

Overview of work done



Carlos Solans
TileCal Valencia meeting
13th July 2009





- Online
 - Commissioning is over, Tile calorimeter in operation mode
 - Lost access to detector
 - Bad channel list status was cleared
 - Frequent calibration runs: Charge injection, optimal filtering validation
 - High rate tests revealed some issues with DSP configuration.
 - Updates to DSP code: QF histogramming, CRC memory checking
 - Updates to RCD code: Automatic stopless recovery
- Analysis
 - Obtaining upper exclusion limit values and coverage results

Reconstruction status



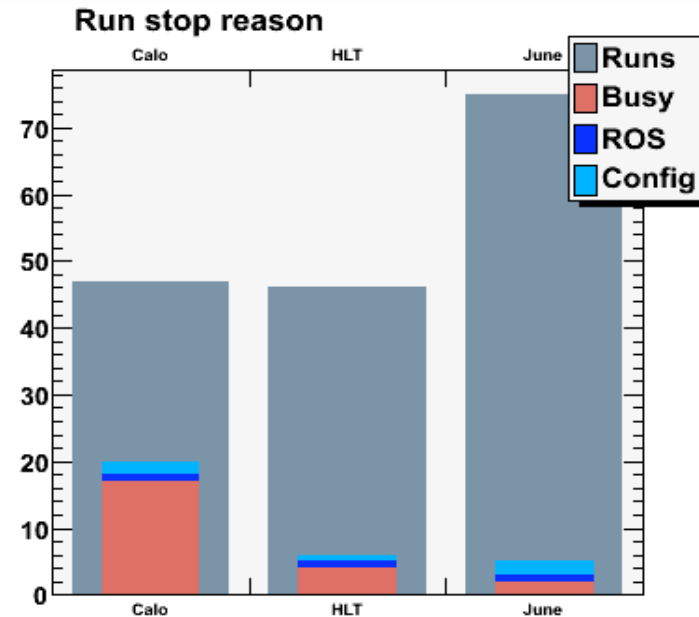
Reconstruction performance can be examined few seconds after the start of the run.



DAQ operation statistics



	Calo Slice week	HLT Slice week	June cosmics
# runs	47	46	75
Tile problem	21	6	5
Tile busy	18	4	2
ROS	1	1	1
Config	2	1	2



- We select all “End of run” entries from the ATLAS ELOG where
 - The affected system contains Tile
 - The online partition is ATLAS
- We classify the Stop reason from Tile: “Tile is busy”, ROS related or mis-configuration problem
- Conclusions:
 - We experienced “Busy” problems during calo slice week.
 - Problem was understood and addressed

High rate tests



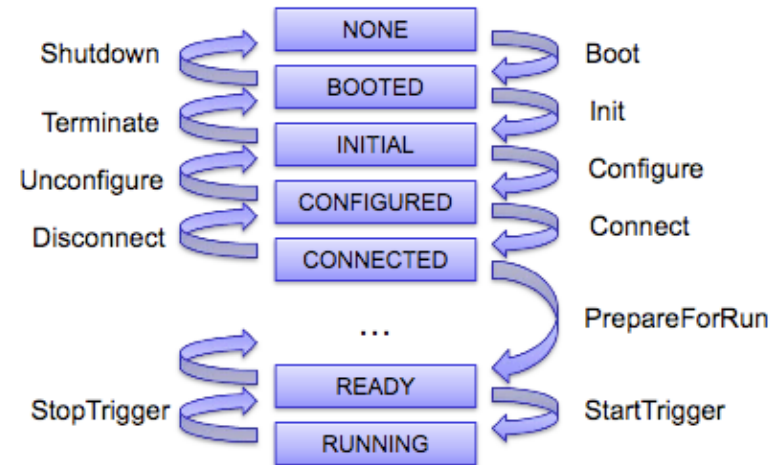
Date	L1 in (kHz)	L1 eff (kHz)	Most busy	Sub-systems	Comments
04/06/2009	150	45	FCAL/HCAL noise level (70%)	Tile+LArg	
04/06/2009	75	45	FCAL/HCAL noise level (40%)	Tile+LArg	
25/05/2009	150	93	L2 (38%)	Tile	All drawers sending data
22/05/2009	150	69	Drawers not sending data (23%)	Tile	
22/05/2009	75	69	Drawers not sending data (5%)	Tile	

- High rate tests were performed after the outcome Calo slice week in April.
 - Dedicated effort was made on the DSP performance and ROD configuration.
 - Using random clock with simple dead time limit (8 triggers in 80us).
- Performance of Tile stand-alone depends on the super-drawers.
 - We get into a busy situation when drawers misbehave: More data than TTC events, lazy drawers.
- Performance of Tile+LArg affected by FCAL/HCAL processing time.
- Requested more frequent testing in ATLAS conditions.

Tile busy problem



- Usually happens in running state, before the first event has been built
 - We blamed the DSP not booting properly...
- Possible source of problems where carefully isolated during ROD configuration, involving:
 - Configure (DSP booting)
 - Prepare for run (Constants loading)
- It was found that a publish Statistics was issued in the middle of the prepare for run in the ATLAS partition
 - Because the command is issued by a timer after the configure transition and the CTP took longer than the timer period to configure.
 - The ROD RCD module plugin was modified not to react on this command until the prepare for run had not finished.
- We suspect that some of the latest “Tile Busy” reasons at the end of the run have nothing to do with this issue anymore.
 - This “busy” problem could not be recovered with automatic stopless recovery because it was a ROD mis-configuration.



<http://indico.cern.ch/contributionDisplay.py?contribId=0&materialId=slides&confId=60006>

DSP constants checking



- Loading of runtime constants into DSPs at prepare for run is a crucial step for running.
 - Optimal filtering
 - Channel status
 - Muld thresholds
- What can we do to check them?
 - When the constants are extracted from COOL and formatted to be sent to DSP we compute their CRC
 - Once they are loaded to the DSP, we ask the DSP to compute the same CRC check on the values loaded.
 - In case of mismatch, fatal error will be issued and run has to be restarted.
- Status:
 - CRC code developed by Vakhtang for CPP and ASM.
 - DSP CRC computation of a region of memory region implemented
 - CPP (ROD RCD) implementation to be done

```
=====
unsigned int cpp_CRC32(char *p, int len) {
    unsigned int *buf = (unsigned int *)p;
    unsigned int crc;
    int i;

    crc = 0xFFFFFFFF;

    while (len-- > 0) {
        crc = crc ^ *buf++;
        for (i = 0; i < 32; i++)
            crc = crc & 1 ? (crc >> 1) ^ 0xEDB88320 : crc >> 1;
    }

    return crc ^ 0xFFFFFFFF;
}
=====
```

- We also store the OF constants per super-drawer in a file per run.
 - /det/tile/conditions
- This work has been pending for some time, but other things always come first.
- This error cannot be recovered automatically, though.

Automatic stopless recovery

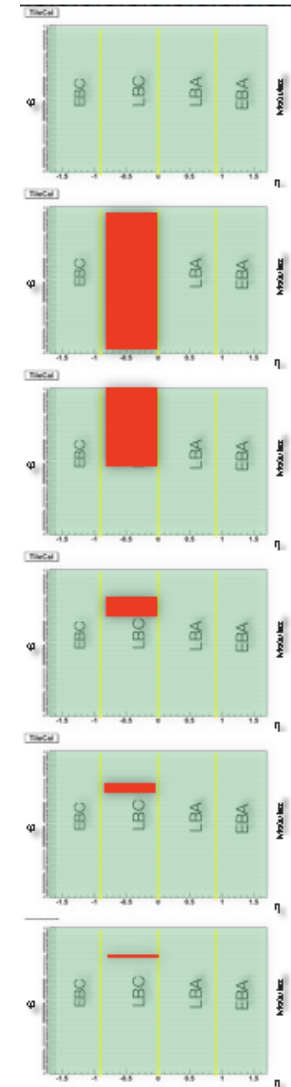


- Each sub-detector should be capable of resuming the run after a busy without stopping the run.
 - The HW causing the busy is disabled and the run is resumed.
- What can go busy?

Affected	# Drawers	Fraction of detector	Note	Reason
ROD crate	64	25.0%	1 Barrel	ECR lost
ROS	32	12.5%	½ Barrel	ROS problem
ROD	8	3.1%	1/8 Barrel	TTC synch lost
ROL	4	1.6%	½ ROD	ROBin problem
DSP	2	0.8%	Phi sector	DSP problem
	1	0.4%		Data problem



We aim for automatic stopless recovery for single ROL or DSP. Anything beyond that should STOP the run.

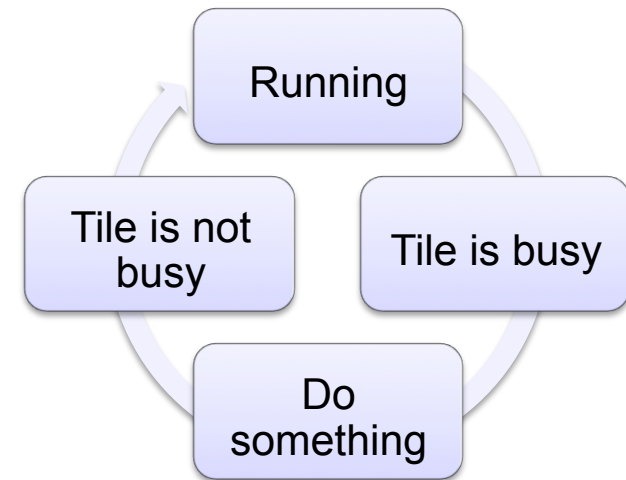


Automatic stopless recovery

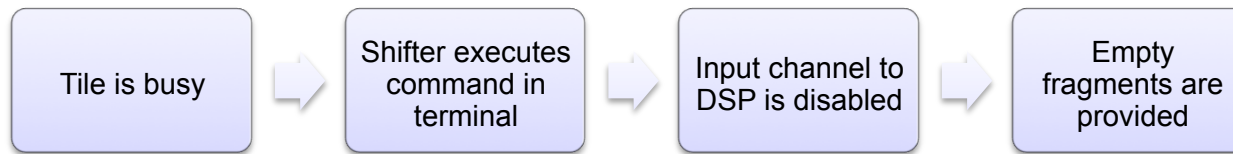


- Implementation

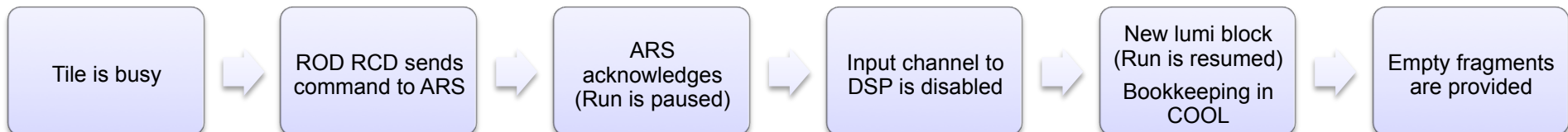
- ROD RCD code can handle disabling input links to the ROD through the command line given HW coordinates.
 - Busy is forced, input channel to DSP is disabled, busy is released.
- VME interrupt mechanism being implemented
 - Disable HW providing busy signal above threshold through VME
 - Requires Automatic Recovery System acknowledgement.
 - Tracing of disabled resourced in COOL

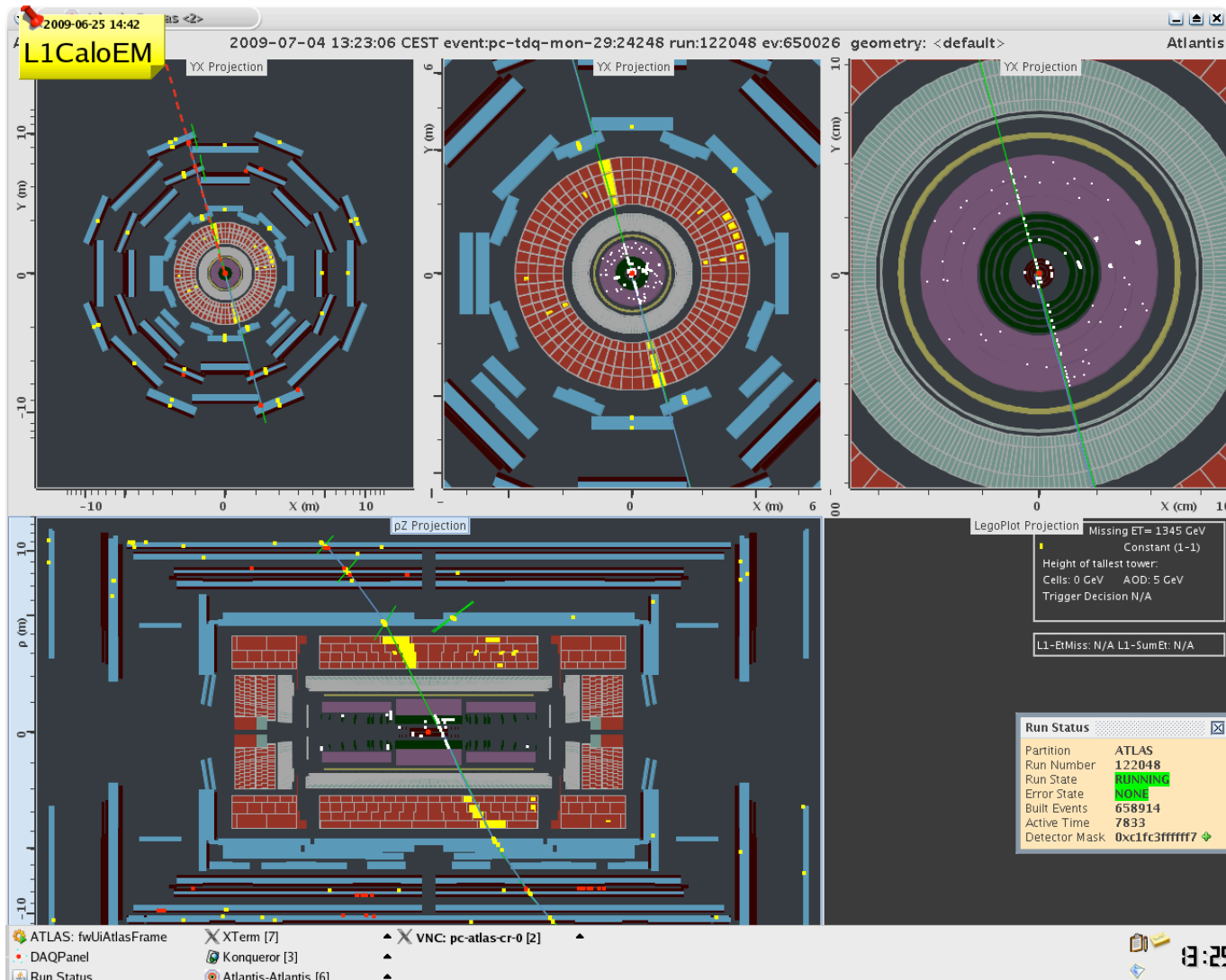


Shifter operated



Automatic

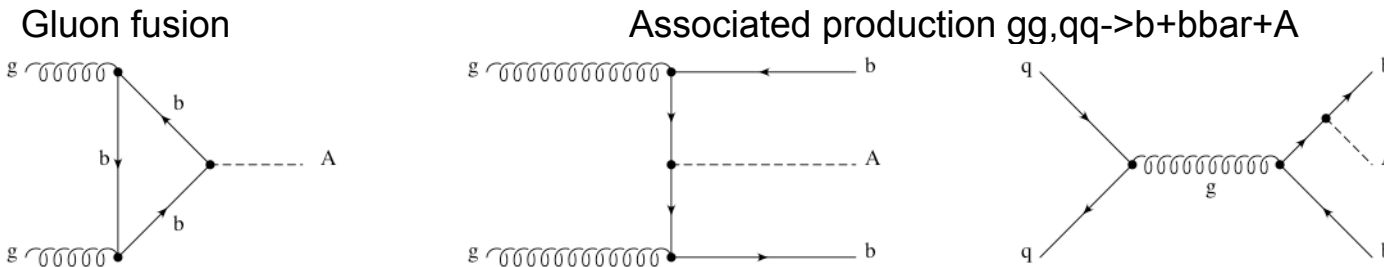




- Participation in June cosmic week as DAQ expert on call and Shift Leader
 - Cosmic event display taken while being shift leader
- Systems in the run:
 - Tile, LAr,
 - TRT, SCT, Pixel,
 - MDT, CSC
 - ZDC, Lucid
- Problems:
 - ROD 3 EBC blew PU up fuse



- MSSM Higgs to tau pairs in lepton hadron channel
- Production mechanisms



- Selection cuts: [MSSMHiggsToTauTauToLH](#)
 - Good electron, muon, jet, identification required
 - Good tau fake rejection required
 - Good missing E_t reconstruction required
- Use Higgs decay products to reconstruct the visible mass a la CDF
 - Set exclusion limits and discovery significance incorporating energy scale systematic uncertainties (AKA template morphing) using a Bayesian method called Profiling

This is a Bayesian method, when data comes, we will only need the data to set a limit. No PDFs, no toy MCs.



- The bayesian credibility limit is given by the p-value definition

$$p = \int_{\theta_L}^{\infty} \mathcal{P}(\theta', \nu | x) d\theta'$$

$$\mathcal{P}(\theta, \nu | x) = \frac{\mathcal{L}(\theta, \nu | x) \pi(\theta, \nu)}{\pi(x)}$$

- We do a cross section scan of the posterior probability

- The prior probability is considered factorizable

$$\pi(\theta, \nu) = \pi(\theta) \times \pi(\nu)$$

θ Parameter of interest

ν Nuisance parameters



- Assume signal prior constant up to a cutoff

$$\pi(\theta) = ct; \theta \in [0, \theta_{cutoff}]$$

$$\mathcal{P}(\theta, \nu | x) = \frac{\mathcal{L}(\theta, \nu | x) \pi(\nu)}{C}$$

- Assume nuisance prior described by truncated Gaussian p.d.f from subsidiary measurements

$$\pi(\nu) = G(\nu, \hat{\nu}, \varepsilon(\nu))$$

- Where we arbitrarily choose the constant

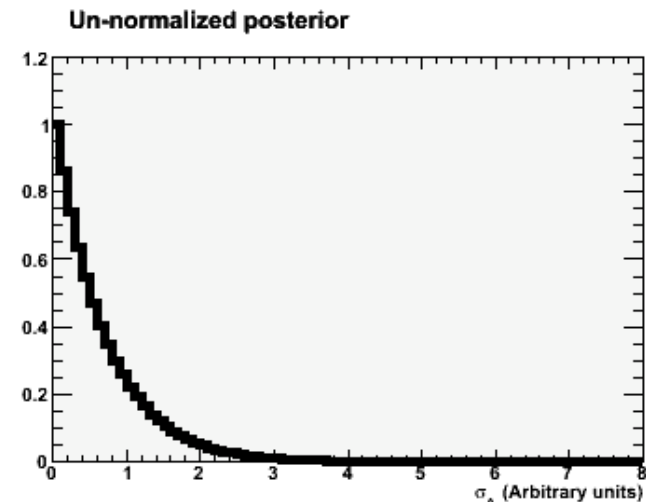
$$C = \mathcal{L}(\theta = 0, \nu | x) \pi(\nu)$$

- We evaluate the un-normalized posterior from

$$\theta \in [0, \theta_{cutoff}]; \mathcal{P} \in [1, 10^{-6}]$$

- We numerically integrate the posterior

$$1 - p = \sum_0^{\theta_L} \frac{\mathcal{L}(\theta, \nu | x) \pi(\nu)}{C}$$



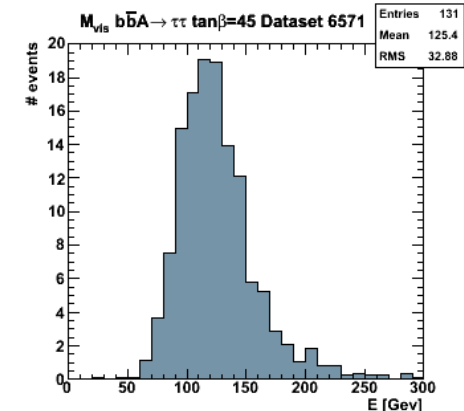
Understanding templates



- Given our observable (M_{vis}) we have efficiency distributions (templates) from MC, pseudo-experiments and eventually data

$$M_{vis} = |p_l^\alpha + p_h^\alpha + p_{miss}^\alpha|$$

$$p_{miss}^\alpha = \left(p_{miss}^x, p_{miss}^y, 0, \sqrt{(p_{miss}^x)^2 + (p_{miss}^y)^2} \right)$$



- The mean values for the binned likelihood function are a function of the parameter of interest and the nuisance parameters

$$\mathcal{L}(\theta, \nu | x) = \prod_i Pois(\mu_i = f(\theta, \nu) | x_i) = \prod_i \frac{\exp(-\mu_i) \mu_i^{x_i}}{x_i!}$$

- Our parameter of interest is the signal cross section
- Nuisance parameters can modify
 - the expected number of counts in each bin

$$\mu_i = L \sigma_A A_i$$

- the shape of the template (In our case by taking into account two other templates)

$$\mu_i = \sigma_A \left(A_i + f_j \frac{A_i^+ - A_i^-}{2} \right)$$

In both cases the prior for the nuisance parameters is the same Gaussian p.d.f.



- We consider a signal + background model in each bin

$$\mu_i = L\sigma_A A_i + L\sigma_Z Z_i$$

- We define a pseudo-experiment as a poisson fluctuation of the background.

$$x_i = Pois(L\sigma_Z Z_i)$$

- We define a binned Likelihood function of the data to the parameters

$$\mathcal{L}(\theta, \nu | x) = \prod_i \frac{\exp(-\mu_i) \mu_i^{x_i}}{x_i!}$$

- We take into account the priors on the nuisance parameters (Luminosity, cross section) with known systematic uncertainties

$$\mathcal{L}(\theta, \nu | x) \times \pi(\nu) = \prod_i \frac{\exp(-\mu_i) \mu_i^{x_i}}{x_i!} \times G(L, \hat{L}, \varepsilon(L)) \times G(\sigma_Z, \hat{\sigma}_Z, \varepsilon(\sigma_Z))$$

- We fit our posterior to the data, which is maximum at the best value of the parameters.

Adding template morphing



- The expected number of events in each bin take into account the variation between the positive and the negative shifted template, via a morphing parameter.

$$\mu_i = L\sigma_A \left(A_i^0 + f_j \frac{A_i^{j+} - A_i^{j-}}{2} \right) + L\sigma_Z \left(Z_i^0 + f_j \frac{Z_i^{j+} - Z_i^{j-}}{2} \right)$$

- The morphing parameter controls the strength of the energy scale uncertainty.
 - The prior for the morphing parameter is a Gaussian p.d.f. with mean zero and standard deviation one.
- Therefore the posterior probability shall take into account these systematic uncertainties

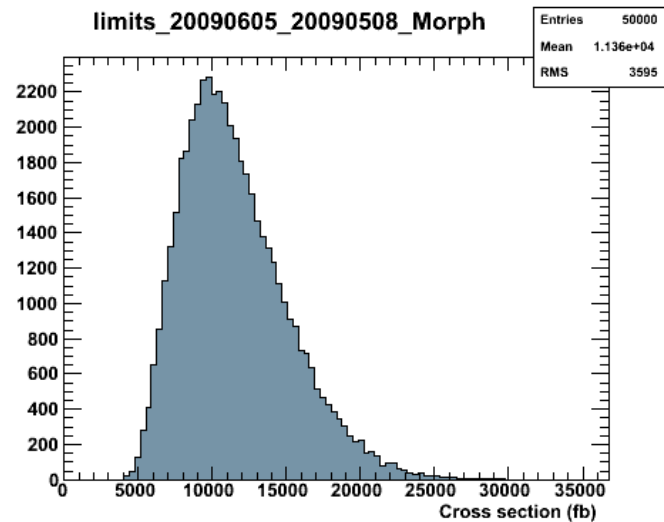
$$\mathcal{L}(\theta, \nu | x) \times \pi(\nu) = \prod_i \frac{\exp(-\mu_i) \mu_i^{x_i}}{x_i!} \times G(L, \hat{L}, \varepsilon(L)) \times G(\sigma_Z, \hat{\sigma}_Z, \varepsilon(\sigma_Z)) \times G(f_j, \hat{f}_j, \varepsilon(f_j))$$



- Setting exclusion limits for single background model
 - mA=150 GeV, tanBeta=45, 50k pseudo-experiments
- Single background model with jet and electron energy shifts
 - $\Delta E(\text{electron})=1\%$, $\Delta E(\text{jet})=5\%$,
- The expected number of events and the posterior probability are given by:

$$\mu_i = L\sigma_A \left(A_i^0 + f_e \frac{A_i^{e+} - A_i^{e-}}{2} + f_j \frac{A_i^{j+} - A_i^{j-}}{2} \right) + L\sigma_Z \left(Z_i^0 + f_e \frac{Z_i^{e+} - Z_i^{e-}}{2} + f_j \frac{Z_i^{j+} - Z_i^{j-}}{2} \right)$$

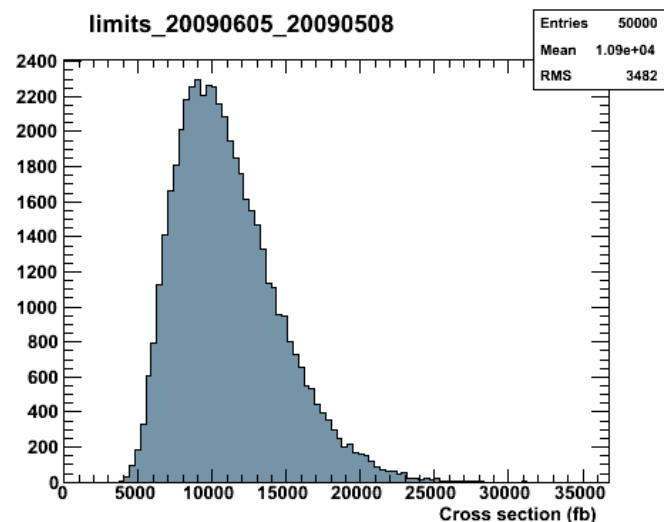
$$\mathcal{L}(\theta, \nu | x) \times \pi(\nu) = \prod_i \frac{\exp(-\mu_i) \mu_i^{x_i}}{x_i!} \times G(L, \hat{L}, \varepsilon(L)) \times G(\sigma_Z, \hat{\sigma}_Z, \varepsilon(\sigma_Z)) \\ \times G(f_j, \hat{f}_j, \varepsilon(f_j)) \times G(f_e, \hat{f}_e, \varepsilon(f_e))$$



Exclusion limit distribution with template morphing

50% (median): 10859 fb
 68% (1sigma): 7919 fb , 15085 fb
 95% (2sigma): 6082 fb , 19863 fb

Adding template morphing increases
 the the width of the sidebands
 and worsens the sensitivity



Exclusion limit distribution without template morphing

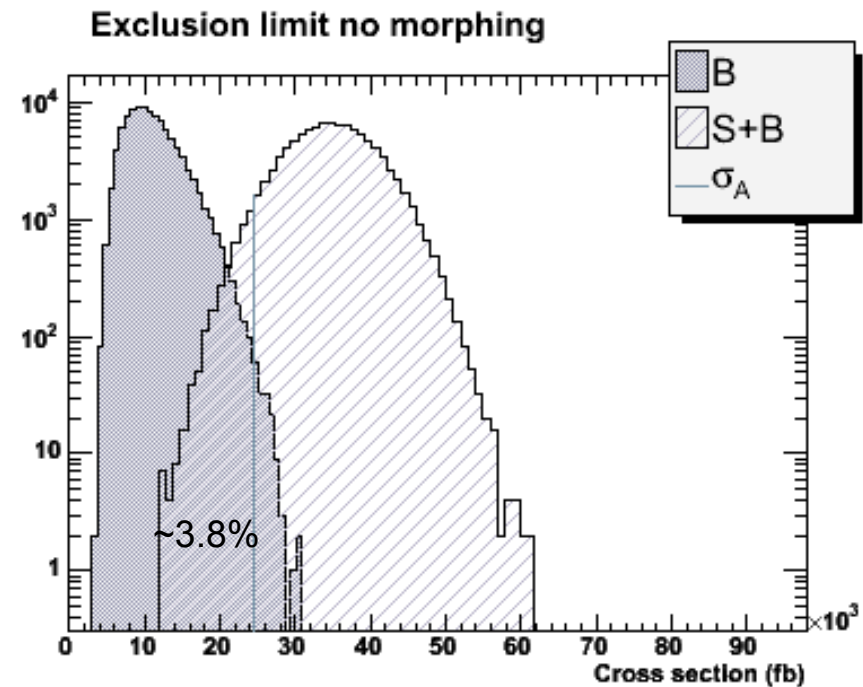
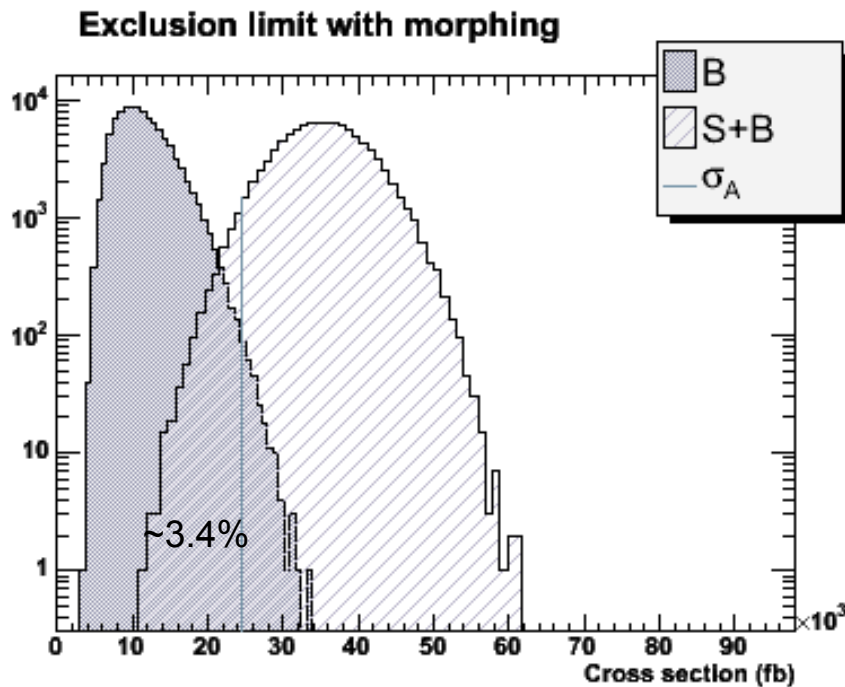
50% (median): 10492 fb
 68% (1sigma): 7552 fb , 14350 fb
 95% (2sigma): 5714 fb , 19128 fb



- We repeat the example but this time adding signal to each pseudo-experiment
 - 100% of the predicted cross-section by MSSM for the given parameters ($m_A=150$ GeV, $\tan\beta=45$, m_h^{\max} scenario, $\sigma_A=24500$ fb)

$$x_i = Pois(L\sigma_Z Z_i) + Pois(L\sigma_A A_i)$$

- The method to set exclusion limits is slightly conservative for the MSSM model presented
 - For conservative limit we expect $< 5\%$ of S+B pseudo-experiments to lie below the true value of S
 - In our case, $\sim 3.4\%$ (morphing) and $\sim 3.8\%$ (no morphing) of the pseudo-experiments lie under the signal cross-section.



Conclusions



- Online work is under control
- Still lot of work to do on my analysis