Experimental Overview

P. Lenzi - CERN
On behalf of ATLAS and CMS

Outline

- Tools and MC production in ATLAS/CMS
  - Generator code releases
  - Processing in grid environment
  - LHE files and all that
  - Fortran VS C++
- Data/MC comparisons
  - W/Z: a benchmark
  - Some observables showing discrepancies

Disclaimer: Not a comprehensive overview. Rather a few points to raise discussion in view of the work to be done during the long shutdown

An occasion to learn from each other

- This workshop represents a very useful opportunity to discuss with the TH community topics ad problems common to ATLAS and CMS
  - We can discuss and find common solutions to problems in view of the new run at high energy
- In the past two years we have come across some of the same problems and followed different, in some cases complementary, paths
  - To name a few:
    - CMS has concentrated on MadGraph, ATLAS on AlpGen and Sherpa
    - ATLAS already has experience in pythia8
    - CMS has developed a dedicated treatment for heavy flavor matching with MadGraph
    - On both sides studies on matching ME generators with pythia8
- The long shutdown 1 (LS1) ahead of us is a good opportunity to review our tools, get ready with new tools, promote studies, and also improve on our capacity of producing efficiently MC events

The challenge of Monte Carlo production

- Most of our computing resources are busy doing MC most of the time
  - Detector simulation is slow (~1-5 min/event)
  - we have comparable number of simulated events as we have real data events
    - In 8 TeV data both CMS and ATLAS have (or will have by the time of Moriond) 4-5 Billions simulated events
      - CMS for example is expecting order of 4B data events by the end of 2012 run
  - Covering the bulk of the needs of analyses for Summer and Winter conferences is an 8/10 months effort
- Points crucial for success
  - High level of scrutiny and validation of the generator codes
  - Stable/recommended code releases to be used throughout the production
  - Scalability, efficient code for a grid environment

Code validation and release cycle

- When we integrate a new version of a generator we go through a validation process
  - Difficult because process dependent
- MC Production cycle in experiments is slower than development cycle for generators
  - Sometimes difficult for us to keep up
  - A release marked as “production release” from the authors would help
    - The development can proceed in parallel
  - “svn co” the head of the repository is not suitable for a long term MC production
    - The head is by definition the place where the development happens, cannot be regarded as stable
    - Difficult to track versions months later

Code structure and grid processing

- It is becoming increasingly difficult to build and link fortran with recent gcc releases
  - ATLAS and CMS had several problems with common block initialization with gcc462
  - The fortran bit that is there all the time is LHAPDF
    - There are plans for C++ version
      - Probably needs fortran wrappers for the fortran codes
- Chaining several generators
  - e.g. MG+pythia6+tauola, or powheg+pythia8
    - How do we make sure we get a consistent setup?
- Parallel processing
  - So far parallelization on the grid is just running the same job with different seeds on different cores
    - But some multi-core nodes are becoming available
      - A job can use all cores in a physical machine
  - Are there any plans for thread-safe developments in any generator?

LHE production on the grid

- The production of LHE files is difficult because
  - Each generator needs to be run in its own special way
  - Often changes in makefiles and code are needed to implement cuts/use of external libraries etc
  - Often handled by individuals in “heroic” efforts
- CMS has developed a wrapper to run any shell script that produces an LHE file
  - The process is transparently treated by our production system as any other job
    - Still needs tuning of the #events/job
- ATLAS is working on something similar
- In general we need codes to produce LHE files that are as much robust as possible
  - Avoid compilation at runtime
  - Avoid hand made changes, everything should be configurable

ATLAS workhorses

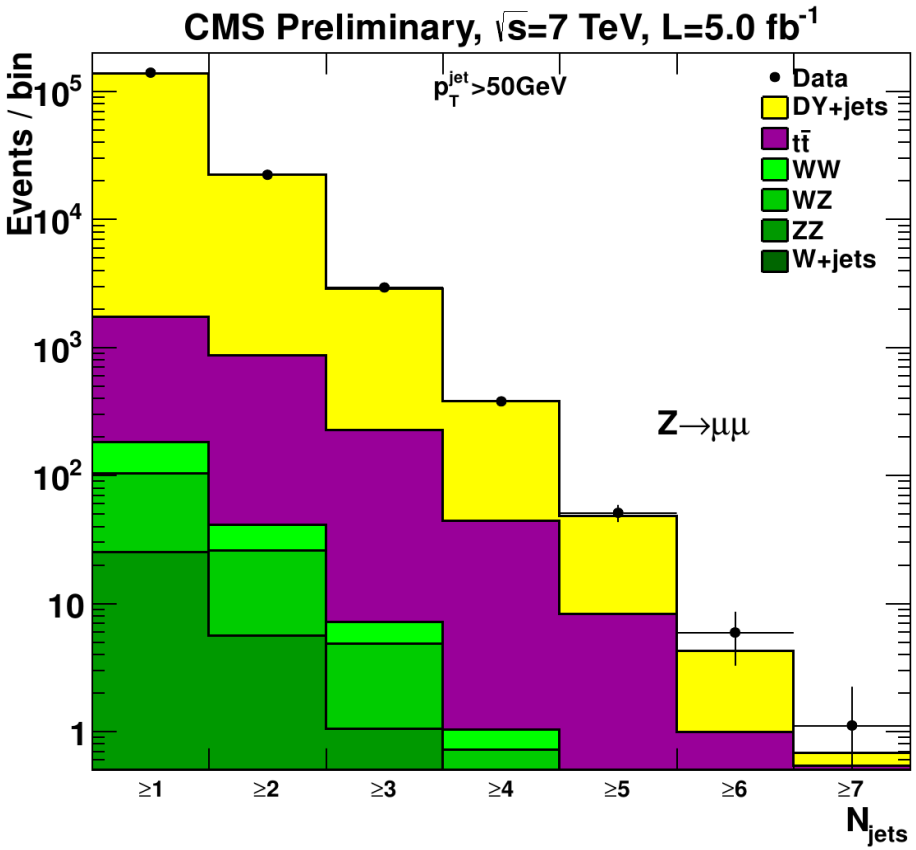
- For multi-jet final states
  - Alpgen with Herwig6 and since recently with pythia6
  - Sherpa
- For processes where NLO is available
  - MC@NLO with herwig6
  - POWHEG with herwig6 and pythia6,8
- For signals
  - pythia8 with dedicated tune
  - Madgraph

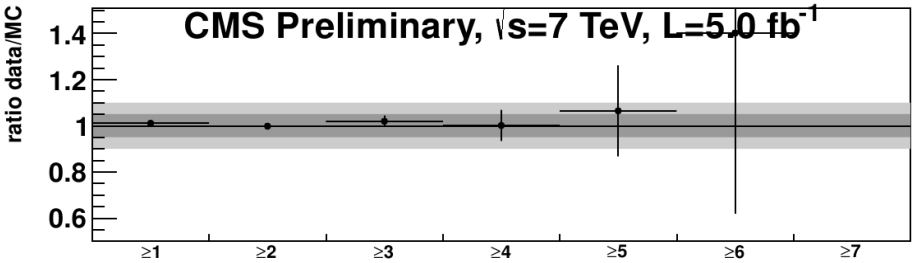
ATLAS: plans for LS1 and after

- Move to modern Herwig++ and Pythia8
- Needs a lot of work on validation
  - Especially for shower + ME generator setups
- Example setups data/MC plots from the theory community would help a lot in this respect
- Develop the possibility of producing LHE files via central production tools
  - The tools themselves need to be easy to run on the grid

CMS workhorses

- Main shower tool
  - Pythia6, tune Z2*
- Main Matrix Element tools
  - Madgraph, Powheg
- Decay tools
  - Tauola (recently switched to tauola++)
  - EvtGen
- Several other tools integrated and used for dedicated studies
  - Herwig6/++, Pythia8
  - MC@NLO
  - Sherpa, Alpgen
  - Dedicated generators for Heavy Ions, Forward physics...



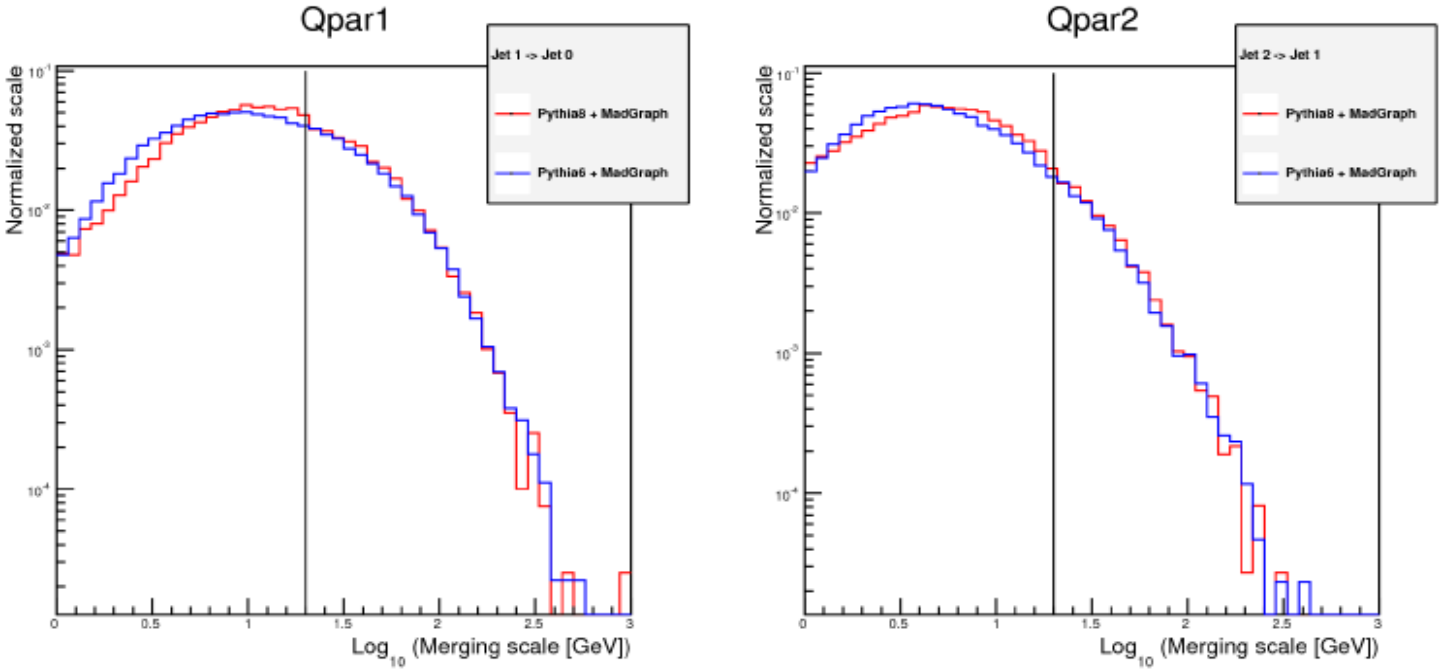


CMS: plans for LS1 and after

- Evaluate and tune pythia8
- Study new generators in more detail
  - “monolithic” tools are somewhat easier to use
    - No LHEs to produce and ship around the world
    - Less “risky” when it comes to possible inconsistencies between the different pieces piped together
  - But we need to retain the spectacular data/MC agreement we have now
- Make more generator level studies without the pressure of data

CMS: plans for LS1 and after

- We are working on MG + Pythia8
  - Work initiated within CMS, Pythia8 authors are also working on something similar
  - CMS version completely based on the existing MG+Pythia6 code, just translate pythia8 event record to something resembling Pythia6 event record before calling the matching routine
- But what is the level of tuning of Pythia8 wrt Pythia6?

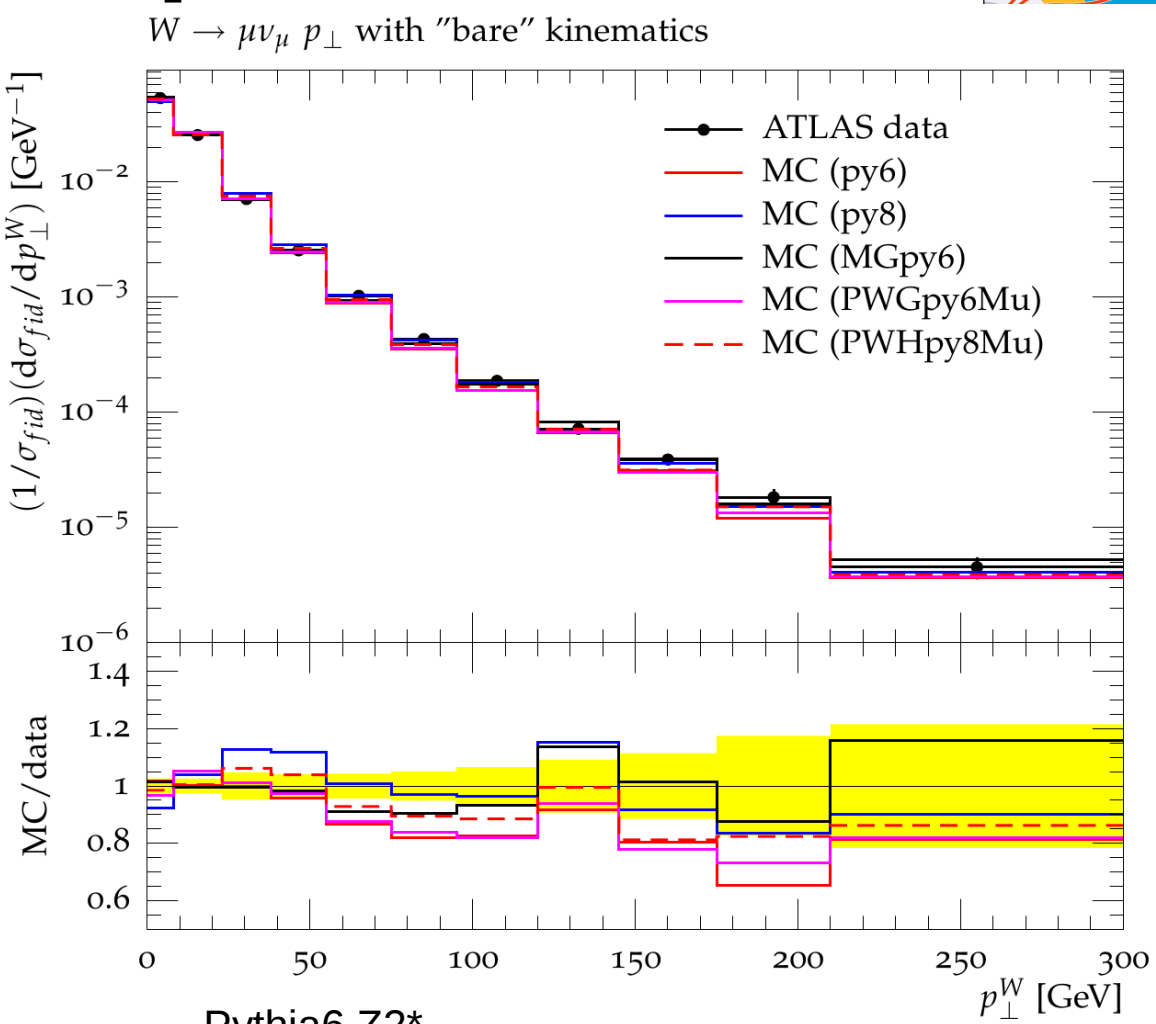


W and Z data/MC

- A brief look at a few observables showing discrepancies
- Not meant to be an extensive overview
- The topic will also be addressed later with dedicated presentations during the workshop

W pt

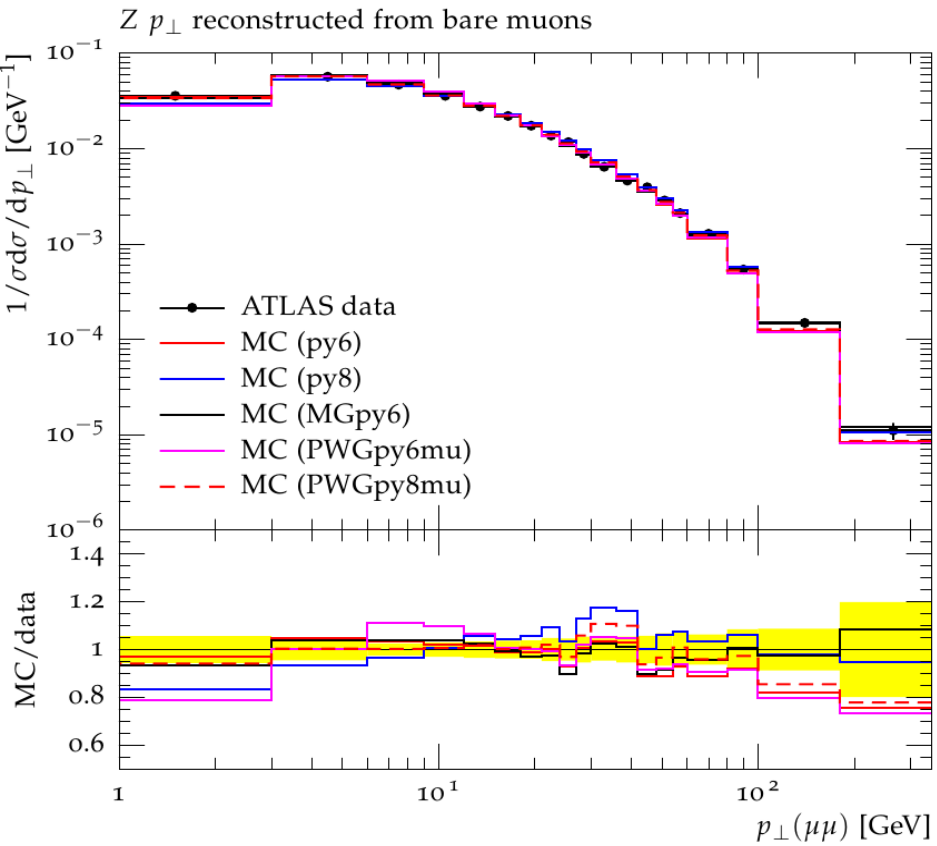
- Pythia6 and MG give the best description
  - indistinguishable below 30 GeV
    - Expected (dominated by 0j ME+shower)
  - MG is above pythia6 at large pt
- POWHEG not so good, especially with py6
  - Not clear why. The Powheg method should be nominally equivalent to pythia6+ ME corrections for this process

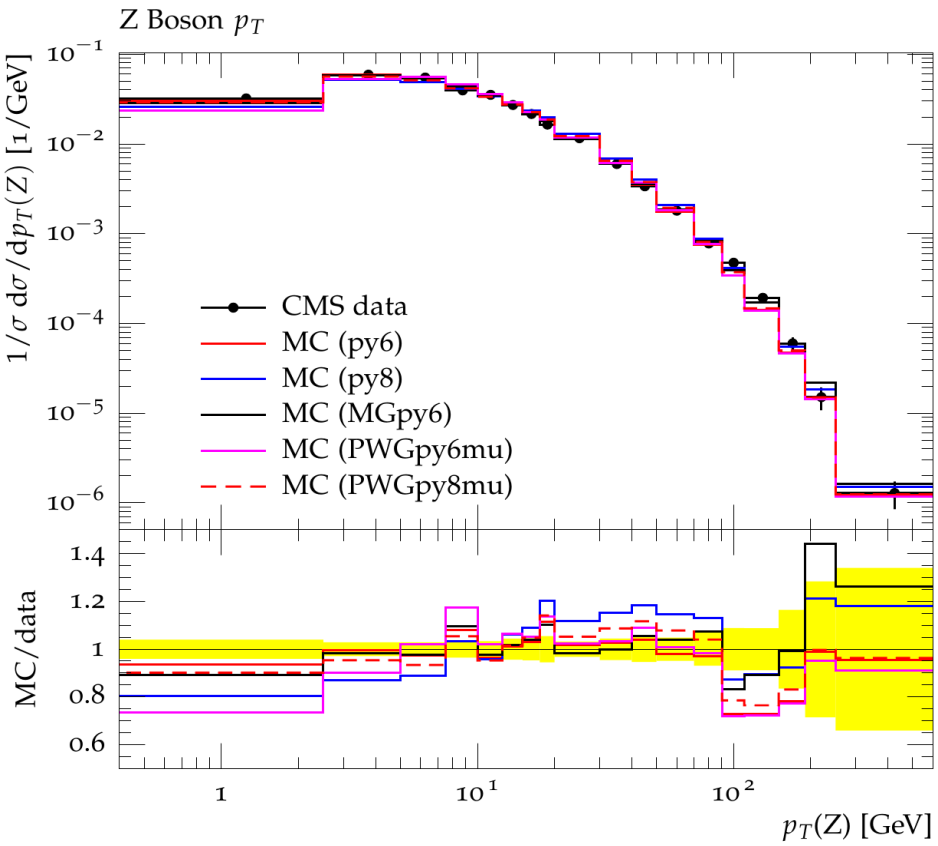


Pythia6 Z2*
Pythia8 4C
MG+pythia6 Z2*
Powheg+Pythia6 Z2*
powheg+pythia8 4C

Z pt

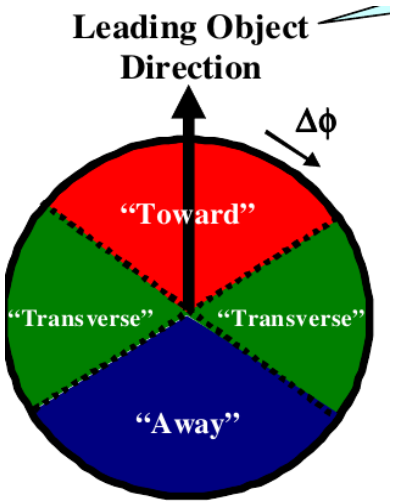
- Similar conclusions for ATLAS and CMS:
  - powheg+pythia6 Z2* and pythia8 4C are quite far from the data
    - Why?
  - MG+Pythia6, pythia6 and powheg+pythia8 similar



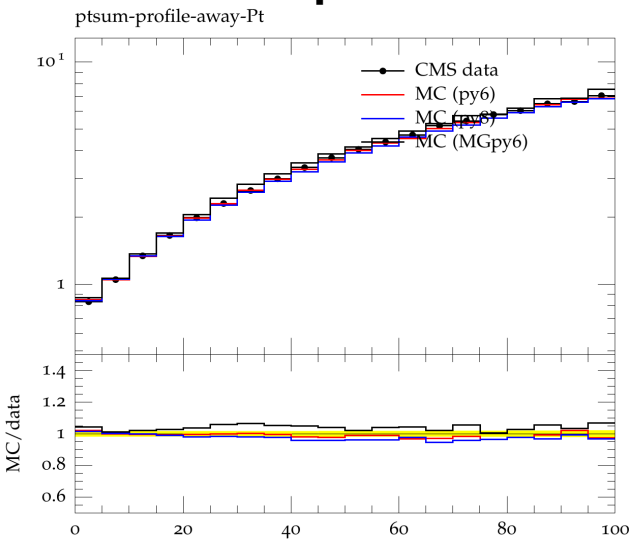


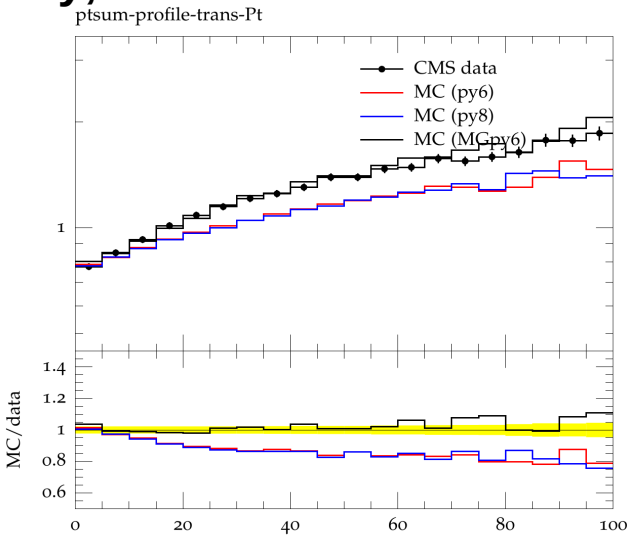
UE in DY

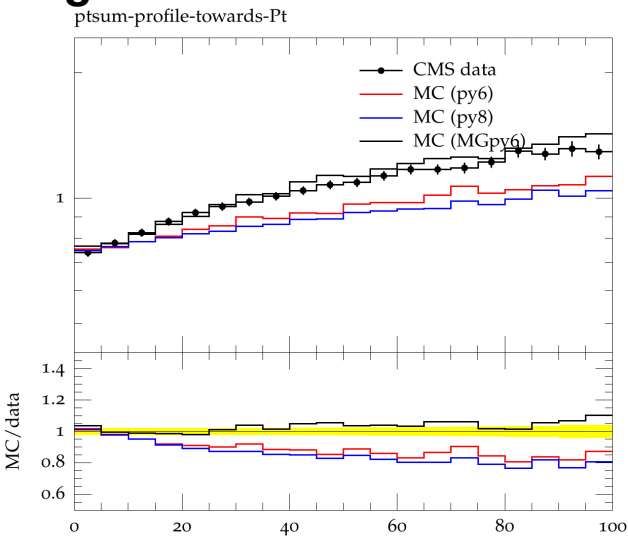
- Classic Rick Field UE observable, using the Z as “leading object”
- We are looking at the hadronic recoil
  - Similar to what you do with Zpt
  - **But this time differentially in terms of how the phase space for the recoil is filled**
- Pythia is able to describe only the away region
  - Z+1jet topology
- **Already at 20 GeV Zpt you need ME+PS if you want to describe the “transverse” and “toward” regions**
  - **Might be relevant for MET**



Σ track pT vs Z Pt in away, transverse and towards regions

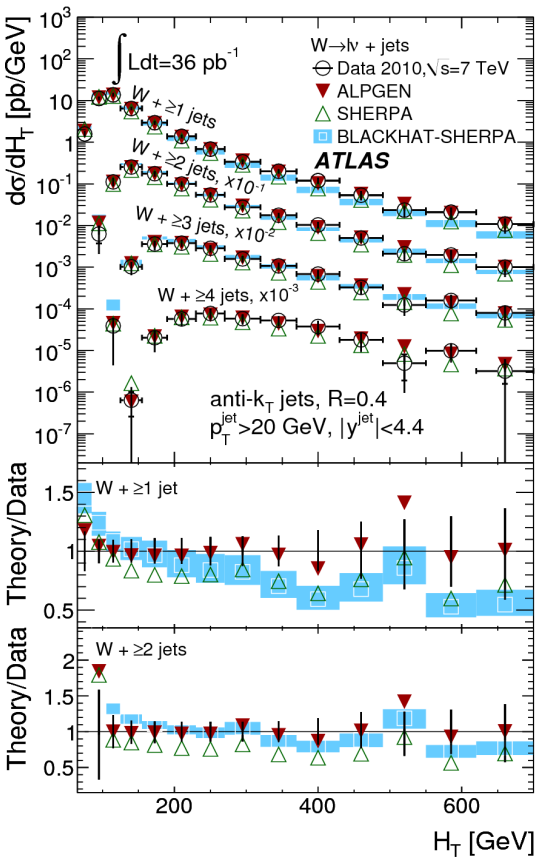


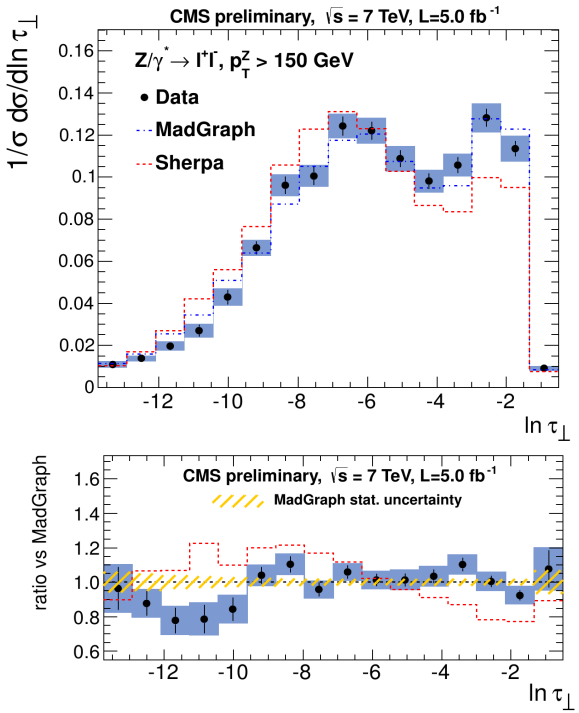


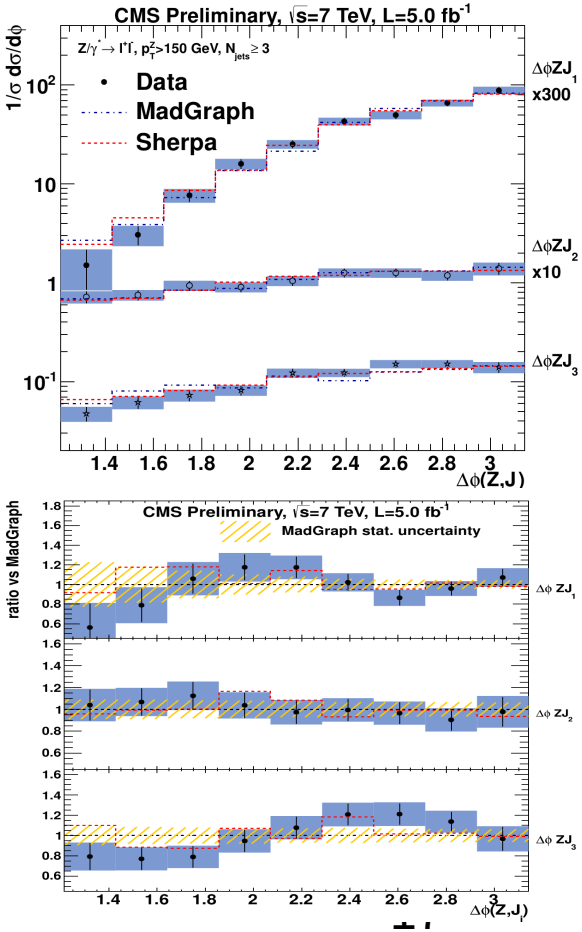


Studies in extreme regions

- Both ATLAS and CMS have looked into more extreme regions ond more complicated observables, relevant for searches
  - ATLAS measurement of Ht in W/Z+jets
  - CMS angular correlations and event shapes in Z+jets







Conclusion

- The LS1 is ahead of us and it represents a good opportunity
  - For ATLAS and CMS to work together on common problems
  - To study more codes and to gain more experience without the pressure of data
  - To work together with theory colleagues to improve the scalability of the codes