

ATLAS のシュミレーションの^{まと}纏めと
プロダクション予定

陣内 修 (東工大)

2009/4/1-3

実験理論共同研究会 「LHC が切り拓く新しい物理」

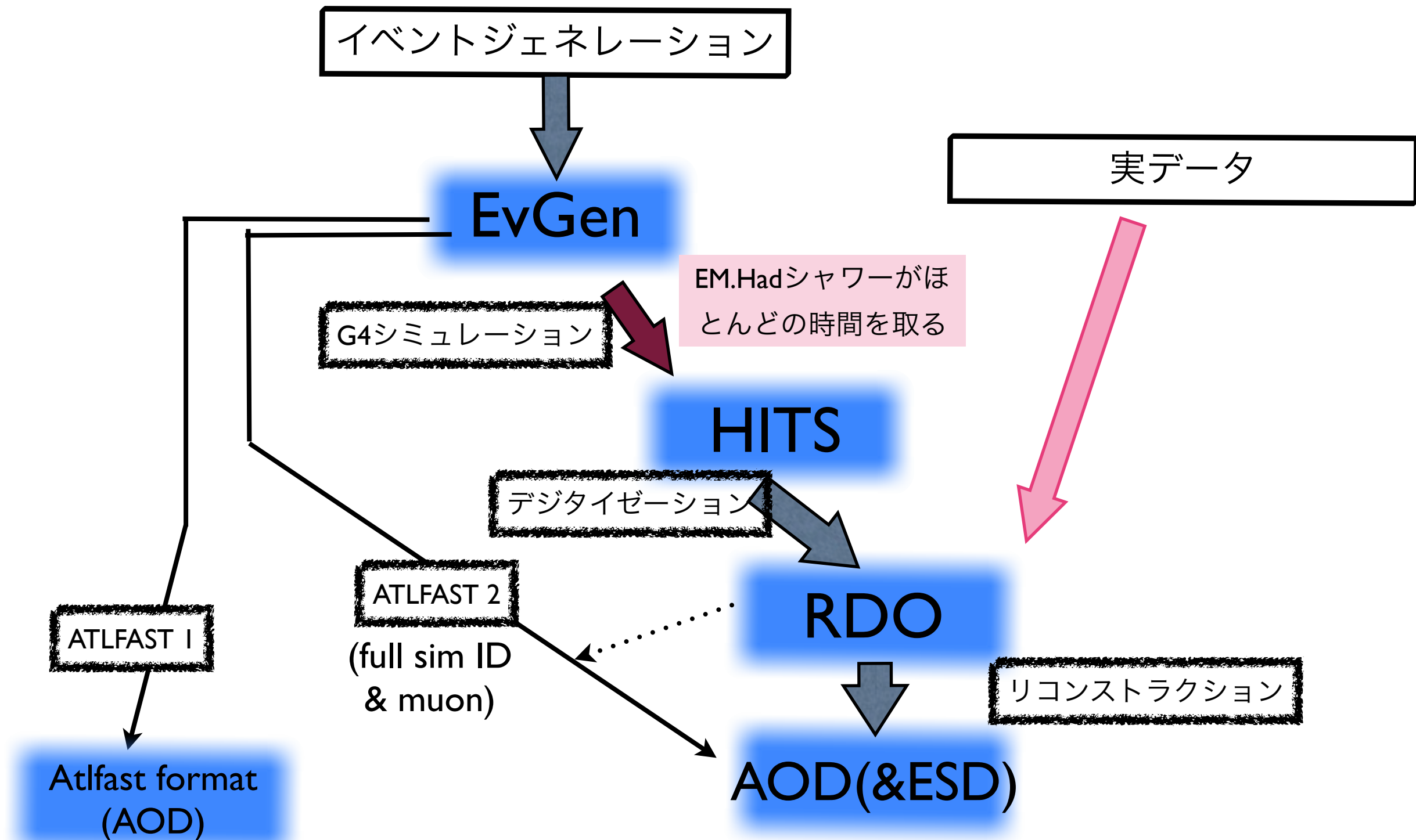
内容

お断り：ここで解説する内容は大まかにはあっているはずですが
一部不正確な部分があるかも知れません & 若干私見を伴っている部分もあります

一般的、概念的な話ではなく、現状でどのようなツール、手法が使えるのか
その長所・短所を比較してみます

- ATLASシミュレーションデータの大まかな流れ
- イベントジェネレーションに関して
 - 標準的なツールと使われ方
- 検出器シミュレーションに関して
 - very fast simulation, 及び fast simulationの現状と性能
 - full simulation の展開 (pile-up, event overlay, long lifetime exotic particles)
- ATLAS MC productionでの使い分けに関して

ATLASシミュレーションデータの大まかな流れ



- AODを解析に使う（または更に下流DPDというコンパクト化したフォーマットを使う）
- G4シミュレーションの速度が一番のボトルネック

MCイベント生成

イベントジェネレーションを扱う上での厄介な点

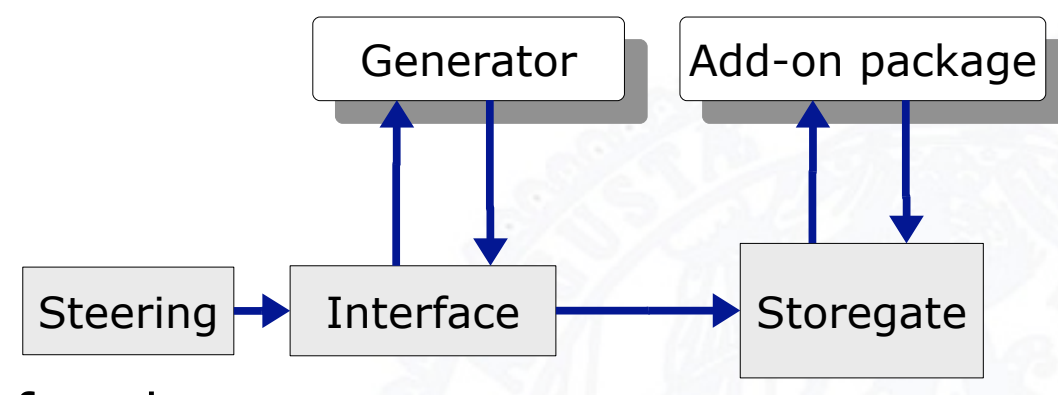
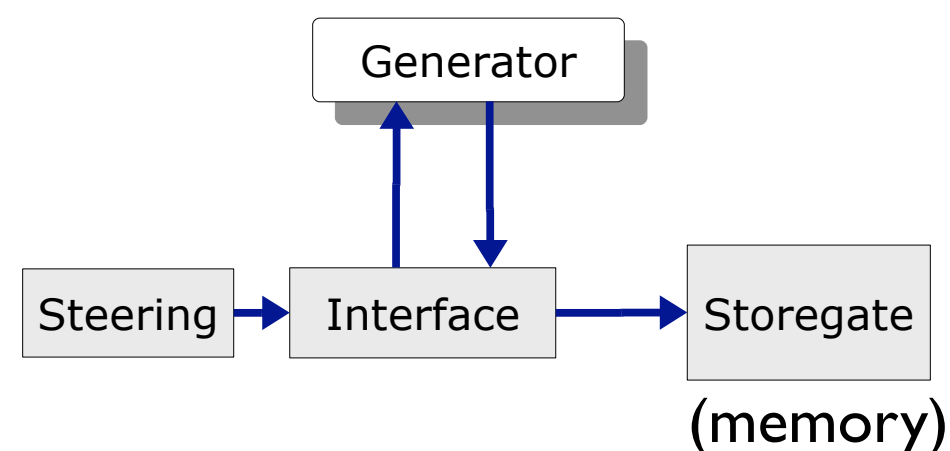
- マーケットに出回っているイベントジェネレーター（モンテカルロ）は多種多様、作者も異なりアウトプットフォーマットも違う。イベントヒストリーの扱いも異なる。これらをどうやって統一的に扱うか。
- アトラスのシミュレーションフレームワークにどう突っ込むか
- モンテカルロ作者が公開しているコード（オフィシャルコード）、Versionとの整合性は
- 次から次に出てくる新しいモンテカルロにどう対処するか
- 技術的な諸問題（i.e. 使われているEvent generatorは主にfortran code, ATLASフレームワークはC++）

これらの問題をATLASでは以下を用いて対処しています。

- Interface Package (ATLASフレームワーク内でHepMCイベントフォーマットで統一的に扱う)
- Genser (MC作者のコードをLHC実験内で共通に管理)
- Les Houches Accord format (MC作者間の共通イベントフォーマット)

Generator Interface in Athena (その 1)

- Parton Shower, Fragmentationまで含むフル Generatorの場合 (all in one type)
- dedicateしたinterfaceを用意して対処する (event historyなども取り扱いが異なるので)
 - Pythia, Herwig (fortran)
 - Sherpa, Pythia8, Herwig++(c++)
 - etc
- 更にAdd-OnタイプのToolは共通のインターフェースで対応する
 - Tauola (tau崩壊のpackage)
 - Photos (QED radiation)
 - EvtGen (B hadron decay channels)
 - 共通の後付けToolを別のGeneratorに付加出来る

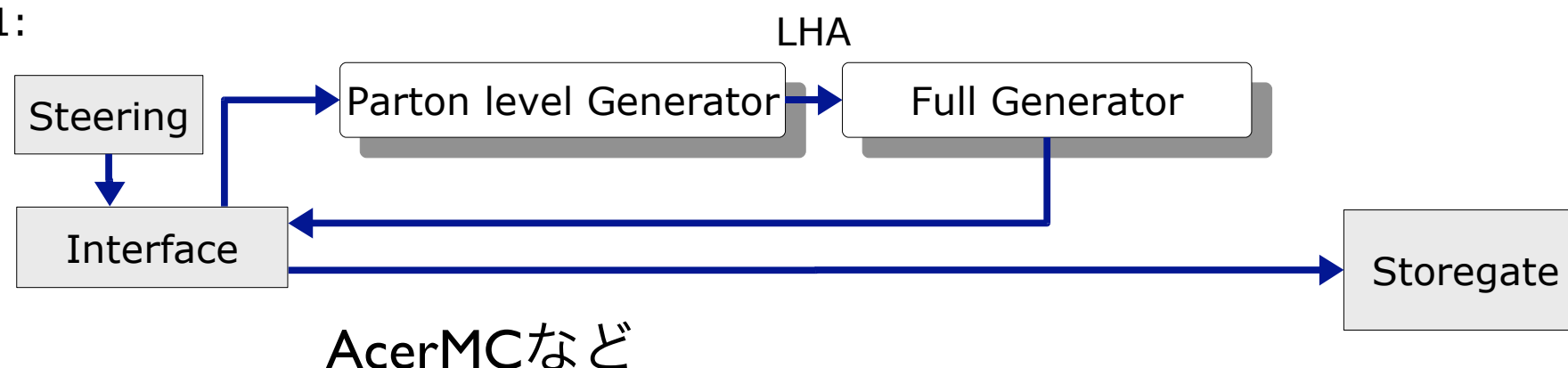


これはATLASでの主なモード 1

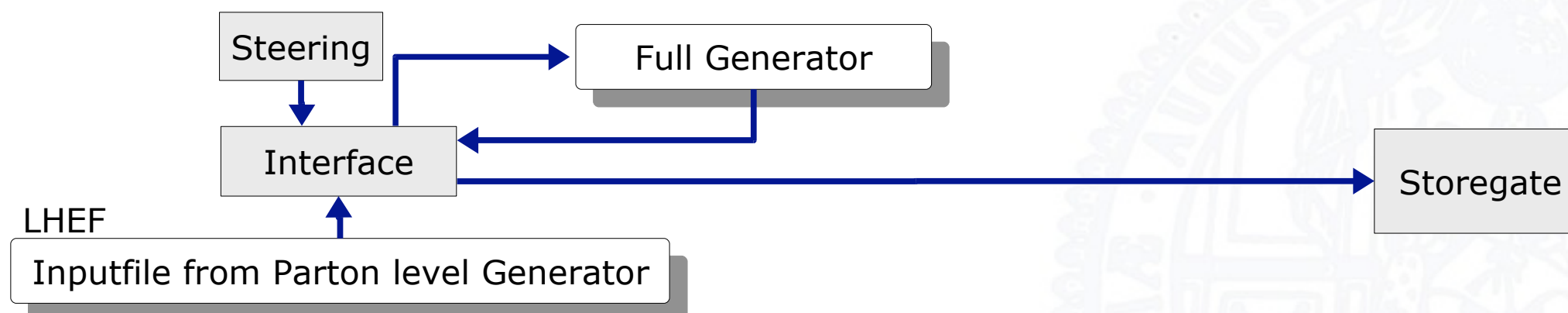
Generator Interface in Athena (その2)

- Matrix Element計算タイプのPartonレベルGenerator
 - AcerMC, AlpGen, Mc@Nlo, MadGraph, Grappa, etc
 - (1) Athenaの中で閉じてParton Shower Generatorにつなげる場合
 - (2) input fileとして読み込んで作る場合 (専用のInterfaceを持つ)

Possibility 1:



Possibility 2:

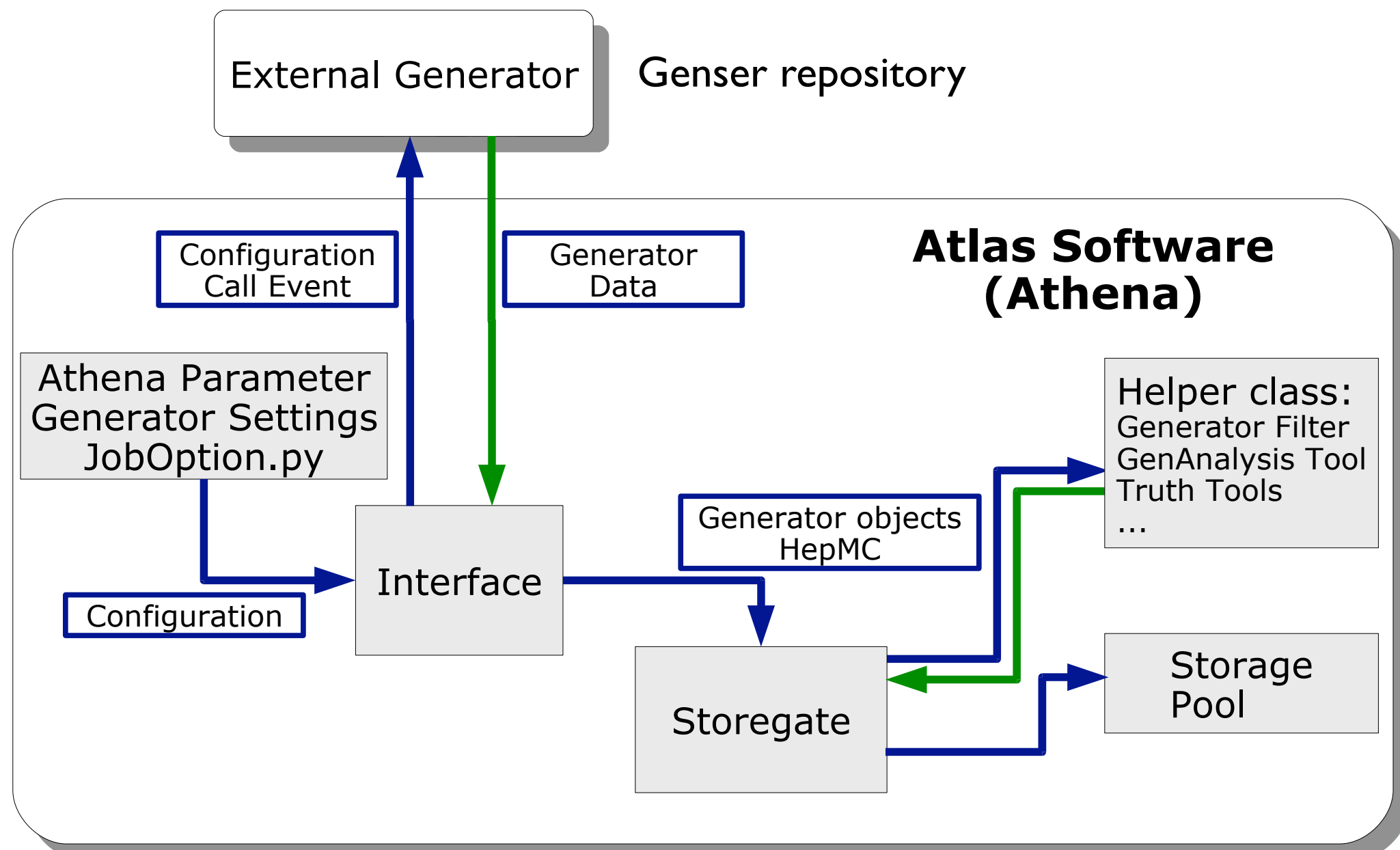


AlpGen, MC@NLO, MadGraph, Grappaなどinputファイルを予め用意しておく

(2) はATLASでの主なやり方2

GENSER (Generator Service Project)

- ATLASはGeneratorのlibraryを自分で持たずGenserというLHC実験共通のRepositoryを用いている
- これによりVersion管理、codeのofficialさ (Author's official)を保証する
- Configurationはユーザーが行い、interfaceを通して外部Generatorを操作



Les Houches Accord フォーマットによるインターフェース

- 新しく誕生するGeneratorがそれぞれ勝手なformatでpartonを吐き出すと対処しきれない→MC authorsの間で共通formatを作った
 - まずLHA Common Block (Fortran HEPRUP, HEPUP)
 - memory上、良く定義された標準フォーマット
 - Pythia/ Herwigのuser processで使われる
- 次にファイルへの書き出しフォーマット
 - LHEF (標準LHフォーマット J.Alwall et al. hep-ph/0609017)
 - LHA common blockをXMLで吐き出す
 - 明確な定義とformat -- 標準化成功！
 - ATLASでもこれに特化したInterfaceがある Lhef_i (どれもこれを使えばよい)

```
<LesHouchesEvents version="1.0">
<header>
  <generator>
    <name version="1.92" date="Mar 12 2008">WHIZARD</name>
    <process_id>qqww</process_id>
  </generator>
</header>
<init>
  2212 2212 7000.0 7000.0 -1 -1 -1 -1 2 1
  0.808449111772444 0.299917935570636E-002 1.000000 1
</init>
<event>
  6 1 1.0 91.188 -1.0 -1.0
  1 -1 0 0 501 0 0.000 0.000 1169.0 1169.01 0.000 0.00 9.00
  2 -1 0 0 502 0 0.000 0.000 -313.94 313.941 0.000 0.00 9.00
  2 1 1 2 501 0 -53.06 89.97 951.60 957.324 0.000 0.00 -1.00
  1 1 1 2 502 0 -15.71 -13.55 -49.774 53.9296 0.000 0.00 -1.00
  24 1 1 2 0 0 -93.82 -40.78 -195.20 234.603 80.41 0.00 0.00
  -24 1 1 2 0 0 162.6 -35.62 148.43 237.094 80.41 0.00 0.00
</event>
```

理論の方でもし新しいモデルを使ったMCをATLASのfull simulationに通したい方がいればこのモードがお勧め

検出器シミュレーション

ATLAS内での方針

- ATLASは準備期間が長かったこともあり、full Geant4 simulationが絶対だと思われています（実イベントのように扱う人さえいる）
- Full simulation全盛（第一優先）の方針はしばらく変わる気配はなし
- しかし大量のBGサンプル生成など、現実的な時間内での生成に無理な場合がある（大量生成に必要なDiskSpaceも問題）。最大のネックはカロリメータ内のシャワー反応にかかる時間

Sample	Z -> e+e-	di-jets	SUSY
Subsystem	Time/Event [%]		
Tracker	3,7	3,8	4,4
Calo - barrel	5,9	7,3	11,0
Calo - endcap	32,0	46,1	33,9
Calo - forward	33,4	14,5	20,6
Calo - hadronic	3,0	6,6	4,7
Muon	5,7	6,4	8,0
Other	9,8	9,0	9,7
Dead	6,5	6,3	7,7
Event	7,7 min	13,6 min	12,8 min

割合

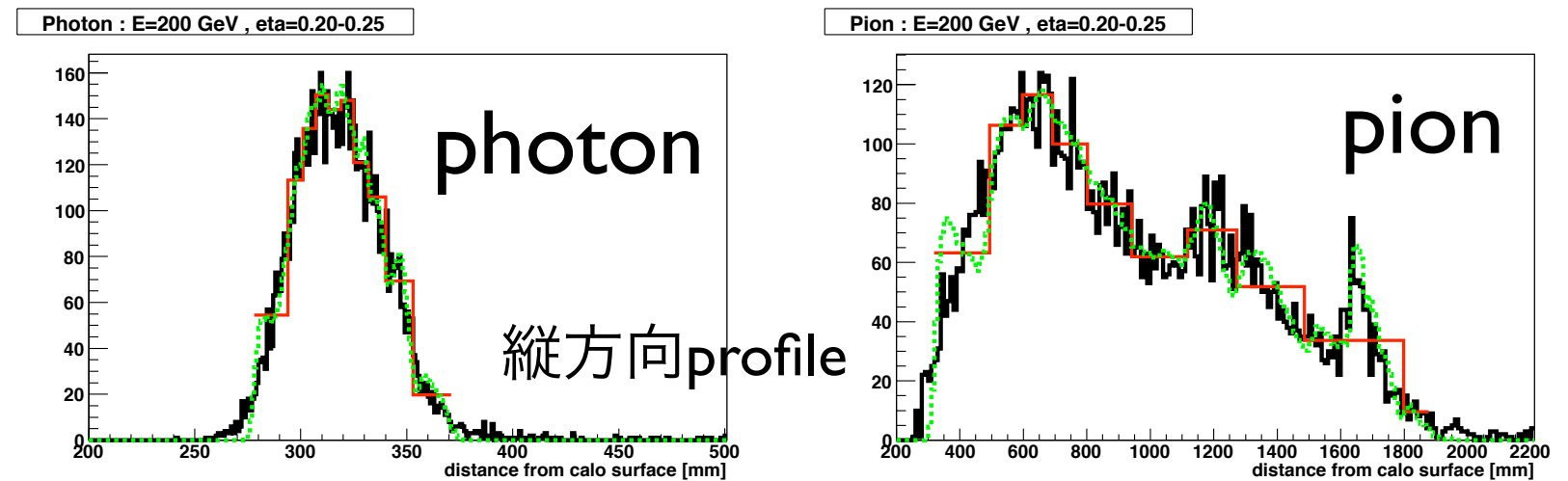
- そこでアプローチとしては
 - ATLFAST-2（G4から独立した独自のCalorimeter simulationを走らせる）
 - FastG4（= Fronzen (parameterized) Shower / G4シミュレーションの枠組みを使う）
 - ATLFAST-I with Parameterization（ATLAS簡易シミュレータ）

ATLFAST-2

- 現在のATLFAST-2 = Full Simulation (tracking+muon)+Fast Caro Simulation (Fast Trackingもdevelopment中)

- Output(AOD)はfull simulationと同じサイズ（中間ファイルは若干節約出来る）

- Calorimeter内でのシャワーシェープをパラメーター化する（右図黒：full sim 赤：モデル）

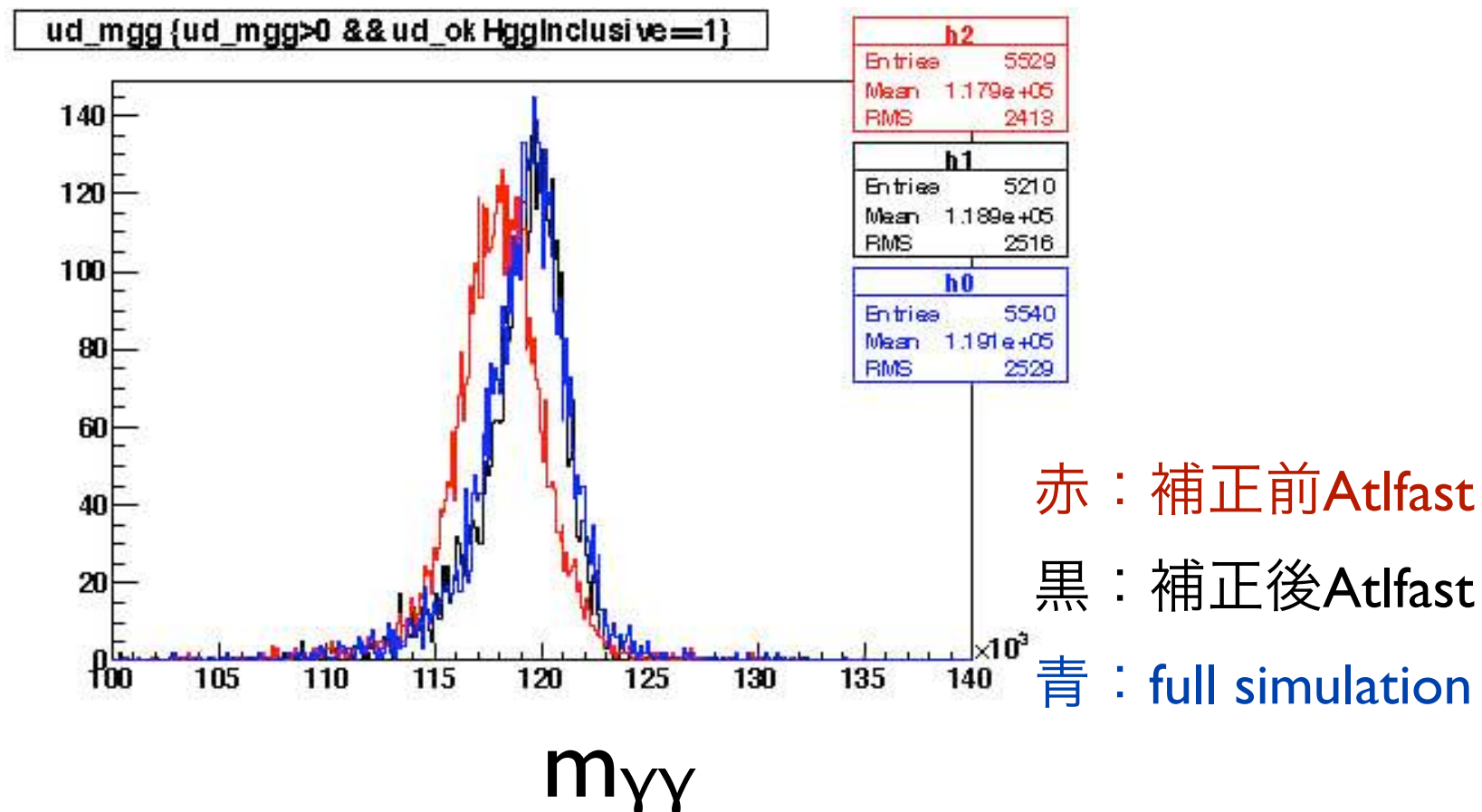


- 第一原理からスタートしているがFull Simulationとは完全には合わない→AODtoAOD correctionと言うparameterizationをする（これはいずれにせよSimulation⇔実データの比較から将来必要になる技術）
- 現在あるAODtoAOD correction (planも含めて) 基本的にobservable variableに対してかける
 - Electrons -- efficiency, E scale, resolution, and fake rate
 - Taus -- shift tau likelihood
 - Muons-- momentum resolution (by FastCalo in calorimeter)
 - Jet -- efficiency, scale & resolutionかなりの部分を強制的に変更（合わない可能性有）

ATLFAST-2 その2

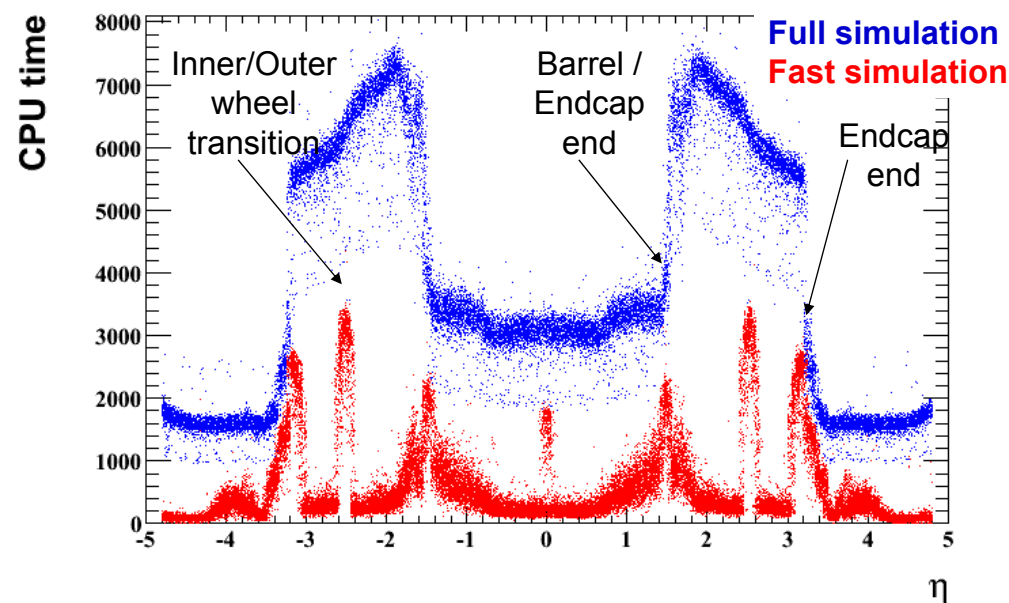
- 前回のRelease(v13)で大掛かりなvalidationが進められた (S.Asai et al.)
- AODtoAODをかけた後でも幾つか課題は残っている (full simとのずれ)
 - Electron ID efficiency が数%低い (同様にConverted Photon)
 - e, mu のIsolation cone energy が低い (平均)
 - Jet energy scale (tau jetも含めて)が高い、high pT (>200GeV) でresolutionがよくなる
 - **Rel.14** では改善されていると期待され、最近**Next round validation**が始まった

$$H \rightarrow \gamma\gamma$$

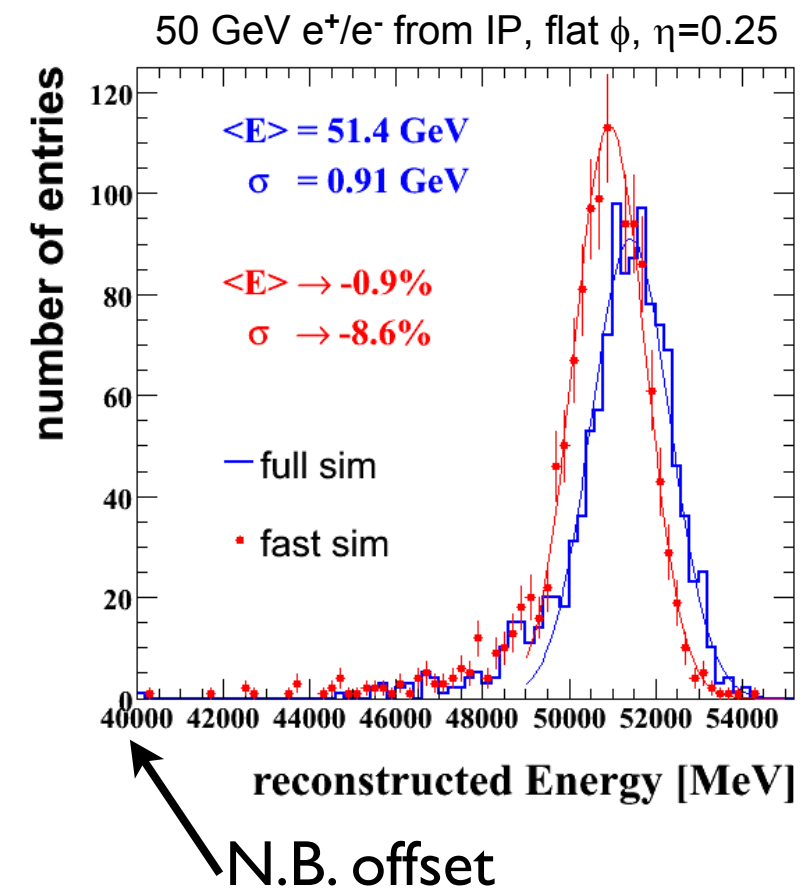


FastG4 (Frozen shower)

- カロリメータ内シャワーにおいてcut-off energy (=1 GeV)以下のElectronはG4で一々追わずに、既成のshower libraryで置き換える(discreteなEとetaでlibraryを用意しておき、内挿して使う)
- もっと低いenergyのもの(<10MeV)はその場でdepositさせる (one spot model)



- physics eventのoverall improvement 実はいざい〜2
- 他に類似のStudyで($E > 1$ GeVに適用) fast showerというshowerの縦横のprofileを関数として扱うものもある (speedの改善は)



ATLFAST-I with Parameterization (concept)

- ATLFAST-IはMCレベルの粒子を基本にして検出器のGeometry、responseを簡素なモデル化したシミュレーションで非常に高速
 - 簡単なテスト等（イベントトポロジーの確認、分布の形）にはとても有効
- しかし簡素化の為、
- 多くの点でFull simulationに合わない（lepton, jetなどのefficiency, energy scale, fake rate等、crack regionでの振る舞い、特にMETなどのテイルの形など）
 - また検出器からの詳細な情報を持たないので検出器のPerformanceが直接効くようなもののスタディーには使えない（b-taggingやtau IDなど）
- 一つ目の問題は、解析がイベントオブジェクトのMomentum, METなどだけで閉じるときにはパラメーター化によって解決することが出来る（例：SUSY解析のBG評価など）



ATLAS NOTE

ATL-PHYS-INT-2009-006

AtlfastCorrectors: Parametrizations of jet transfer function and lepton identifications for enhancement of ATLFAST I results

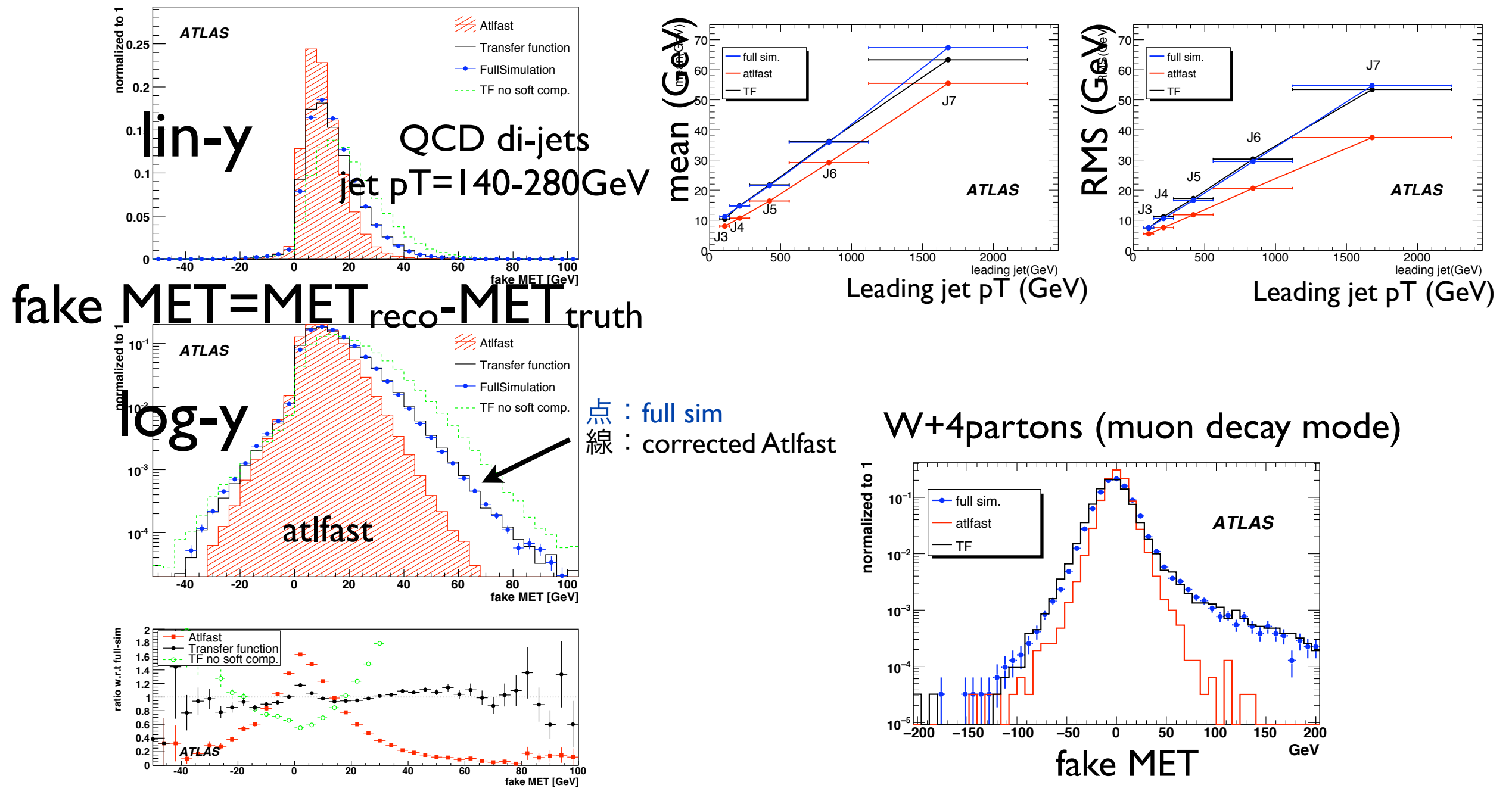
Shoji ASAI¹, Ambreesh K. GUPTA², Naoko KANAYA³,
Yousuke KATAOKA³, Osamu JINNOUCHI⁴

そこでAtlfastCorrectorsという補正パッケージを用意した

- CalorimeterでのJet Responseをjet eta, Eでマッピング
responseはDouble gaussianによってテイル成分まで考慮
- lepton (e, mu)のefficiency, fake rate, fake時のenergy shift等を考慮
inefficient, fakeした時のobject間の差し引き（例：IDされなかったelectronはjetとして残る）
- これらの効果を考慮してMissingEtの再計算

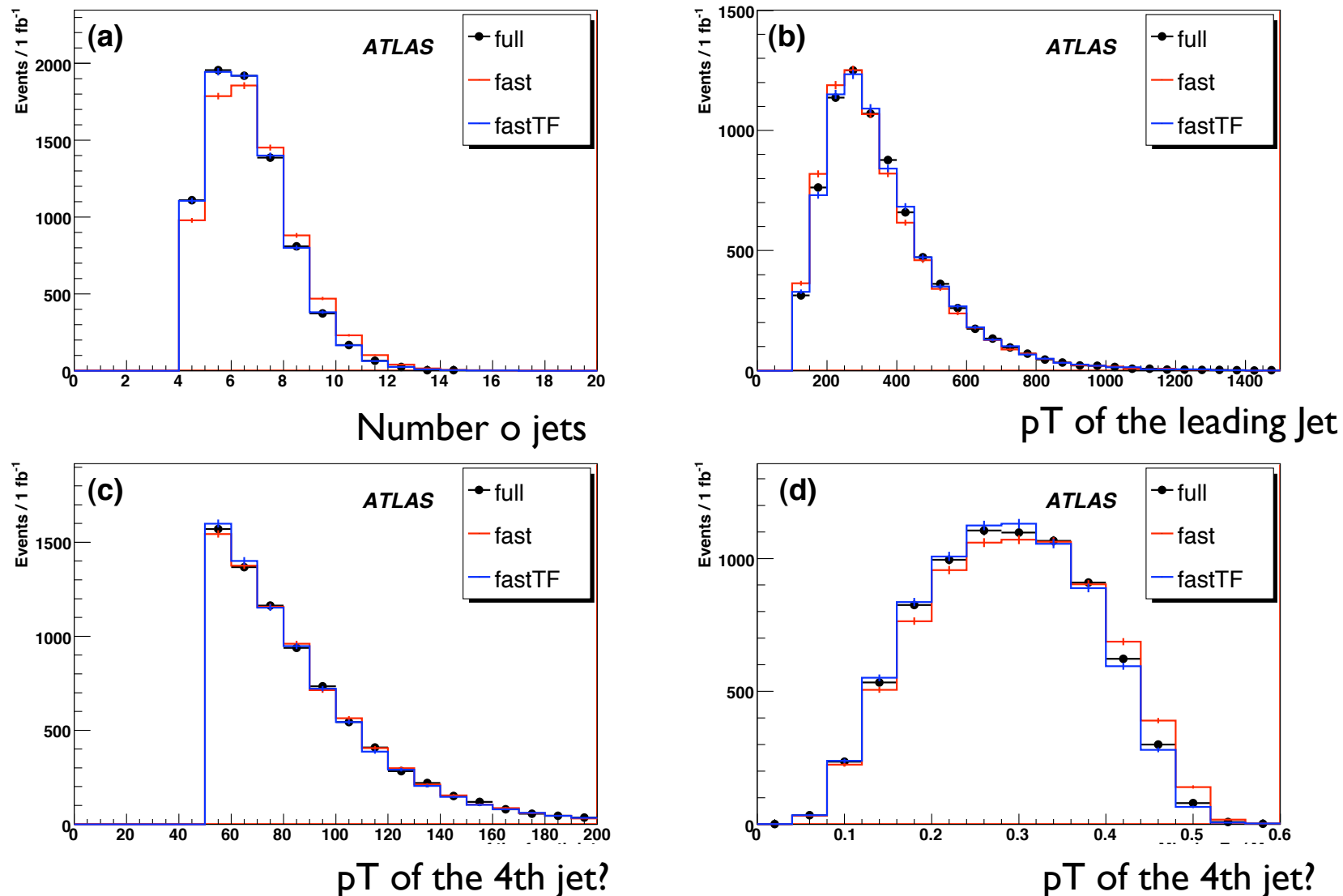
→full simulationにtuneしてperformanceを再現

ATLFAST-I with Parameterization (Performance)



- QCD BGのテイル部分まで良く再現している
- muonのefficiencyの効果も考慮しているおかげでfull simulation MET独特の効果なども再現

ATLFAST-I with Parameterization (Performanceその2)



赤ヒスト : Atlfast default
青ヒスト : Atlfast 補正後
黒点 : full simulation

- シグナルサンプルのアクセプタンス等も良く再現している
- SUSYのパラメータスキャン（数千ポイント）など、FastSimulationを利用したい場面で特に有効
- 基本的にfull simとの合いはとてもよい。Low pT (15GeV) 付近は若干ずれが出てくるので適応範囲に注意

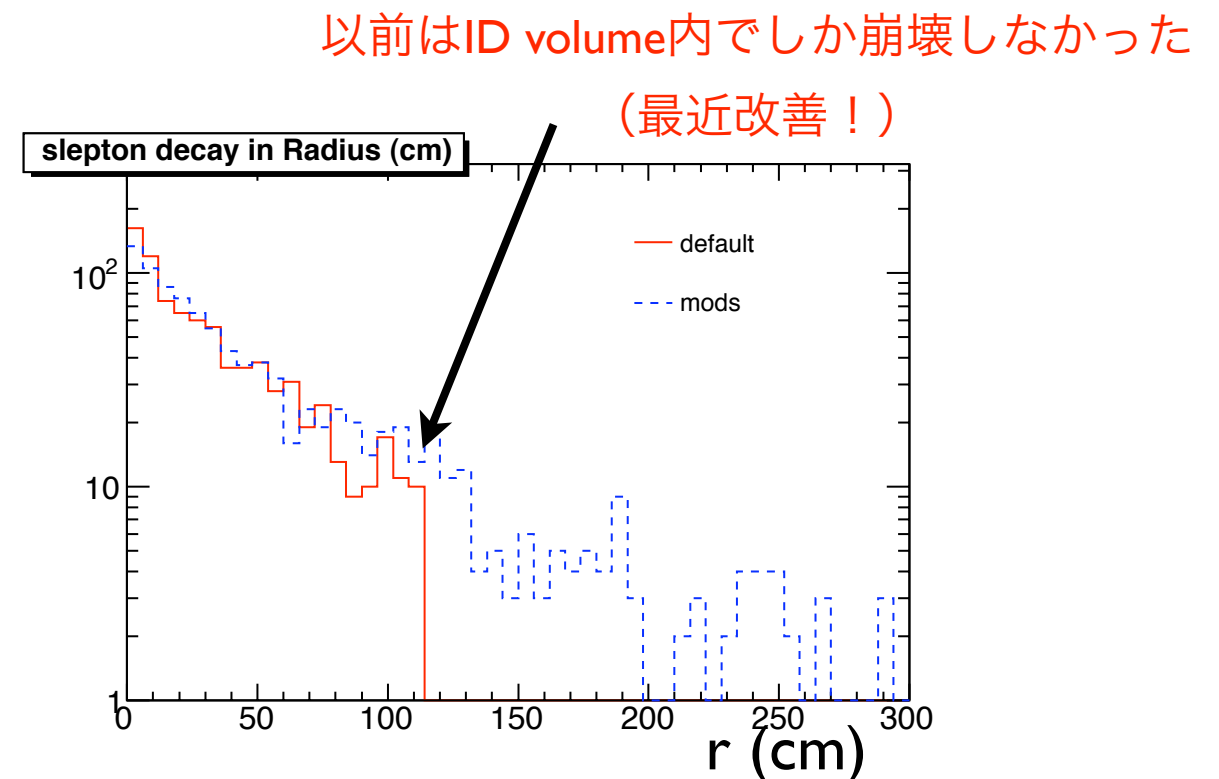
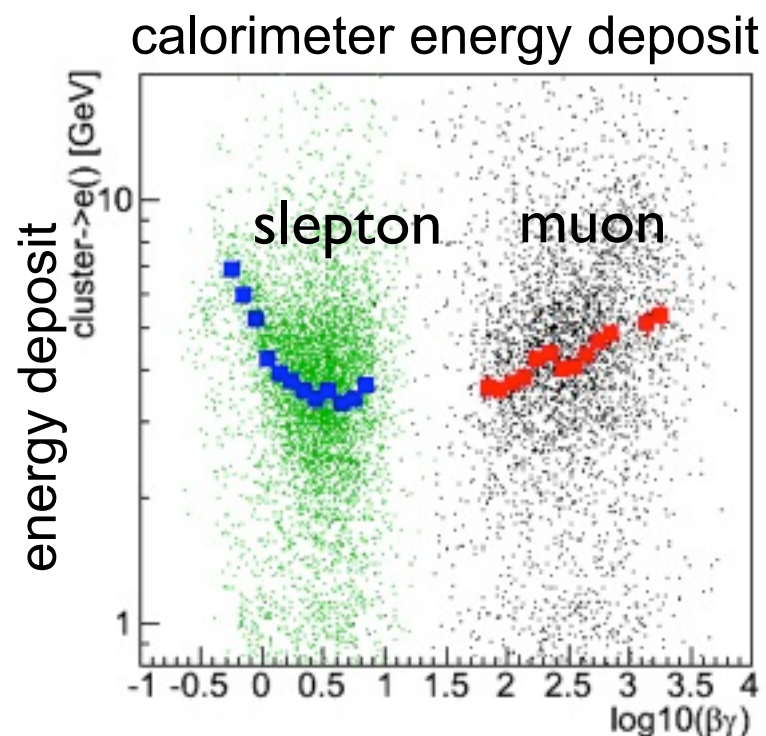
SUSY event

種類	speed sec/event	disk space for AOD	purpose
full simulation	20-30min	200-300kB/event	detailed study for small stat
Atlfast-2	1-5 (with fast tracking) x10 (with G4 tracking)	same as full sim	detailed study for large stat
fastG4	~10-15min	same as full sim	detailed study for large stat
Atlfast-1 with param	0.5-1	20-40kB/event	non detector-performance study with huge stat

- 現時点でofficial productionに組み込まれているのはAtlfast-2のみ
- FastG4は今年あたりから徐々に使われ始める
- Fast-1 + paramはquick testなどの名目では現時点でも開発が進んでいるがofficial sampleに使われる市民権を得るにはもう少し宣伝活動が必要

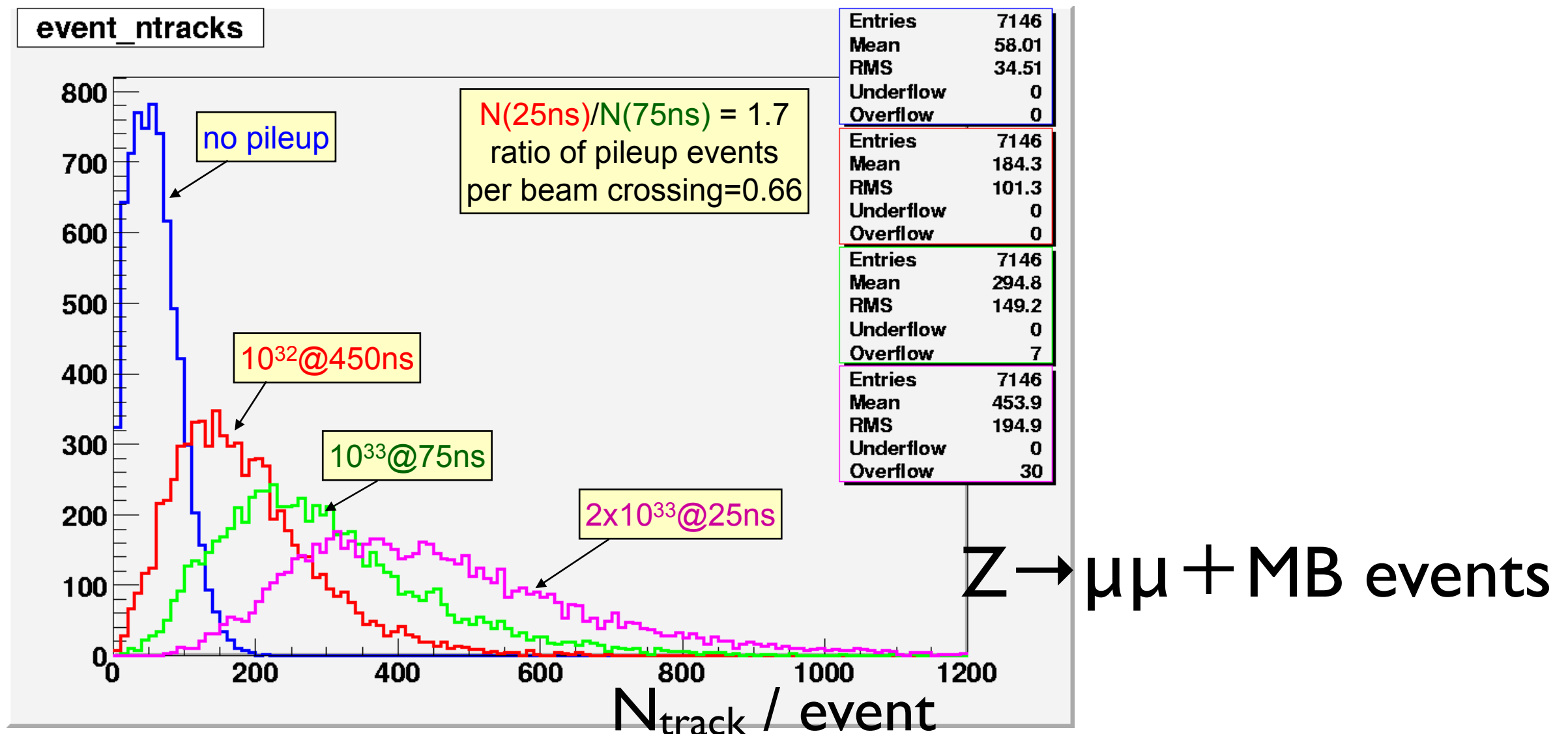
(その他のトピック 1) Long lived exotic particleの扱いについて

- ATLASではctauが1cm以上の粒子はMC generatorレベルでは安定粒子として扱い (Ksなども)、decay及びdecay前に飛んでいる間の物質とのinteractionをGeant4に任せている
- Exoticな安定粒子はATLAS defaultでは相互作用なしですっぽぬけるだけ(neutralinoはこれで良かった)
- mass, chargeを与えてIonization, decayなどをさせるPackage群を作った
 - 登場するparticleのmass, charge, decay mode (Brも定義可)、decay lengthを自由に定義できる
 - mass, decay length等はscriptによってrun timeに変更可
 - ▶ 長生き Slepton (検出器内での崩壊有無両方) → Lepton + Gravitino
 - ▶ 長生き Neutralino → Gamma + Gravitino
 - ▶ 長生き Chargino → Neutralino + Pi (or e+nu) などがある (東君サンプル)
 - ▶ 長生き Gluino, StopがR-hadronを形成しhadronic Interactionをするモードもある



(その他のトピック 2) パイルアップ・イベントオーバーレイについて

- pileup events overlay for more realistic estimation
 - 23(4.6) events overlap for 10^{34} (10^{33}) $\text{cm}^{-2}\text{s}^{-1}$
 - signal+ MB + Cavern + Beam Gas/Halo merging at HIT (simulation) level
(ソースによってタイミングが違う)
 - TRT $\pm 50\text{nsec}$, LAr 500nsec, MDT 1700nsec insensitive を考慮



Simulation: The Next Round MC09

Goals: Why are we doing this anyway?

2008.11.10 physics coordinationより

- Exercise elements that will be needed for 2009 running
 - Particularly those that have not been sufficiently exercised already
 - Pileup: 2009 is not 2008 – we need to be ready for pileup
 - Fast(er) simulation: ATLFast2, Frozen Showers, perhaps FATRAS when ready
 - Geant 4.9

production plan

- どのプロセスを作るかはphysics, C/P groupからのリクエスト次第（時間の関数として変遷する）2008.11時点でのリストをBackupに載せる
- どの手法を使うかはその時のGridのCpu&Disk状況次第（しかし拳動が保証されていないものはどっち道使えない）

まとめ

- Event Generationについて：
 - 幾つかのメジャー古株MCを安定してInterface出来るようになっている
 - LHC共通のGenerator codeを利用(GENSER)しかしAtlas defaultパラメータでsteeringする
 - 新たに次々登場するMCには共通フォーマットを用いて対処している
- Simulationについて：
 - 時間短縮のため、幾つかのfast simulation手法が試されている
 - full simulationも幾つか新しい機能が取り入れられている