RECAST workflows in ATLAS SUSY

Ben Hodkinson

UNIVERSITY OF OXFORD

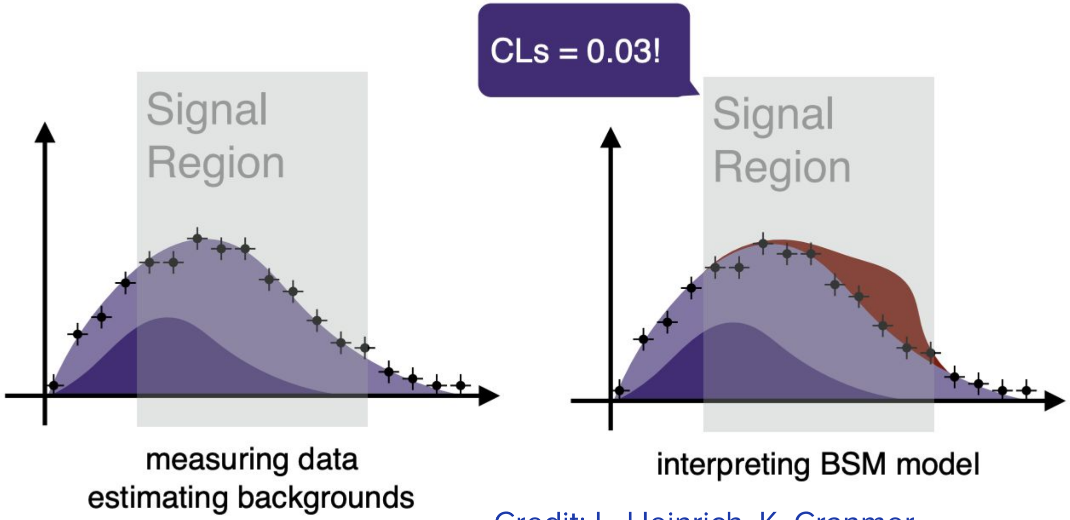
ATLAS EXPERIMENT

Overview

1) **What is RECAST and why do we need it?**
2) **My experience authoring RECAST workflows**
   - ➢ Anatomy of a workflow
   - ➢ Challenges
3) **My experience using existing RECAST workflows**
   - ➢ ATLAS pMSSM scan
   - ➢ Other ATLAS reinterpretation efforts
   - ➢ Challenges
4) **Incentivising RECAST**

Introduction

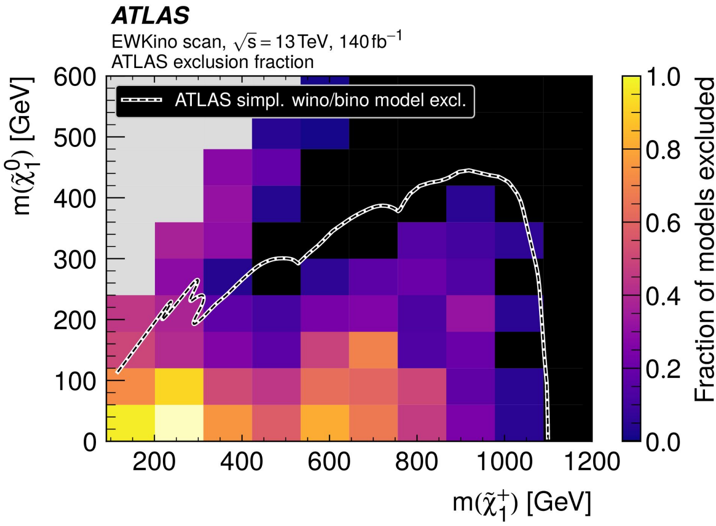
- Goal of the ATLAS SUSY group: discover new physics!
- An ATLAS search takes o(years) from initial idea to publication
  - ➢ Most of the work goes into taking data, designing/validating the analysis strategy, understanding Standard Model backgrounds
  - ➢ Plugging in a SUSY signal model at the end to discover/exclude is the easy bit
- In the absence of a discovery, how do we extract maximum value from this work?



Credit: L. Heinrich, K. Cranmer

Re-using ATLAS SUSY analyses

Project 19-dim pMSSM into 2D planes and evaluate our sensitivity:



Run-2 EW pMSSM scan result

(Re-)use cases:

- Process new signal points for statistical combinations
- Apply SUSY searches to non-SUSY models and vice versa
- Check new models aren't already excluded
  - ➢ Concentrate our efforts on viable scenarios
- Interpretation in wider SUSY parameter spaces (eg. pMSSM scans)
  - ➢ Build a coherent picture of the **global** ATLAS constraints on SUSY
  - ➢ Identify gaps in our search programme

How do we do any of this once the original analysis team has disbanded?

This requires **preservation** of an analysis workflow

RECAST

https://recast.docs.cern.ch/

- RECAST: Complete preservation of ATLAS analysis software environment and workflow

Analysis environment

→ Preserved in Docker images

\+

Analysis commands

→ How to use the preserved software

\+

Analysis workflow

→ How to connect each analysis step

Automate the re-execution of an analysis on a new signal:
Initial event sample → Calculate kinematic variables → Event selections → Statistical analysis

recast

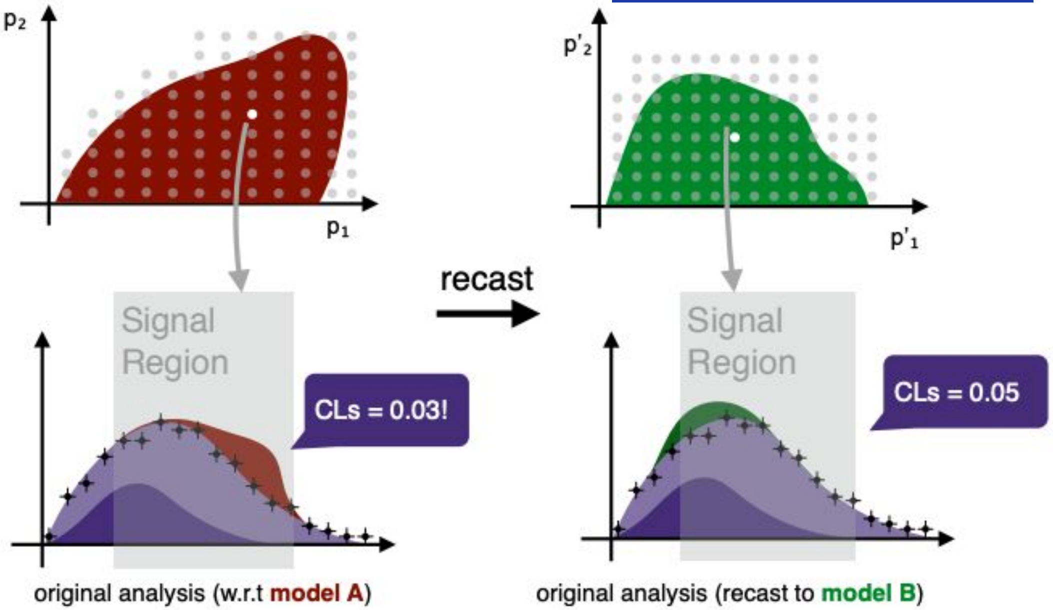
reana

- REANA: Infrastructure for running RECAST on the cloud
  - ➢ I’ve run over 5000 analysis jobs with REANA for 1000s models and eight analyses!

RECAST

- Only re-processing the **signal** (eg. a SUSY model)
  - ➢ Data and Standard Model backgrounds were already processed and preserved originally
- Analysis code and workflow preserved such that it can be re-run on new input
  - ➢ Most analyses have their own code framework and scripts
- Only used **internally** in ATLAS
  - ➢ Requires ATLAS-simulated signal event samples

Credit: L. Heinrich, K. Cranmer



RECAST user experience

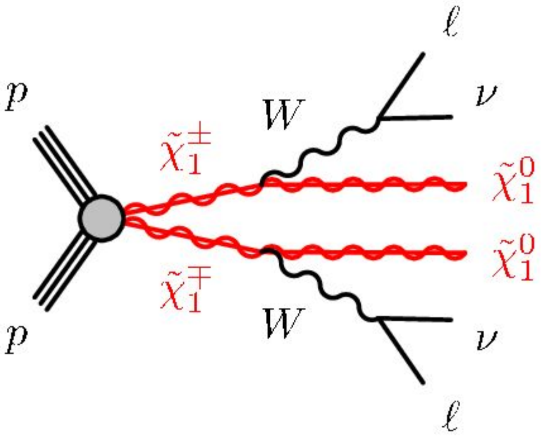
Two types of "user":

i. Analyser authoring a RECAST implementation for their analysis
ii. Analyser using RECAST to reinterpret an existing analysis

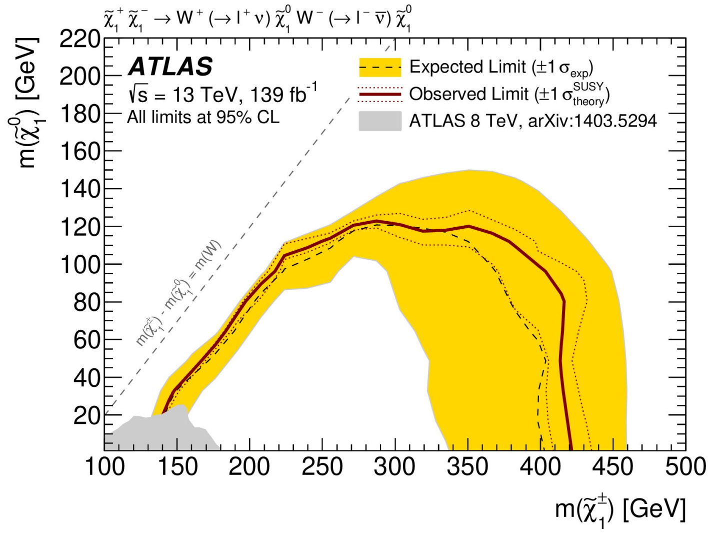
I will discuss my experience with both

(i) RECASTing an analysis

Anatomy of a RECAST workflow



- Example: [ATLAS two lepton + MET search]()
- Inputs:
  - ➢ Dataset ID
  - ➢ Cross-section, K-factor, filter efficiency
  - ➢ Sample name
  - ➢ EOS path to pile-up re-weighting files
  - ➢ XRootD path to signal samples
- Outputs:
  - ➢ Fit result (CLs)
  - ➢ Intermediate files (ntuples, workspace etc.)
- Steps:
  - ➢ Calculate kinematic variables and produce flat ntuple X 3 (simulated samples for each year of data-taking)
  - ➢ Merge ntuples
  - ➢ Calculate region yields
  - ➢ Perform fit



Anatomy of a RECAST workflow

Example: ATLAS two lepton + MET search

```
ewkmel:
  process:
    process_type: interpolated-script-cmd
    interpreter: bash
    script: |
      /recast_auth/getkrb.sh #used to acquire kerberos tokens
      # Set up the shell environment
      source /home/atlas/release_setup.sh
```

[Bash script for running this step]

```
      # Run
      python RunLocally.py --directory {XROOTD_PATH} --isatlfast --driver {DRIVER} --syst --submitDir {SUBMIT_DIR} --campaign {CAMPAIGN}
      cd {SUBMIT_DIR}/data-output
      N=$(ls *.root | wc -l)
      echo ======================
      echo $N root files created
      echo ======================
      if [ $N -ne 1 ]; then echo ERROR $N ROOT FILES CREATED - expected one. EXITING.; exit 1; fi;
  publisher:
    publisher_type: interpolated-pub
    publish:
      output: '{SUBMIT_DIR}'
  environment:
    environment_type: 'docker-encapsulated'
    image: gitlab-registry.cern.ch/bhodkins/ewkmel
    #imagetag: master-c152a3f6
    imagetag: master-77fb58e3
```

Steps.yml specifies workflow steps

“Ewkmel” is one of five steps in this workflow
➔ performs initial preselections

Output to be used by next step (← `output: '{SUBMIT_DIR}'`)

Docker image – built automatically using gitlab CI (← `image: gitlab-registry.cern.ch/bhodkins/ewkmel`)

Anatomy of a RECAST workflow

Example: ATLAS two lepton + MET search

```
stages:
- name: ewkmel_mc16a
  dependencies: [init]
  scheduler:
    scheduler_type: singlestep-stage
    parameters:
      XROOTD_PATH: {'step': init, output: XROOTD_PATH}
      PRW_PATH: {'step': init, output: PRW_PATH}
      DRIVER: xrootd
      SUBMIT_DIR: '{workdir}/submitDir_mc16a'
      CAMPAIGN: mc16a
      DSID: {'step': init, output: DSID}
    step: {$ref: 'steps.yml#/ewkmel'}

- name: loopsusy_mc16a
  dependencies: [init, ewkmel_mc16a]
  scheduler:
    scheduler_type: singlestep-stage
    parameters:
      INPUT_PATH: {'step': ewkmel_mc16a, output: output}
      DRIVER: local
      SUBMIT_DIR: '{workdir}/submitDir_mc16a'
      XSEC_PB: {'step': init, output: XSEC_PB}
      FILTER_EFF: {'step': init, output: FILTER_EFF}
      KFACTOR: {'step': init, output: KFACTOR}
      DSID: {'step': init, output: DSID}
      POINT: {'step': init, output: POINT}
    step: {$ref: 'steps.yml#/loopsusy'}
```

recast

workflow.yml specifies how the steps are chained together

This step needs to run after **ewkmel**

Where to find necessary output from ewkmel step

User-specified inputs

2L + MET workflow

Convert to flat ntuple with initial preselections

Merge files from each MC campaign

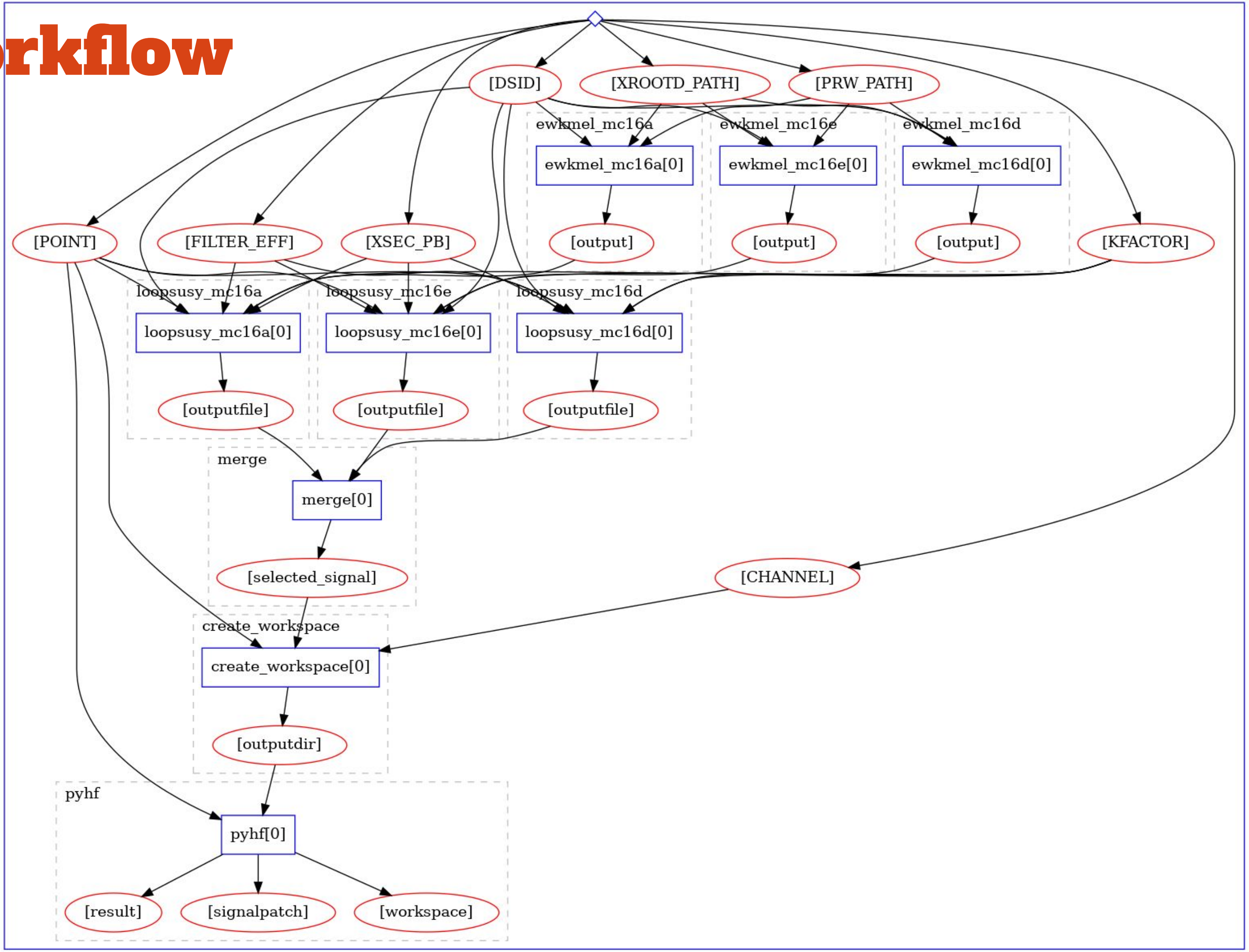
Calculate region yields (HistFitter workspace format)

Perform statistical fit

2023-05-30 12:50:22

reana

2L + MET workflow



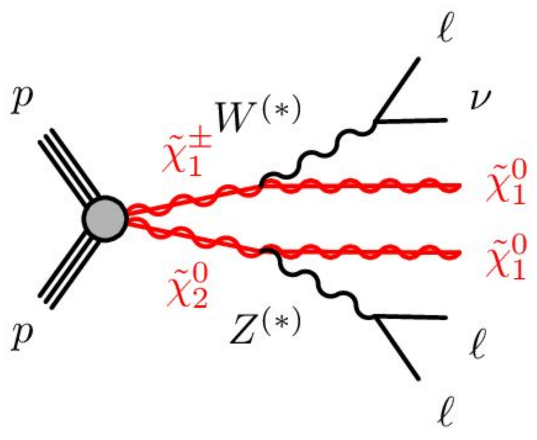
User experience as a RECAST author

- **It is in principle “easy” to implement an analysis in RECAST**
  - ➢ Just have to write two YAML files (steps.yml and workflow.yml)
  - ➢ Many pieces should already exist (gitlab CI docker images + analysis scripts)
- **But in practice:**
  - ➢ Docker, gitlab CI, YAML, kubernetes, RECAST, REANA aren’t necessarily in the typical HEP PhD student toolkit
  - ➢ User might not be familiar with every step of their analysis
- **My experience:** Looks simple on paper but debugging was not straightforward
- **Often RECAST is a bit of an afterthought – cultural issue**
  - ➢ Understandably lower priority than the publication
  - ➢ Analysis team may have disbanded / students graduated
  - ➢ In my case, I wasn’t on the original analysis team and implemented the RECAST over a year after the paper publication

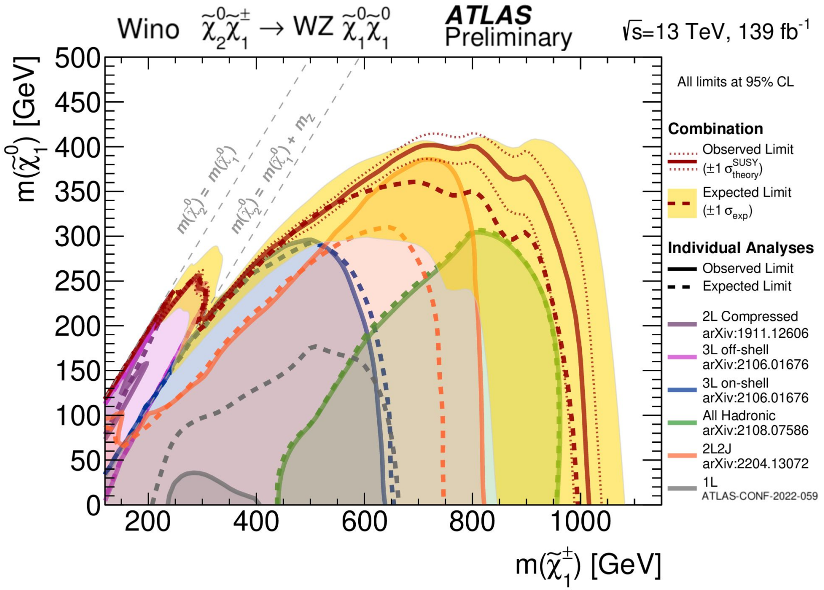
(ii) RECAST in action

Supersymmetry

- Minimal Supersymmetric Standard Model (MSSM) has **> 100** unknown parameters
- Searches are optimized & interpreted using **2D** slices of the MSSM
    - ➢ **“Simplified models”**
    - ➢ Results in easy-to-interpret 2D exclusion plots
    - ➢ Far from a complete picture of our sensitivity/exclusion of the full MSSM
    - ➢ How do we connect searches for different simplified models?



[SUSY-2020-05]

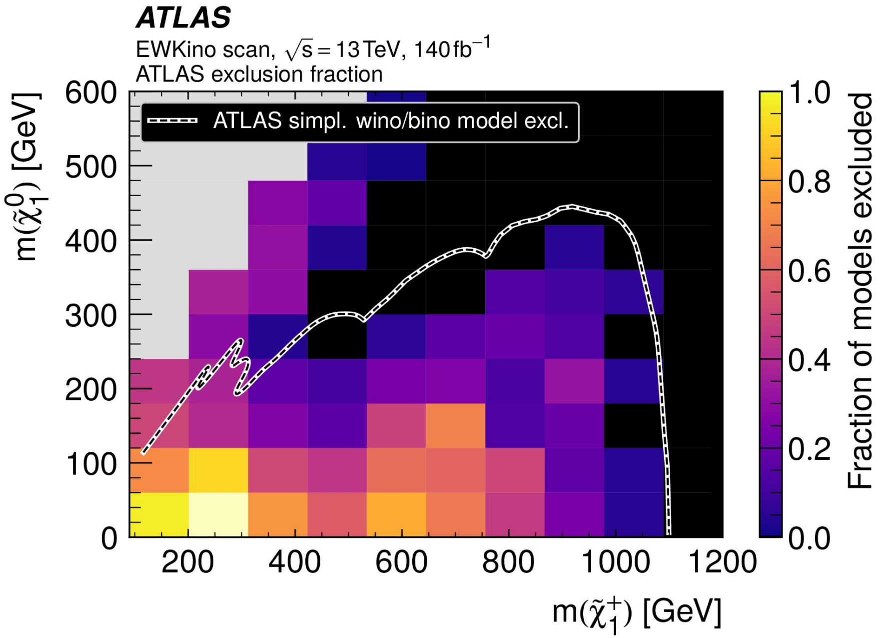


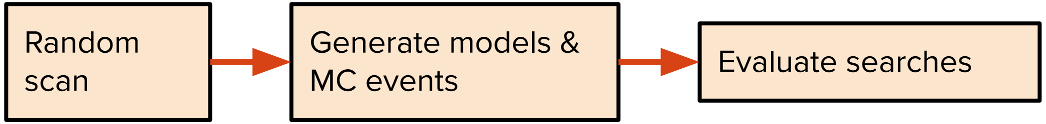
pMSSM scans

- **pMSSM** = 19-dimensional space of viable SUSY models

- Includes a range of production and decay modes

- pMSSM scans are important for understanding our **sensitivity** to realistic SUSY scenarios and highlight **gaps** to be targeted



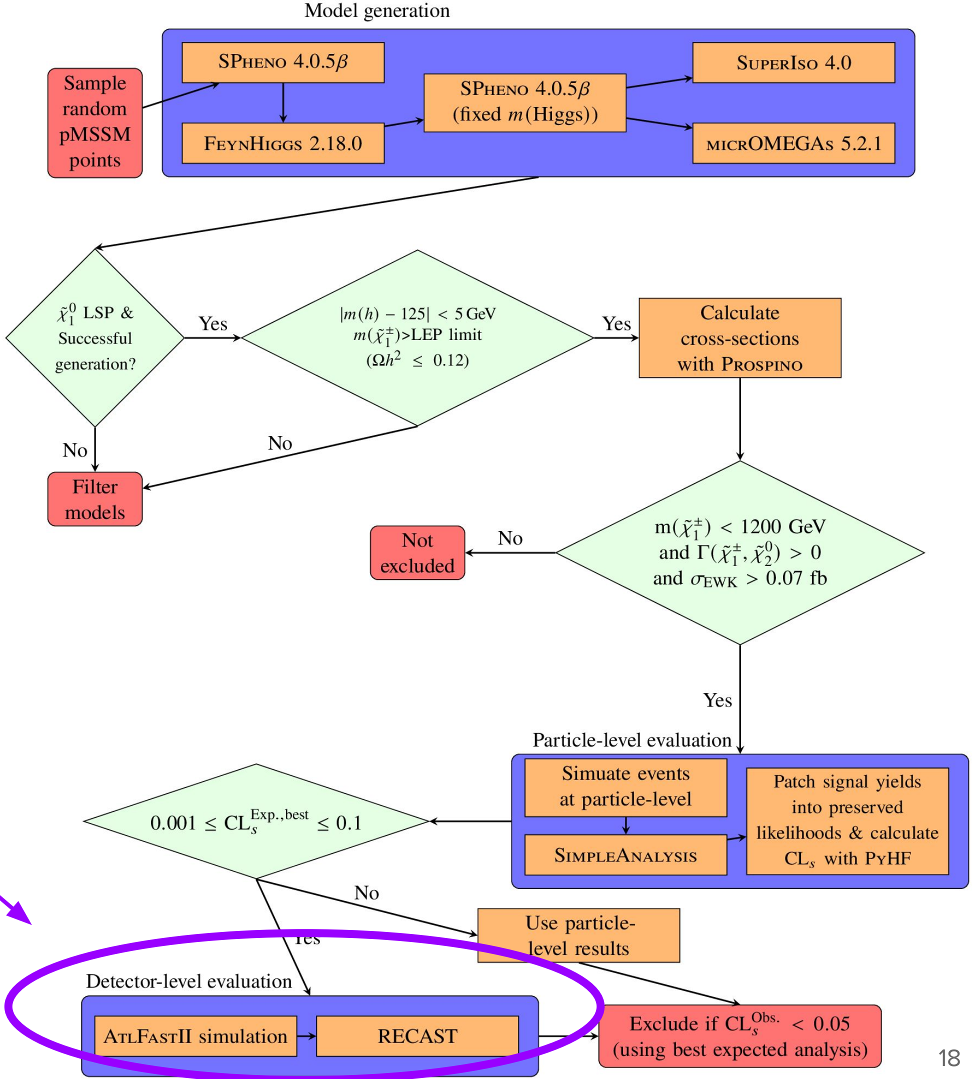


Run-2 EW pMSSM scan result

This requires re-running our searches on 1000s of new signal models

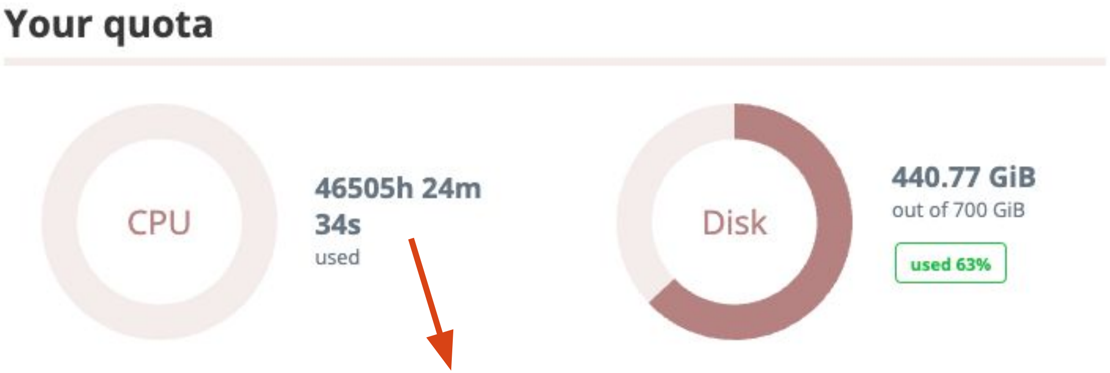
Workflow

- Workflow to evaluate exclusion for a sample of pMSSM models
- Implemented in python using mySQL database to store results of each step
- Various constraints applied to pick out interesting models
- **RECAST is used to apply SUSY searches to these models**

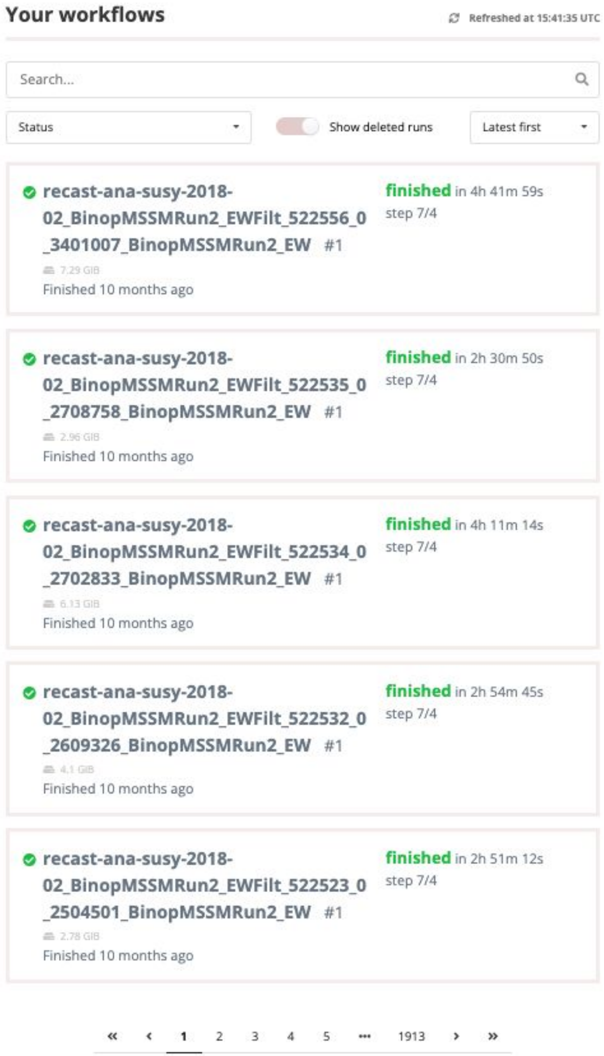


recast and reana in the pMSSM scan

- Eight analyses run using RECAST
- 1878 models were processed with RECAST
  - ➢ 9% of the 21,177 models in the scan
- 9561 REANA jobs
  - ➢ Including many failed tests and re-runs
  - ➢ Web-page monitoring very useful



4.9 hrs per job on average



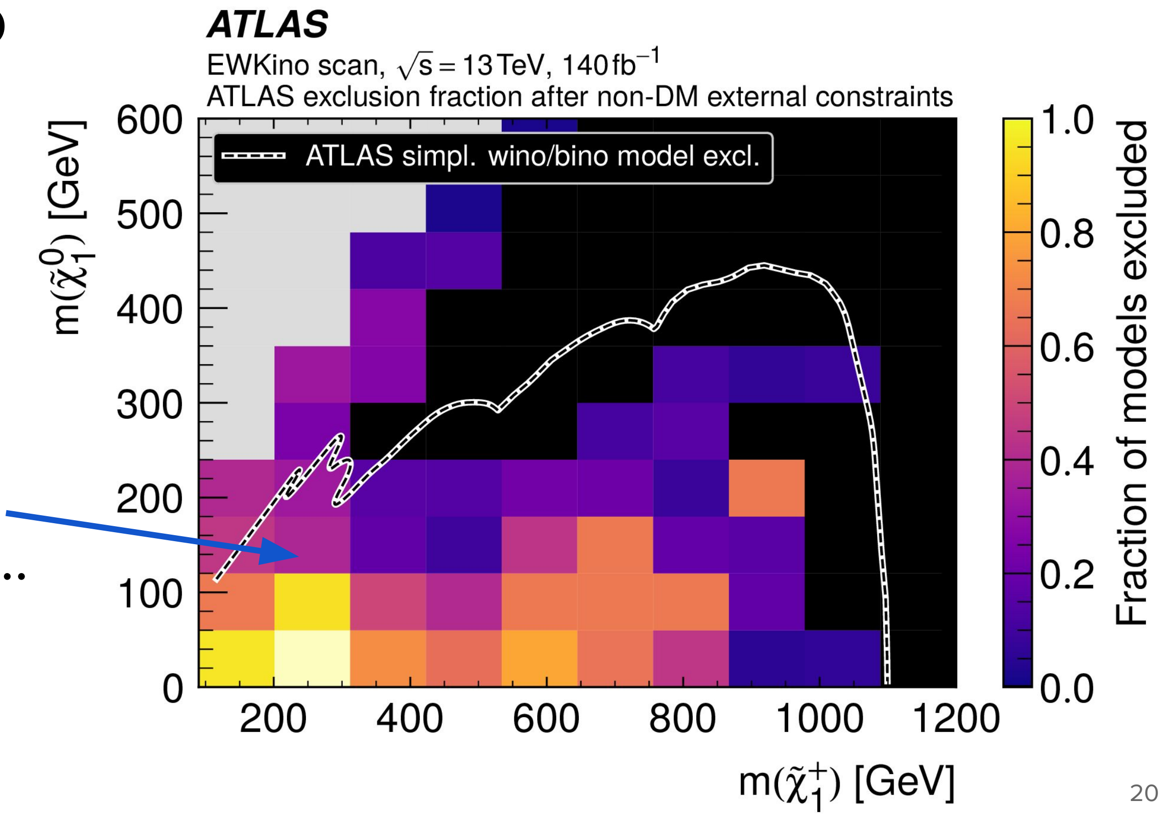
pMSSM results

Links

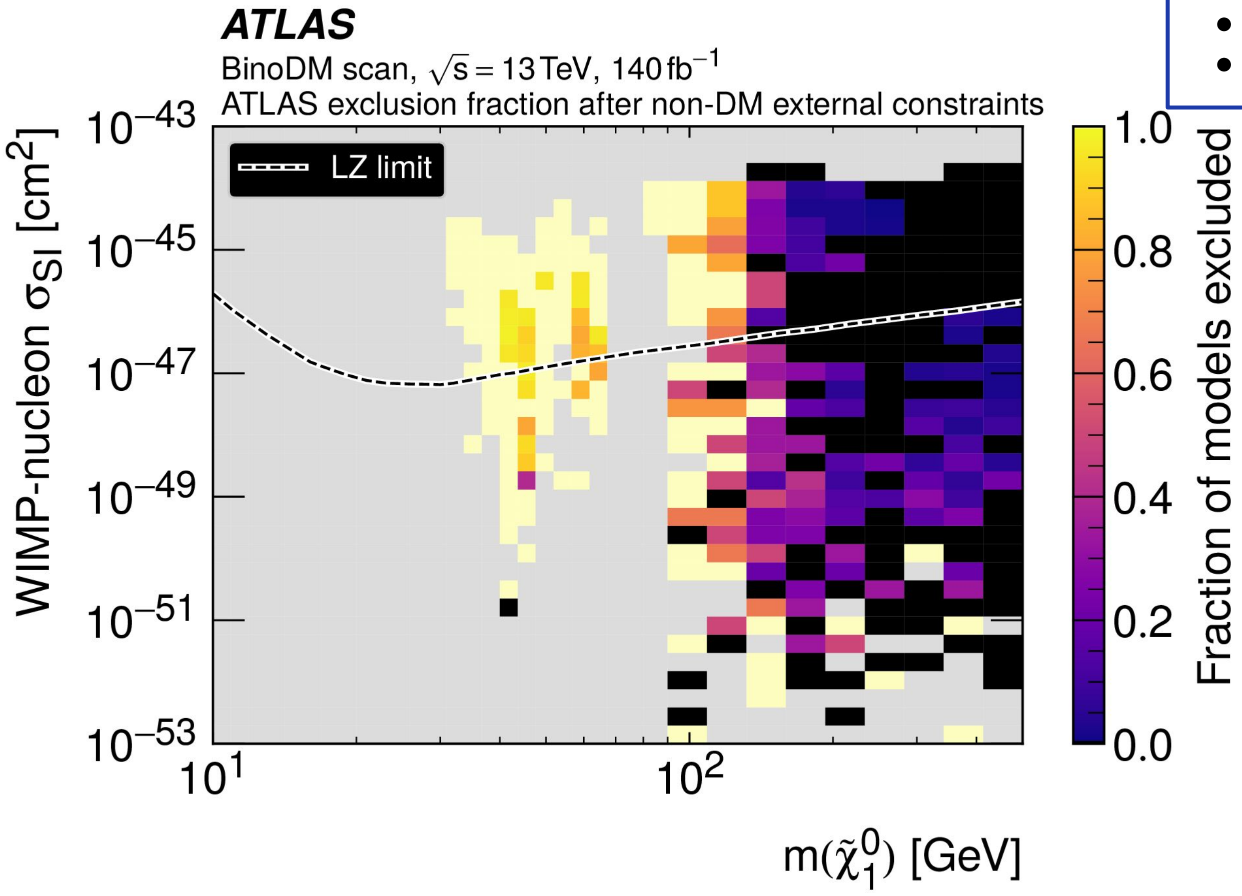
- pMSSM paper
- CERN courier article on pMSSM results
- ATLAS briefing on pMSSM paper
- pMSSM paper HEP data entry

Project 19D models into 2D planes and compare to simplified model results

Reveal that bins inside simplified model contours don't have 100% exclusion...



pMSSM results



Links

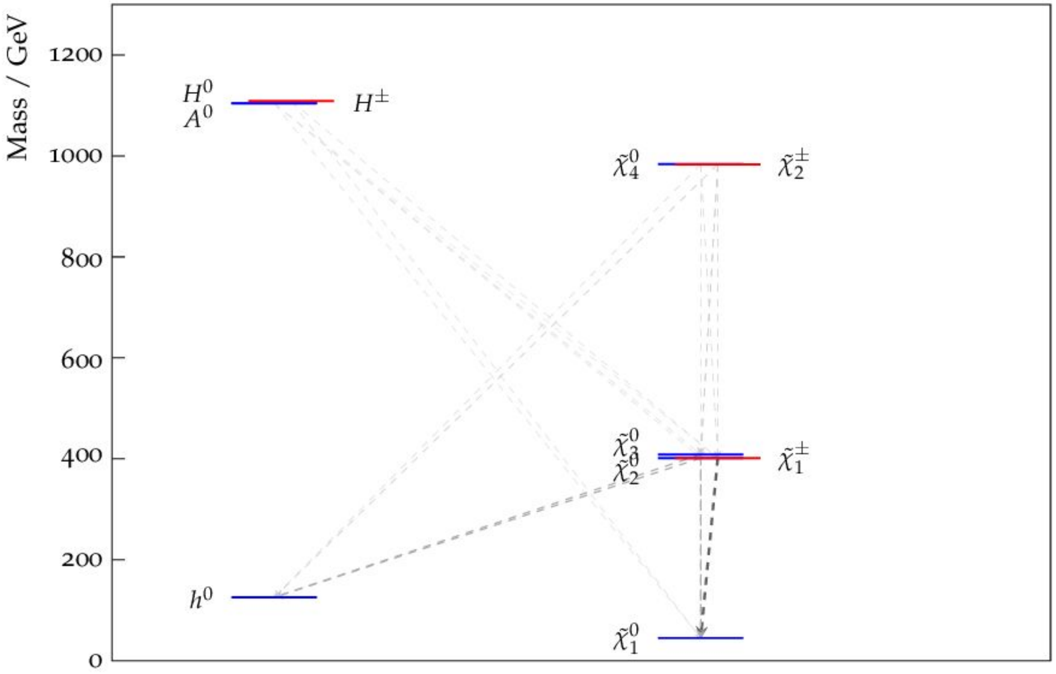
- pMSSM paper
- CERN courier article on pMSSM results
- ATLAS briefing on pMSSM paper
- pMSSM paper HEP data entry

- **pMSSM models come with dark matter interaction cross-sections**
- **Can compare ATLAS to direct detection experiments**

Benchmark models

Links

- pMSSM paper
- CERN courier article on pMSSM results
- ATLAS briefing on pMSSM paper
- pMSSM paper HEP data entry

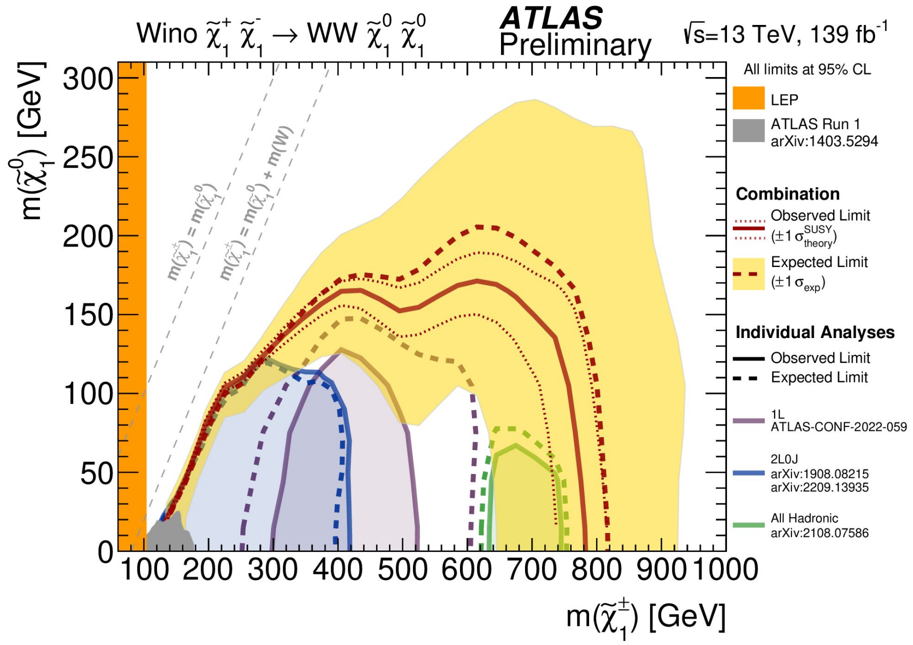


Identified SUSY models that have not been excluded

Smaller production cross-section than typical simplified models
➔ Greater "Higgsino" content of $\tilde{\chi}^\pm_1/\tilde{\chi}^0_2$

RECAST in action: ATLAS SUSY Combinations

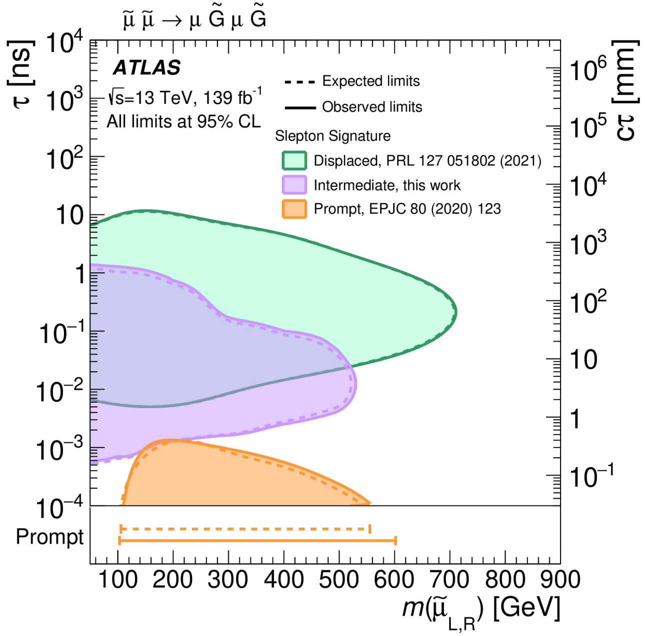
- Statistical combinations of multiple searches which targeted the same simplified models
- Problem: Lack of harmonization in the signal points available from each analysis
- Solution: Generate new signal points and use RECAST to produce necessary inputs for the combination
- More details in [the paper]



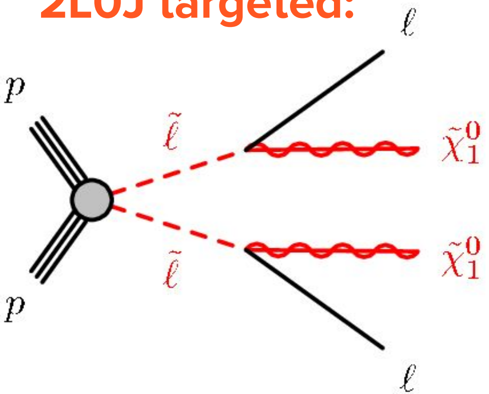
RECAST in action: Displaced smuon reinterpretation

[SUSY-2020-09]

- Recent search for **displaced muons** targeted GMSB model with long-lived smuons
- RECAST reinterpretation allowed comparison with **2L0J** search for prompt sleptons
  - ➢ Puts the **micro-displaced** search in context and makes the message of the paper stronger

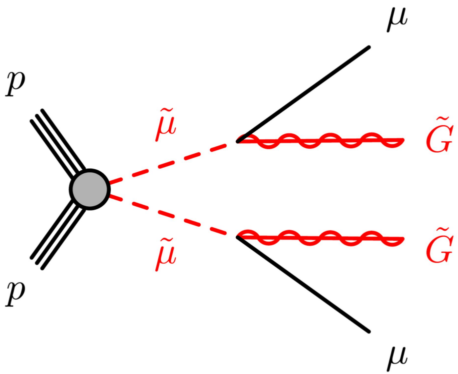


2L0J targeted:



RECAST

reinterpretation



Challenges using RECAST

- **Harmonisation/standardisation of inputs**
  - ➢ Every analysis works slightly differently
  - ➢ How are inputs accessed (eg. cross-sections, pile-up reweighting files)?
    - ■ CVMFS, input parameter, file on EOS?
  - ➢ Theoretical uncertainties?
  - ➢ Event yield normalization → inconsistent approaches
    - ■ Impacts whether to provide total or per-sub-process cross-section as input
  - ➢ pMSSM team created a framework to handle automated submission of REANA jobs with the various input formats

- **Hidden dependencies on original analysis samples (ie. simplified models) and input file naming**
  - ➢ Hard-coded if-statements related to sample ID's or names
  - ➢ Appropriate sample format ("derivation") not documented properly

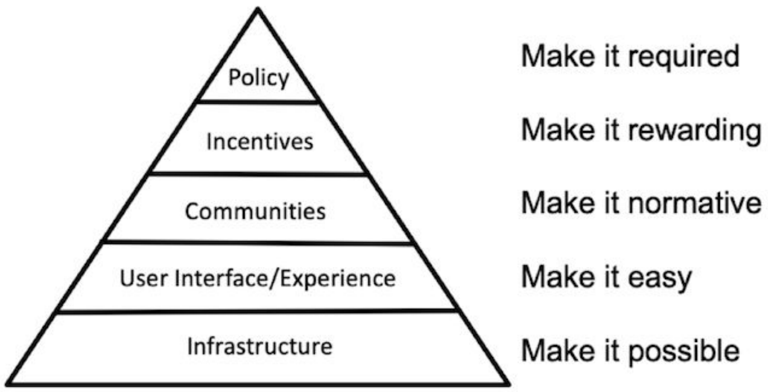
Challenges using RECAST

- **Ideally RECAST implementations work out of the box**
  - ➢ When this is the case, the experience is very smooth
- **In practice:**
  - ➢ pMSSM team spent a lot of time on technical implementation, validation and debugging of RECASTs
  - ➢ RECASTs which don't reproduce original analysis results
  - ➢ RECASTs which don't work with new signal points
  - ➢ Paper was descoped somewhat due to time spent on this
- **Found it difficult to get analysis support – often analysis experts have moved on**
  - ➢ This is precisely the problem RECAST is supposed to solve!

RECAST incentives

Two types of "user":

i. Analyser authoring a RECAST implementation for their analysis
ii. Analyser using RECAST to reinterpret an existing analysis (eg. the pMSSM analysis team)



Open science pyramid

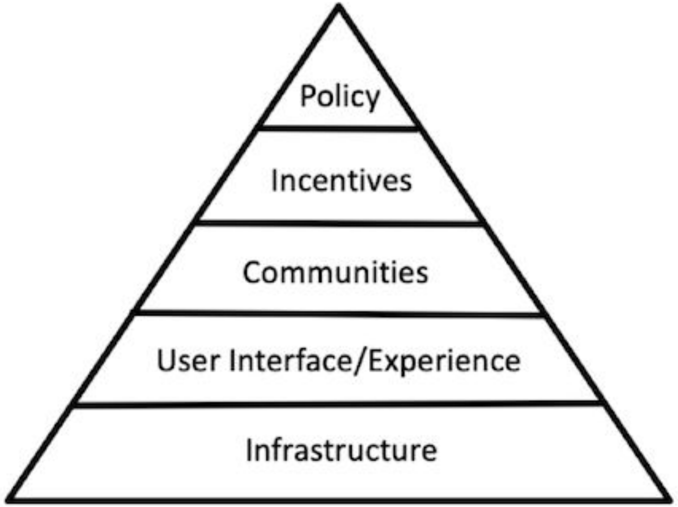
What is the incentive for user (i) do make a (good quality) RECAST for user (ii) ?

➢ User (i) already has a publication from their analysis and has higher priorities than RECAST-development (thesis-writing, moved to new analysis/experiment/field)

➢ It's user (ii) who gets the publication a few years later from using RECAST

➢ Often user (ii) has to finish/fix a RECAST, which defeats the purpose!

RECASTincentives



Open science pyramid

Make it required → Would be very unpopular (Student can't graduate because a paper is held up by RECAST development?)

Make it rewarding → ???

Make it normative → **~50 SUSY RECASTs available***
19 available* for use in pMSSM scans via intermediary framework

Make it easy → **Easy in principle, teething problems in practice**

Make it possible ✓

* in various states of development and validation

Conclusions

- **Reinterpretation produces very valuable physics results**
  - Link our searches, build a coherent picture of sensitivity, identify gaps in sensitivity

- **RECAST is an important tool for preserving and re-deploying analyses**

- **REANA provides a smooth experience running RECAST workflows at scale**

- **Open question:** How to incentivise current analysers to produce good-quality, validated RECAST workflows that work out-of-the-box for future users?

- **The ATLAS SUSY group has a large and growing collection of RECAST implementations – how should we use them?**
  - Currently used internally for ATLAS reinterpretation efforts
  - Original RECAST proposals envisaged cross-collaboration between theorists and experimentalists → “Request” / “Response” system

Backup

EWK pMSSM scan

Random scan with uniform priors over parameters relevant to production of charginos and neutralinos:

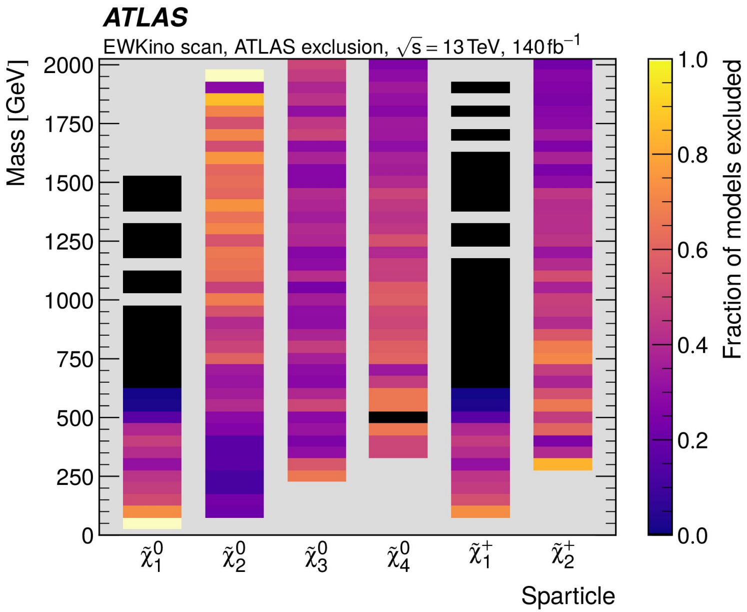
Parameter	min	max	Note
$M_{\tilde{L}_1}$ (=$M_{\tilde{L}_2}$)	10 TeV	10 TeV	Left-handed slepton (first two gens.) mass
$M_{\tilde{e}_1}$ (=$M_{\tilde{e}_2}$)	10 TeV	10 TeV	Right-handed slepton (first two gens.) mass
$M_{\tilde{L}_3}$	10 TeV	10 TeV	Left-handed stau doublet mass
$M_{\tilde{e}_3}$	10 TeV	10 TeV	Right-handed stau mass
$M_{\tilde{Q}_1}$ (=$M_{\tilde{Q}_2}$)	10 TeV	10 TeV	Left-handed squark (first two gens.) mass
$M_{\tilde{u}_1}$ (=$M_{\tilde{u}_2}$)	10 TeV	10 TeV	Right-handed up-type squark (first two gens.) mass
$M_{\tilde{d}_1}$ (=$M_{\tilde{d}_2}$)	10 TeV	10 TeV	Right-handed down-type squark (first two gens.) mass
$M_{\tilde{Q}_3}$	2 TeV	5 TeV	Left-handed squark (third gen.) mass
$M_{\tilde{u}_3}$	2 TeV	5 TeV	Right-handed top squark mass
$M_{\tilde{d}_3}$	2 TeV	5 TeV	Right-handed bottom squark mass
M_1	-2 TeV	2 TeV	Bino mass parameter
M_2	-2 TeV	2 TeV	Wino mass parameter
μ	-2 TeV	2 TeV	Bilinear Higgs mass parameter
M_3	1 TeV	5 TeV	Gluino mass parameter
A_t	-8 TeV	8 TeV	Trilinear top coupling
A_b	-2 TeV	2 TeV	Trilinear bottom coupling
A_τ	-2 TeV	2 TeV	Trilinear τ lepton coupling
M_A	0 TeV	5 TeV	Pseudoscalar Higgs boson mass
$\tan\beta$	1	60	Ratio of the Higgs vacuum expectation values

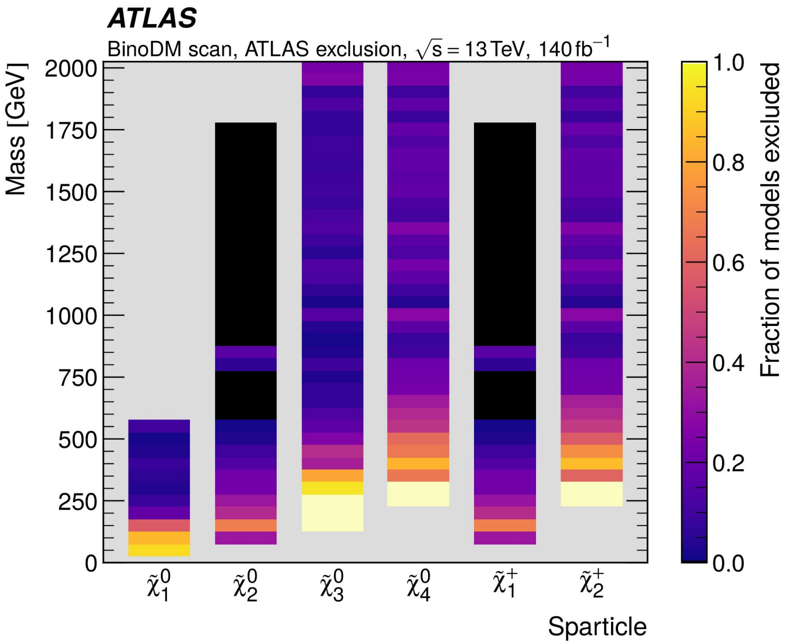
Links

- pMSSM paper
- CERN courier article on pMSSM results
- ATLAS briefing on pMSSM paper
- pMSSM paper HEP data entry

Overall exclusion

ATLAS exclusion of each sparticle (after all external and dark matter constraints)





EWKino scan
Mainly wino/higgsino LSP

Bino-DM scan

pMSSM model samples

The pMSSM team can provide analysis teams with pMSSM MC samples from the Run-2 scans
How should we use these?

1) Validation

- Check RECAST works with new signal points
- CI check?

→ **O(1) models**

2) Benchmarks used in analysis development

- Check selections aren't over-tuned to simplified models
- See what unique pMSSM sensitivity you are accessing

→ **O(10) relevant non-excluded models**

3) Integrate pMSSM workflow with developing analyses → evaluation on full Run-2 pMSSM model sets

- Rather than packing up and re-doing this exercise in 3 years, can we make pMSSM interpretations a central and continuous part of our analysis?
- Analyses document pMSSM interpretation plots → show what unique models your search is sensitive to.
- The pMSSM team can then focus on the global picture rather than processing individual analyses.

→ **O(100) non-excluded models**