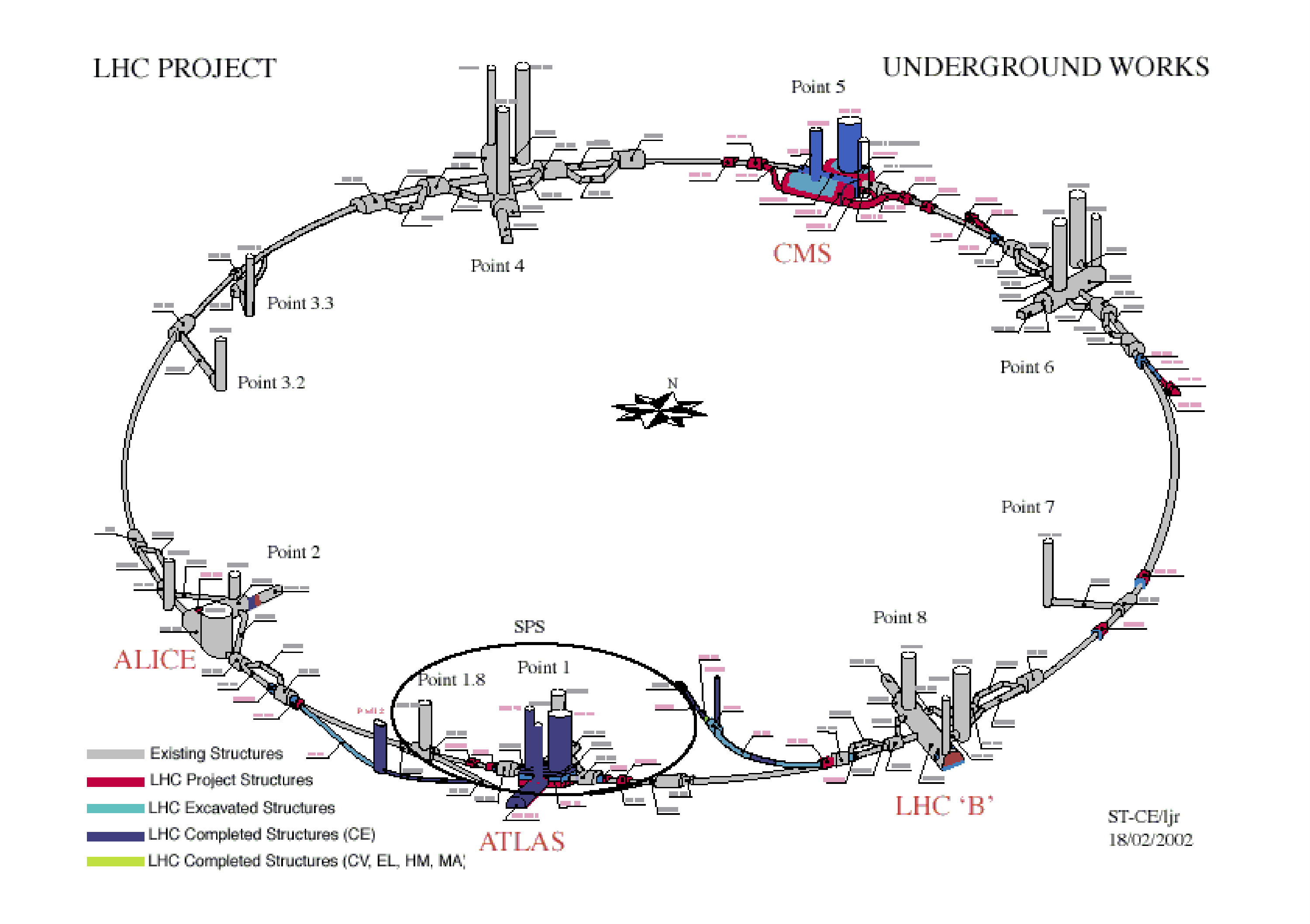
TSUNG-DAO LEE INSTITUTE

对撞机上发生了什么

- 高能粒子对撞

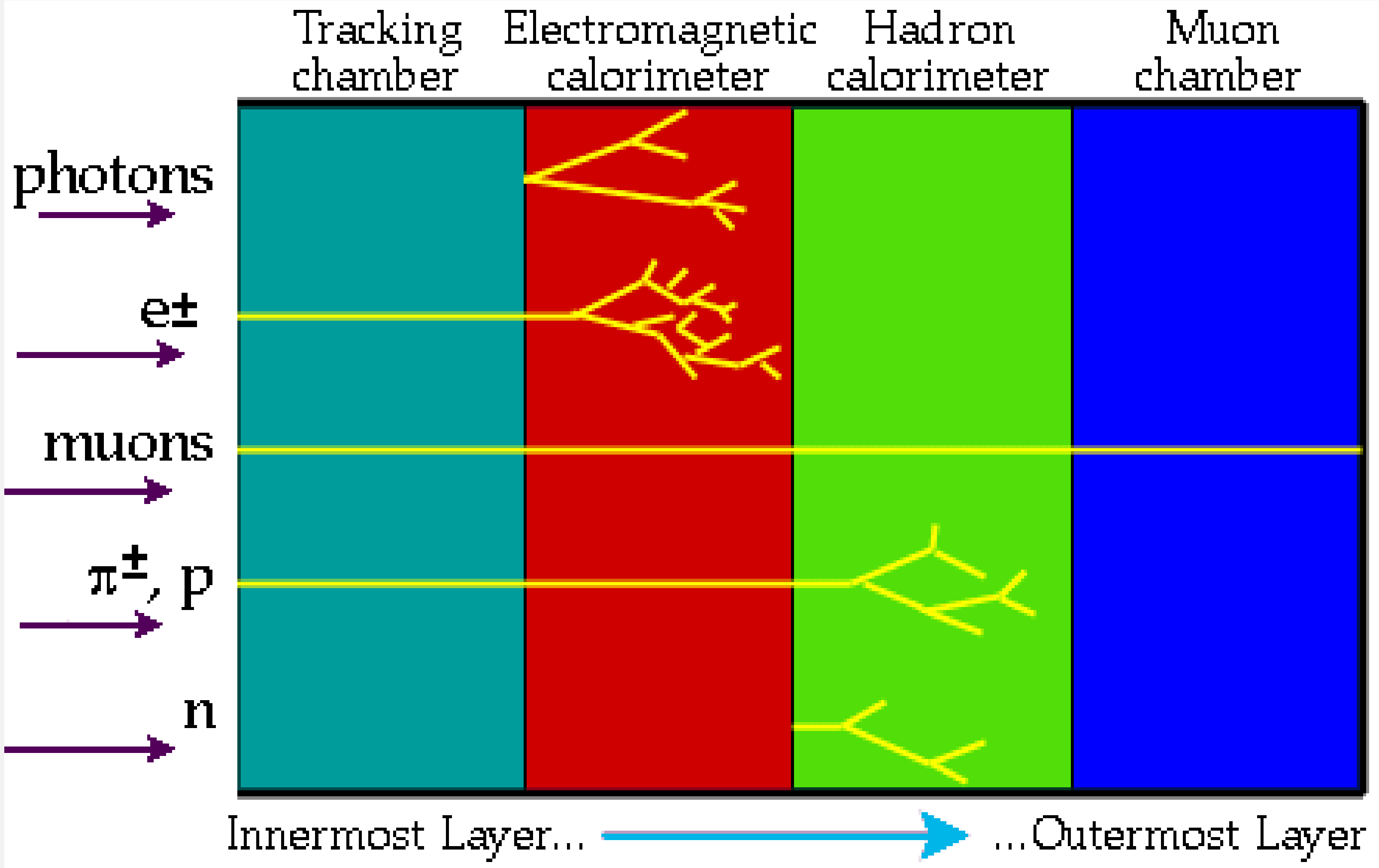
通过各种过程产生不同的粒子，大多数寿命极短，只有少数几种寿命大于 10^{-8}秒，如：

- ✓ 光子
- ✓ 电子（$e^{\pm}$）
- ✓ μ子（$\mu^{\pm}: 2\times 10^{-6}$ s）
- ✓ π介子（$\pi^{\pm}: 2.6\times 10^{-8}$ s）
- ✓ K介子（$K^{\pm}: 2\times 10^{-8}$ s）
- ✓ 质子和反质子



对撞机上发生了什么

- 不同粒子与探测器的相互作用不同



对撞机上发生了什么

- 利用磁谱仪进行粒子鉴别、测量

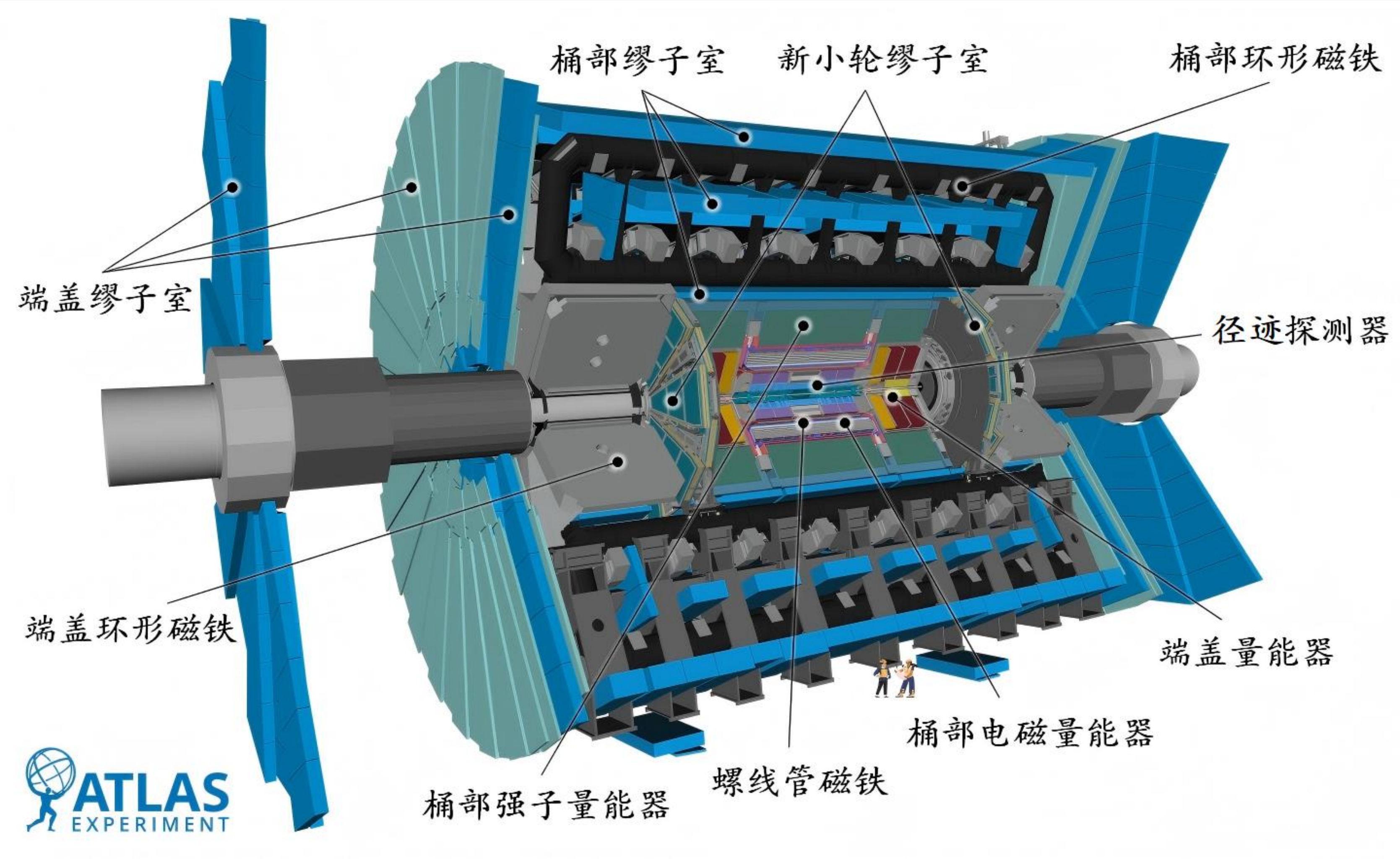
子探测器	测量参数	功能
μ子计数器	μ位置	μ鉴别
强子量能器	强子能量	强子鉴别
线圈和磁铁	p、dE/dx	
电磁量能器	光子、电子能量	光子、电子鉴别
切仑科夫计数器	速度	π、K、质子分辨
飞行时间计数器	速度	π、K、质子分辨
中心径迹室	p、dE/dx	π、K、质子分辨
顶点探测器	对撞和衰变顶点位置	b、c夸克分辨

无损探测（切仑科夫计数器、飞行时间计数器、中心径迹室、顶点探测器）

对撞点

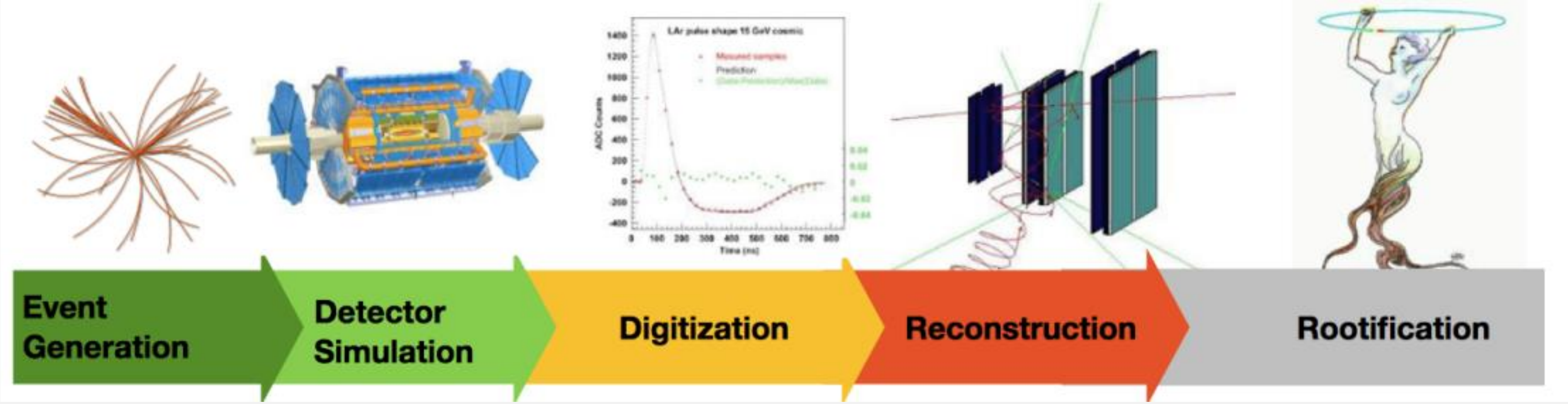
对撞机上发生了什么

- **ATLAS**
  - ✓ 径迹探测器
  - ✓ 电磁量能器
  - ✓ 强子量能器
  - ✓ 缪子探测器
  - ✓ 磁铁



探测器模拟

- 对撞机上过程的全模拟



事例产生子:
- ✓ MadGraph
- ✓ Pythia
- ✓ Herwig
- ✓ Sherpa
- …

探测器模拟
- 探测器建模
- 粒子与物质相互作用：粒子在探测器中的传输和反应过程

电子学模拟
- 触发判选系统模拟
- 数据读出系统模拟

重建

分析

Michele Selvaggi，MC4BSM

探测器模拟

- 为什么进行探测器模拟？

  发现新物理理论精确预言的——particle level

  实验收集的——detector level

- 为什么要进行探测器快速模拟？

✓ 平衡计算成本与效率

  全模拟 GEANT4——100 s /event

  快速模拟 Delphes——10 ms /event

✓ 适用于：

  - 新物理模型大规模参数扫描
  - 大规模产生蒙卡样本
  - 唯象预言实验灵敏度和排除限
  - 事例筛选和分析策略研究
  - 不同实验方案快速比较

  ……

PART 02

怎么做？

TSUNG-DAO LEE INSTITUTE

Delphes的安装与使用

- 网址

https://delphes.github.io/

- 安装

需要先安装gcc、tcl、root，然后

```
wget http://cp3.irmp.ucl.ac.be/downloads/Delphes-3.5.1.tar.gz
tar -zxf Delphes-3.5.1.tar.gz

cd Delphes-3.5.1
make -j 4
```

或者

```
cd Delphes-3.5.1
mkdir build
cd build

cmake -DCMAKE_INSTALL_PREFIX=../install ..
make -j 4 install
```

Delphes的安装与使用

李政道研究所 TSUNG-DAO LEE INSTITUTE

- 输入文件

  LHE、HepMC、StdHep、ProMC

- 输出文件

  ROOT

- 运行

  根据不同的输入文件格式选择可执行文件，例如对于LHE文件：

```
./DelphesLHEF cards/delphes_card_CMS.tcl delphes_output.root input.lhef
```

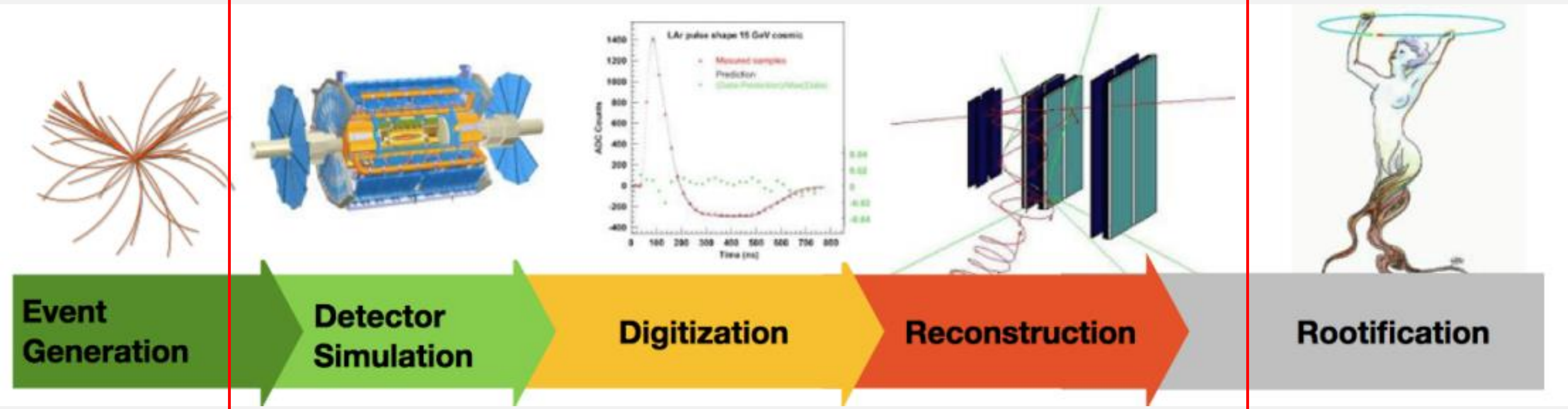
基于tcl脚本的配置文件　输出文件　输入文件

Delphes参数化模拟

李政道研究所 TSUNG-DAO LEE INSTITUTE

- 参数化模拟

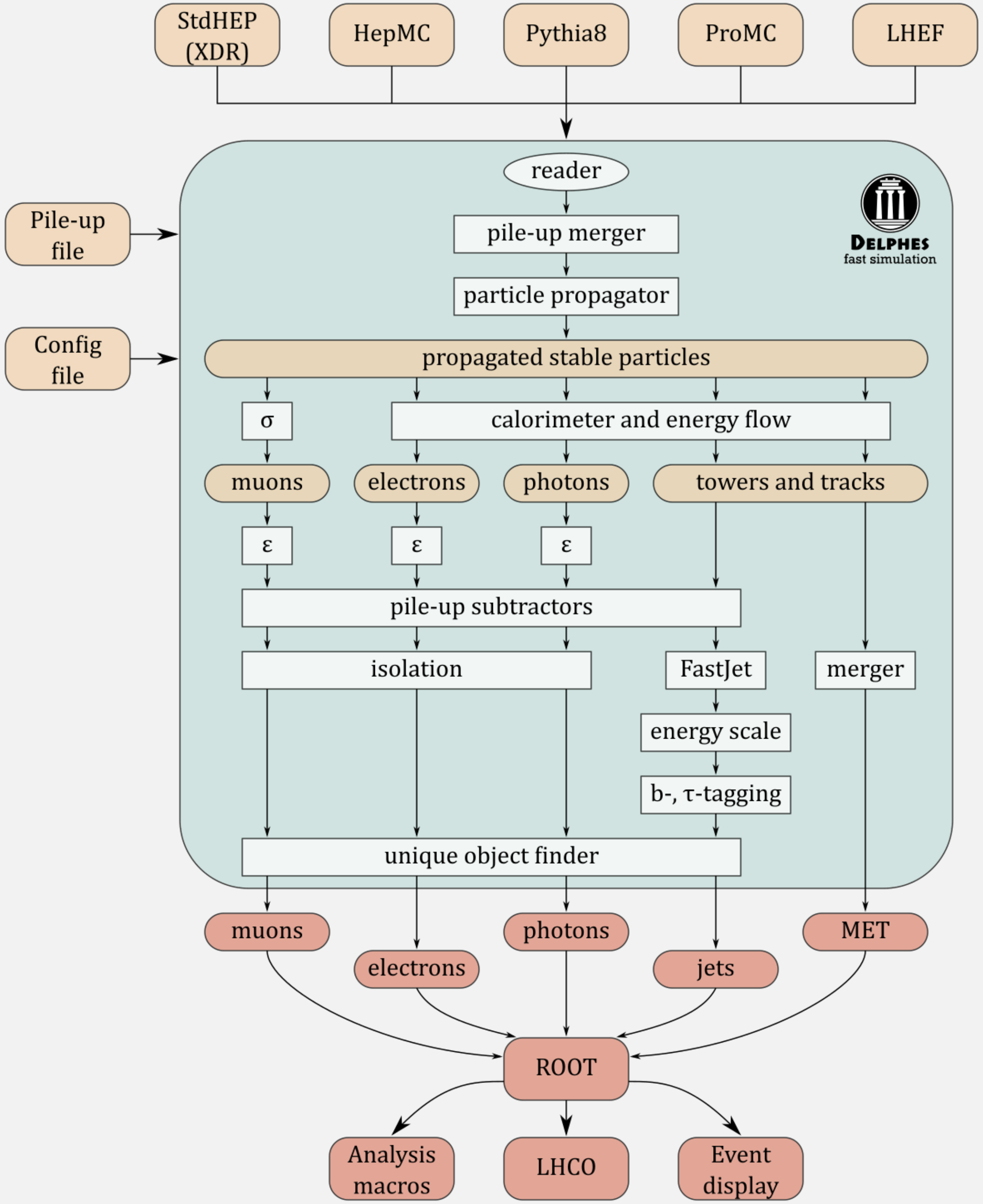
参数化模拟探测器、电子学系统、粒子重建过程的效果。
不考虑具体的相互作用。

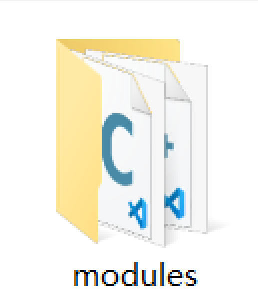


Delphes参数化模拟

- 模块化的程序结构

模块化系统允许用户通过配置tcl文件来调用模块、添加模块、更改数据流以及更改输出信息。





Delphes参数化模拟

- 配置文件举例

以 delphes_card_ATLAS.tcl 为例：

ExecutionPath可以设置各个模块的执行顺序

```
set ExecutionPath {
  ParticlePropagator

  ChargedHadronTrackingEfficiency
  ElectronTrackingEfficiency
  MuonTrackingEfficiency

  ChargedHadronMomentumSmearing
  ElectronMomentumSmearing
  MuonMomentumSmearing

  TrackMerger

  ECal
  HCal

  Calorimeter
  EFlowMerger
  EFlowFilter

  PhotonEfficiency
  PhotonIsolation

  ElectronFilter
  ElectronEfficiency
  ElectronIsolation

  ChargedHadronFilter

  MuonEfficiency
  MuonIsolation

  MissingET

  NeutrinoFilter
  GenJetFinder
  GenMissingET

  FastJetFinder

  JetEnergyScale

  JetFlavorAssociation

  BTagging
  TauTagging

  UniqueObjectFinder

  ScalarHT

  TreeWriter
}
```

Delphes参数化模拟

6 ParticlePropagator

- **ParticlePropagator**

```
####################################
# Propagate particles in cylinder
####################################

module ParticlePropagator ParticlePropagator {
  set InputArray Delphes/stableParticles

  set OutputArray stableParticles
  set ChargedHadronOutputArray chargedHadrons
  set ElectronOutputArray electrons
  set MuonOutputArray muons

  # radius of the magnetic field coverage, in m
  set Radius 1.15
  # half-length of the magnetic field coverage, in m
  set HalfLength 3.51

  # magnetic field
  set Bz 2.0
}
```

① 粒子在磁场中的传播、

② 粒子分组：

chargedHadrons，electrons，muons等

Delphes参数化模拟

```
ChargedHadronTrackingEfficiency
ElectronTrackingEfficiency
MuonTrackingEfficiency
```

- **Efficiency**

```
#################################
# Electron tracking efficiency
#################################

module Efficiency ElectronTrackingEfficiency {
  set InputArray ParticlePropagator/electrons
  set OutputArray electrons

  # set EfficiencyFormula {efficiency formula as a function of eta and pt}

  # tracking efficiency formula for electrons
  set EfficiencyFormula {                                                     (pt <= 0.1)   * (0.00) +
                                                   (abs(eta) <= 1.5) * (pt > 0.1   && pt <= 1.0)   * (0.73) +
                                                   (abs(eta) <= 1.5) * (pt > 1.0   && pt <= 1.0e2) * (0.95) +
                                                   (abs(eta) <= 1.5) * (pt > 1.0e2)                * (0.99) +
                             (abs(eta) > 1.5 && abs(eta) <= 2.5) * (pt > 0.1   && pt <= 1.0)   * (0.50) +
                             (abs(eta) > 1.5 && abs(eta) <= 2.5) * (pt > 1.0   && pt <= 1.0e2) * (0.83) +
                             (abs(eta) > 1.5 && abs(eta) <= 2.5) * (pt > 1.0e2)                * (0.90) +
                             (abs(eta) > 2.5)                                                  * (0.00)}
}
```

Delphes参数化模拟

25	PhotonEfficiency
29	ElectronEfficiency
34	MuonEfficiency

- Efficiency

```
####################
# Photon efficiency
####################

module Efficiency PhotonEfficiency {
  set InputArray ECal/eflowPhotons
  set OutputArray photons

  # set EfficiencyFormula {efficiency formula as a function of eta and pt}

  # efficiency formula for photons
  set EfficiencyFormula {                                           (pt <= 10.0) * (0.00) +
                                            (abs(eta) <= 1.5) * (pt > 10.0)  * (0.95) +
                          (abs(eta) > 1.5 && abs(eta) <= 2.5) * (pt > 10.0)  * (0.85) +
                          (abs(eta) > 2.5)                                   * (0.00)}
}
```

Delphes参数化模拟

李政道研究所 TSUNG-DAO LEE INSTITUTE

```
ChargedHadronMomentumSmearing
ElectronMomentumSmearing
MuonMomentumSmearing
```

- **MomentumSmearing**

```
###################################
# Momentum resolution for electrons
###################################

module MomentumSmearing ElectronMomentumSmearing {
  set InputArray ElectronTrackingEfficiency/electrons
  set OutputArray electrons

  # set ResolutionFormula {resolution formula as a function of eta and energy}

  # resolution formula for electrons
  set ResolutionFormula {                  (abs(eta) <= 0.5) * (pt > 0.1) * sqrt(0.03^2 + pt^2*1.3e-3^2) +
                         (abs(eta) > 0.5 && abs(eta) <= 1.5) * (pt > 0.1) * sqrt(0.05^2 + pt^2*1.7e-3^2) +
                         (abs(eta) > 1.5 && abs(eta) <= 2.5) * (pt > 0.1) * sqrt(0.15^2 + pt^2*3.1e-3^2)}
}
```

MomentumSmearing

```
// apply smearing formula
//pt = gRandom->Gaus(pt, fFormula->Eval(pt, eta, phi, e) * pt);

res = (res > 1.0) ? 1.0 : res;

pt = LogNormal(pt, res * pt);
```

Delphes参数化模拟

- Merger:信息整合

```
set ExecutionPath {
 ParticlePropagator

  ChargedHadronTrackingEfficiency
  ElectronTrackingEfficiency
  MuonTrackingEfficiency

  ChargedHadronMomentumSmearing
  ElectronMomentumSmearing
  MuonMomentumSmearing

  TrackMerger

  ECal
  HCal

  Calorimeter
  EFlowMerger
  EFlowFilter

  PhotonEfficiency
  PhotonIsolation

  ElectronFilter
  ElectronEfficiency
  ElectronIsolation

  ChargedHadronFilter

  MuonEfficiency
  MuonIsolation

  MissingET

  NeutrinoFilter
  GenJetFinder
  GenMissingET

  FastJetFinder

  JetEnergyScale

  JetFlavorAssociation

  BTagging
  TauTagging

  UniqueObjectFinder

  ScalarHT

  TreeWriter
}
```

Delphes参数化模拟

```
ECal
HCal
```

- **SimpleCalorimeter**

HCAL:

```
# 10 degrees towers
set PhiBins {}
for {set i -18} {$i <= 18} {incr i} {
  add PhiBins [expr {$i * $pi/18.0}]
}
foreach eta {-3.2 -2.5 -2.4 -2.3 -2.2 -2.1 -2 -1.9 -1.8 -1.7 -1.6 -1.5 -1.4 -1.3 -1.2
   -1.1 -1 -0.9 -0.8 -0.7 -0.6 -0.5 -0.4 -0.3 -0.2 -0.1 0 0.1 0.2 0.3 0.4 0.5 0.6 0.7
    0.8 0.9 1 1.1 1.2 1.3 1.4 1.5 1.6 1.7 1.8 1.9 2 2.1 2.2 2.3 2.4 2.5 2.6 3.3} {
  add EtaPhiBins $eta $PhiBins
}

# 20 degrees towers
set PhiBins {}
for {set i -9} {$i <= 9} {incr i} {
  add PhiBins [expr {$i * $pi/9.0}]
}
foreach eta {-4.9 -4.7 -4.5 -4.3 -4.1 -3.9 -3.7 -3.5 -3.3 -3 -2.8 -2.6 2.8 3 3.2 3.5
  3.7 3.9 4.1 4.3 4.5 4.7 4.9} {
  add EtaPhiBins $eta $PhiBins
}
```

Delphes参数化模拟

```
ECal
HCal
```

- **SimpleCalorimeter**

  HCAL:

```
# energy fractions for K0short and Lambda
add EnergyFraction {310} {0.7}
add EnergyFraction {3122} {0.7}

# http://arxiv.org/pdf/hep-ex/0004009v1
# http://villaolmo.mib.infn.it/ICATPP9th_2005/Calorimetry/Schram.p.pdf
# set HCalResolutionFormula {resolution formula as a function of eta and energy}
set ResolutionFormula {                    (abs(eta) <= 1.7) * sqrt(energy^2*0.0302^2 + energy*0.5205^2 + 1.59^2) +
                         (abs(eta) > 1.7 && abs(eta) <= 3.2) * sqrt(energy^2*0.0500^2 + energy*0.706^2) +
                         (abs(eta) > 3.2 && abs(eta) <= 4.9) * sqrt(energy^2*0.09420^2 + energy*1.00^2)}
```

Delphes参数化模拟

23	EFlowFilter	28	ElectronFilter
32	ChargedHadronFilter	39	NeutrinoFilter

李政道研究所 TSUNG-DAO LEE INSTITUTE

- PdgCodeFilter

```
module PdgCodeFilter EFlowFilter {
  set InputArray EFlowMerger/eflow
  set OutputArray eflow

  add PdgCode {11}
  add PdgCode {-11}
  add PdgCode {13}
  add PdgCode {-13}
}
```

```
module PdgCodeFilter ChargedHadronFilter {
  set InputArray HCal/eflowTracks
  set OutputArray chargedHadrons

  add PdgCode {11}
  add PdgCode {-11}
  add PdgCode {13}
  add PdgCode {-13}
}
```

```
module PdgCodeFilter ElectronFilter {
  set InputArray HCal/eflowTracks
  set OutputArray electrons
  set Invert true
  add PdgCode {11}
  add PdgCode {-11}
}
```

```
module PdgCodeFilter NeutrinoFilter {

  set InputArray Delphes/stableParticles
  set OutputArray filteredParticles

  set PTMin 0.0

  add PdgCode {12}
  add PdgCode {14}
  add PdgCode {16}
  add PdgCode {-12}
  add PdgCode {-14}
  add PdgCode {-16}

}
```

Delphes参数化模拟

- **Isolation**

26	PhotonIsolation
30	ElectronIsolation
35	MuonIsolation

```
module Isolation PhotonIsolation {
  set CandidateInputArray PhotonEfficiency/photons
  set IsolationInputArray EFlowFilter/eflow

  set OutputArray photons

  set DeltaRMax 0.5

  set PTMin 0.5

  set PTRatioMax 0.12
}
```

$$I(P) = \frac{\sum\limits_{i \neq P}^{\Delta R < R,\ p_T(i) > p_T^{min}} p_T(i)}{p_T(P)}$$

Delphes参数化模拟

40	GenJetFinder
43	FastJetFinder

TSUNG-DAO LEE INSTITUTE 李政道研究所

- FastJetFinder

```
module FastJetFinder FastJetFinder {
  set InputArray Calorimeter/towers

  set OutputArray jets

  # algorithm: 1 CDFJetClu, 2 MidPoint, 3 SIScone, 4 kt, 5 Cambridge/Aachen, 6 antikt
  set JetAlgorithm 6
  set ParameterR 0.6

  set JetPTMin 20.0
}
```

Delphes参数化模拟

45	JetEnergyScale

李政道研究所 TSUNG-DAO LEE INSTITUTE

- EnergyScale

```
module EnergyScale JetEnergyScale {
  set InputArray FastJetFinder/jets
  set OutputArray jets

  # scale formula for jets
  set ScaleFormula {  sqrt( (3.0 - 0.2*(abs(eta)))^2 / pt + 1.0 )  }
}
```

Delphes参数化模拟

47	JetFlavorAssociation

- JetFlavorAssociation

```
module JetFlavorAssociation JetFlavorAssociation {

  set PartonInputArray Delphes/partons
  set ParticleInputArray Delphes/allParticles
  set ParticleLHEFInputArray Delphes/allParticlesLHEF
  set JetInputArray JetEnergyScale/jets

  set DeltaR 0.5
  set PartonPTMin 1.0
  set PartonEtaMax 2.5

}
```

Delphes参数化模拟

- **BTagging**

```
module BTagging BTagging {
  set JetInputArray JetEnergyScale/jets

  set BitNumber 0

  # add EfficiencyFormula {abs(PDG code)} {efficiency formula as a function of eta and pt}
  # PDG code = the highest PDG code of a quark or gluon inside DeltaR cone around jet axis
  # gluons PDG code has the lowest priority

  # based on ATL-PHYS-PUB-2015-022

  # default efficiency formula (misidentification rate)
  add EfficiencyFormula {0} {0.002+7.3e-06*pt}

  # efficiency formula for c-jets (misidentification rate)
  add EfficiencyFormula {4} {0.20*tanh(0.02*pt)*(1/(1+0.0034*pt))}

  # efficiency formula for b-jets
  add EfficiencyFormula {5} {0.80*tanh(0.003*pt)*(30/(1+0.086*pt))}
}
```

```
50    TauTagging
```

- TrackCountingTauTagging

```
# instructions: {n-prongs} {eff}

# 1 - one prong efficiency
# 2 - two or more efficiency
# -1 - one prong mistag rate
# -2 - two or more mistag rate

set BitNumber 0

# taken from ATL-PHYS-PUB-2015-045 (medium working point)
add EfficiencyFormula {1} {0.70}
add EfficiencyFormula {2} {0.60}
add EfficiencyFormula {-1} {0.02}
add EfficiencyFormula {-2} {0.01}
```

52 UniqueObjectFinder

Tsung-Dao Lee Institute 李政道研究所

- UniqueObjectFinder

```
##########################################################
# Find uniquely identified photons/electrons/tau/jets
##########################################################

module UniqueObjectFinder UniqueObjectFinder {
# earlier arrays take precedence over later ones
# add InputArray InputArray OutputArray
  add InputArray PhotonIsolation/photons photons
  add InputArray ElectronIsolation/electrons electrons
  add InputArray MuonIsolation/muons muons
  add InputArray JetEnergyScale/jets jets
}
```

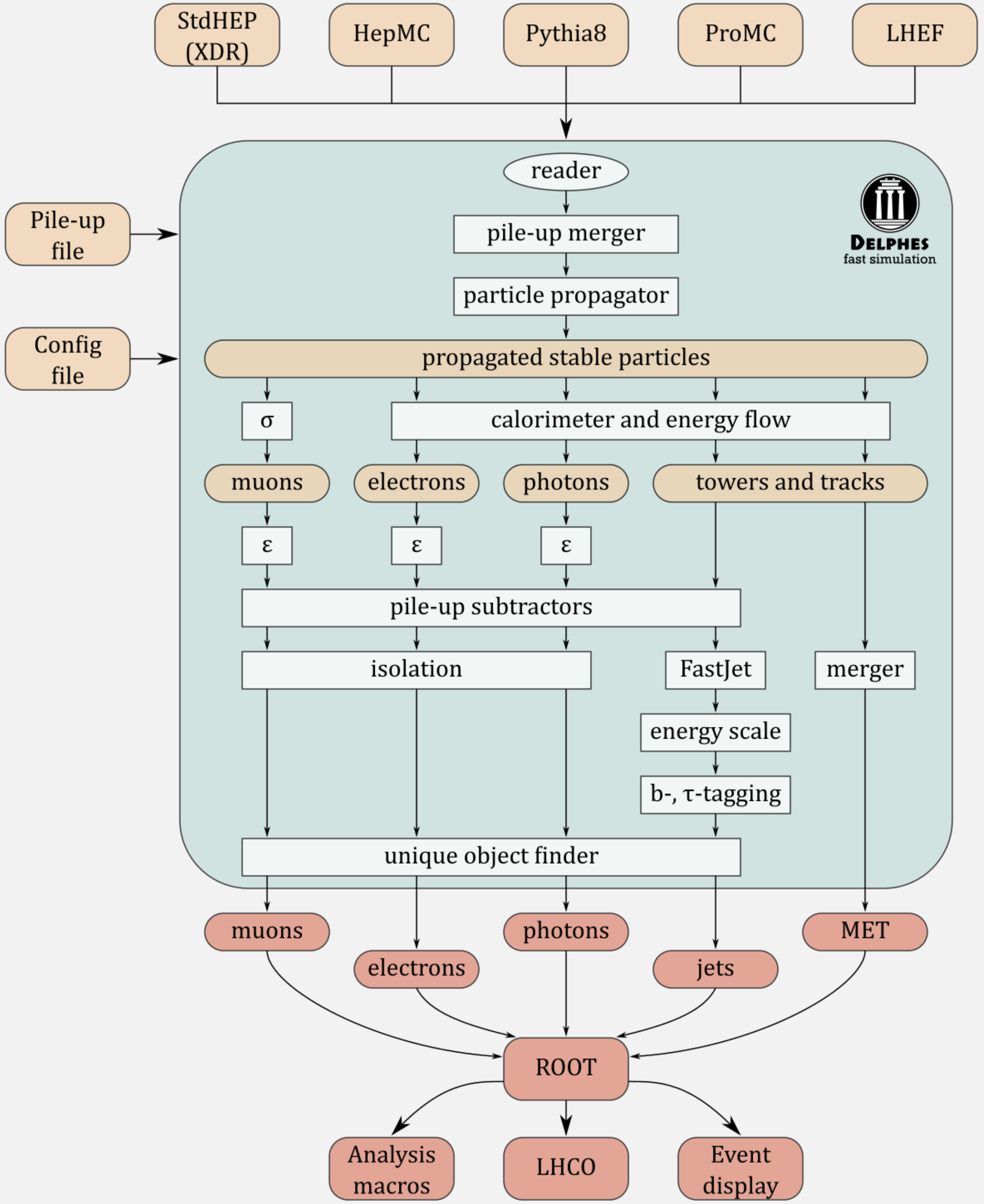
56 TreeWriter

- TreeWriter

```
# tracks, towers and eflow objects are not stored by default in the output.
# if needed (for jet constituent or other studies), uncomment the relevant
# "add Branch ..." lines.

module TreeWriter TreeWriter {
# add Branch InputArray BranchName BranchClass
  add Branch Delphes/allParticles Particle GenParticle

  add Branch TrackMerger/tracks Track Track
  add Branch Calorimeter/towers Tower Tower

  add Branch HCal/eflowTracks EFlowTrack Track
  add Branch ECal/eflowPhotons EFlowPhoton Tower
  add Branch HCal/eflowNeutralHadrons EFlowNeutralHadron Tower

  add Branch GenJetFinder/jets GenJet Jet
  add Branch GenMissingET/momentum GenMissingET MissingET

  add Branch UniqueObjectFinder/jets Jet Jet
  add Branch UniqueObjectFinder/electrons Electron Electron
  add Branch UniqueObjectFinder/photons Photon Photon
  add Branch UniqueObjectFinder/muons Muon Muon
  add Branch MissingET/momentum MissingET MissingET
  add Branch ScalarHT/energy ScalarHT ScalarHT
}
```

Delphes参数化模拟

- 总体流程



添加新的模块

- 添加新模块
  - copy template/example module to the new one:

```
cp modules/ExampleModule.h modules/NewModule.h
cp modules/ExampleModule.cc modules/NewModule.cc
```

  - edit new module:

```
vi modules/NewModule.h
vi modules/NewModule.cc
```

  - add new module to modules/ModulesLinkDef.h:

```
#include "modules/NewModule.h"
...
#pragma link C++ class NewModule+;
```

  - regenerate Makefile:

```
./configure
```

  - rebuild Delphes:

```
make -j 4
```

Pileup

以delphes_card_ATLAS_PileUp.tcl为例：

- **PileUpMerger**
  - ✓ 将Bias事例与硬过程事例混合
  - ✓ 可以设置顶点沿z轴和时间t的分布
  - ✓ 相对于z轴将事件旋转随机角度φ

```
module PileUpMerger PileUpMerger {
  set InputArray Delphes/stableParticles

  set ParticleOutputArray stableParticles
  set VertexOutputArray vertices

  # pre-generated minbias input file
  set PileUpFile MinBias.pileup

  # average expected pile up
  set MeanPileUp 50

   # maximum spread in the beam direction in m
  set ZVertexSpread 0.25

  # maximum spread in time in s
  set TVertexSpread 800E-12

  # vertex smearing formula f(z,t) (z,t need to be respectively given in m,s)
  set VertexDistributionFormula {exp(-(t^2/160e-12^2/2))*exp(-(z^2/0.053^2/2))}
}
```

Pileup

以delphes_card_ATLAS_PileUp.tcl为例：

- **TrackPileUpSubtractor**

```
module TrackPileUpSubtractor TrackPileUpSubtractor {
# add InputArray InputArray OutputArray
  add InputArray Calorimeter/eflowTracks eflowTracks
  add InputArray ElectronFilter/electrons electrons
  add InputArray MuonMomentumSmearing/muons muons

  # assume perfect pile-up subtraction for tracks with |z| > fZVertexResolution
  # Z vertex resolution in m
  set ZVertexResolution {0.0001}
}
```

Pileup

- **FastJetGridMedianEstimator、TrackPileUpSubtractor**

利用fastjet area method剪除pileup对jets的贡献

Use the FastJet Area approach (Cacciari, Salam, Soyez)

- compute ρ = event pile-up density
- jet correction : $pT \rightarrow pT - \rho A$ (JetPileUpSubtractor)
- isolation : $\sum pT \rightarrow \sum pT - \rho\pi R^2$ (Isolation module itself)

简单的事例分析

- 注意：需要 source DelphesEnv.sh
- 以 p p > z > mu+ mu-为例，假设要求对末态有2个 pT >20GeV的缪子，并画它们不变质量的分布：

```
#include <iostream>
#include <sstream>
#include <stdexcept>
using namespace std;

#ifdef __CLING__
R__LOAD_LIBRARY(libDelphes)
#include "classes/DelphesClasses.h"
#include "external/ExRootAnalysis/ExRootTreeReader.h"
#include "external/ExRootAnalysis/ExRootResult.h"
#include "external/ExRootAnalysis/ExRootTreeBranch.h"
#include "TClonesArray.h"
#else
class ExRootTreeReader;
class ExRootResult;
#endif


```

简单的事例分析

```
void drawm()
{
  gSystem->Load("libDelphes");

  // Create chain of root trees
  TChain *t = new TChain("Delphes");
  t->Add("mumu_pythia.root");

  // Create object of class ExRootTreeReader
  ExRootTreeReader *treeReader = new ExRootTreeReader(t);

  // Book the histogram of invariant mass
  TH1F *h = new TH1F( "h", "h", 50, 0., 200. );

  // Get pointers to branches used in this analysis
  TClonesArray *branchEvent = treeReader->UseBranch("Event");
  TClonesArray *branchMuon = treeReader->UseBranch("Muon");

  // Loop over all events
  Long64_t numberOfEntries = treeReader->GetEntries();
  for(Int_t entry = 0; entry < numberOfEntries; ++entry)
  {
    // Load selected branches with data from specified event
    treeReader->ReadEntry(entry);

```

简单的事例分析

```
    TLorentzVector momentum;
    Int_t nMuon = 0;
    for ( unsigned int ii = 0 ; ii < branchMuon->GetEntries() ; ii ++ )
    {
      Muon *mu = (Muon*) branchMuon->At( ii );
      if ( mu->PT > 20. ){
        nMuon ++ ;
        momentum += mu->P4();
      }
    }
    // 2 muons
    if ( nMuon !=2 ) continue;
    // invariant mass of muons
    Double_t inv_mass = momentum.M();
    h->Fill(inv_mass);

  }
  delete treeReader;

  // Draw histogram
  auto c1 = new TCanvas("c1","c1",750,550);
  h->Draw();

}
```

PART 03

试试看！

TSUNG-DAO LEE INSTITUTE

演示

- **p p > z > mu+ mu-**

  MadGraph、Pythia产生事例；
  Delphes模拟；
  画不变质量的分布；
  改变mu的探测效率、分辨率，观察分布变化；
  添加pileup;
  产生p p > mu+ mu-，对不变质量加cut。

- **p p > t t~**

  MadGraph、Pythia产生事例，Delphes模拟；
  查看root文件；
  查看MissingET的分布。

李政道研究所
TSUNG-DAO LEE INSTITUTE

谢 谢！