GAMBIT

the Global And Modular BSM Inference Tool

Andy Buckley, University of Glasgow
("borrowing" enormously from previous tutorials
by Martin White, Jonathan Cornell, Anders Kvellestad)

Possible discoveries and assumptions

- We might discover something decaying visibly:

  Default assumption: something to do with EWSB

- We might discover something decaying (semi-) invisibly

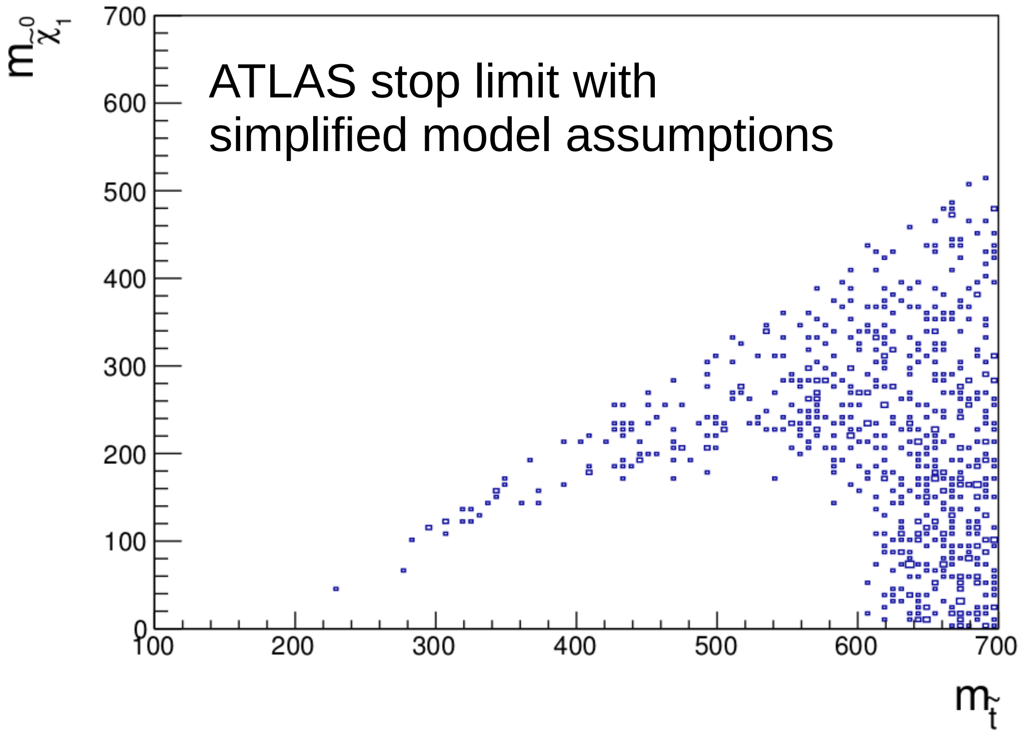
  Default assumption: something to do with DM

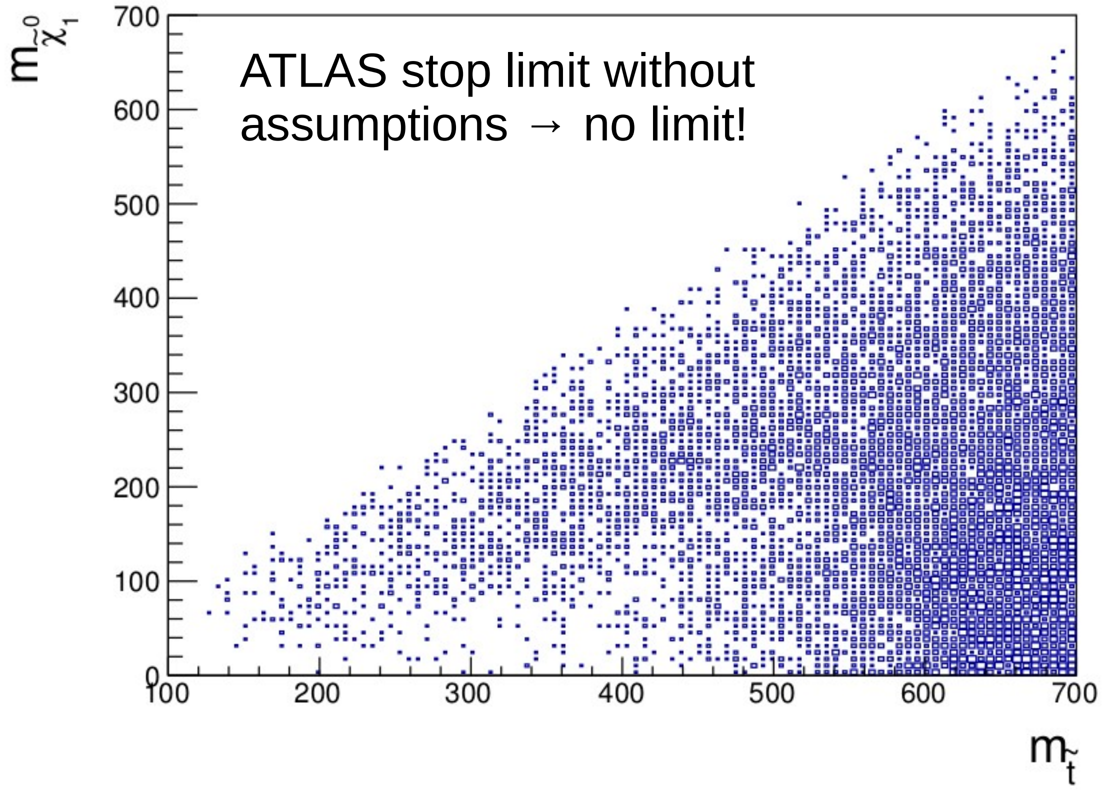
- We might discover nothing extra at all at the LHC

  How do we make further progress?

What about LHC non-discoveries?

- They tell us a lot, but are infamously hard to reinterpret – how should we do that?





The answer: use as much data as possible

- Combine ATLAS+CMS null and positive results to test specific theories
- Don't forget LHCb!
- Don't forget other experiments...

Other experiments

- low-energy accelerators
- measurements of the magnetic moment of the muon
- beam dump/fixed target
- electroweak precision tests
- dark matter direct detection experiments
- searches for antimatter in cosmic rays
- nuclear cosmic ray ratios
- radio astronomy data
- effects of dark matter on reionisation, recombination and helioseismology
- the observed dark matter cosmological abundance
- neutrino masses and mixings
- gamma ray searches (e.g. FERMI-LAT, HESS, CTA, etc)

How to combine data

- Correct answer is to use a global statistical fit
- Frequentist or Bayesian methods available
- Calculate a **combined likelihood**:

$$\mathcal{L} = \mathcal{L}_{\rm collider}\mathcal{L}_{\rm DM}\mathcal{L}_{\rm flavor}\mathcal{L}_{\rm EWPO}\cdots$$

<u>Parameter estimation</u>

Given a particular model, which set of parameters best fits the available data

(Rigorous exclusion limits and parameter measurements)

<u>Model comparison</u>

Given a set of models, which is the best description of the data, and how much better is it?

(Model *X* is now worse than model *Y*)

The dream

→ Global fit results

- Recent years have seen an explosion of tools that make study of user-defined Lagrangians easier

  - e.g. Feynrules → Madgraph, CalcHEP → Micromegas, MadDM, NLOCT + much, much more

- Even so, a general global fit tool requires some very tricky innovations:

  - calculations are not allowed to know about Lagrangian parameters – how do you do that?
  - how do you make an easy interface for tying existing code together?
  - how do you store parameters in a scale independent way, but reintroduce scales in calculations?
  - how do you make LHC constraints model independent?
  - how do you make astrophysical constraints model independent?
  - ***how do we do all of this fast enough to get convergence within the age of the universe?***

GAMBIT: The Global And Modular BSM Inference Tool

gambit.hepforge.org

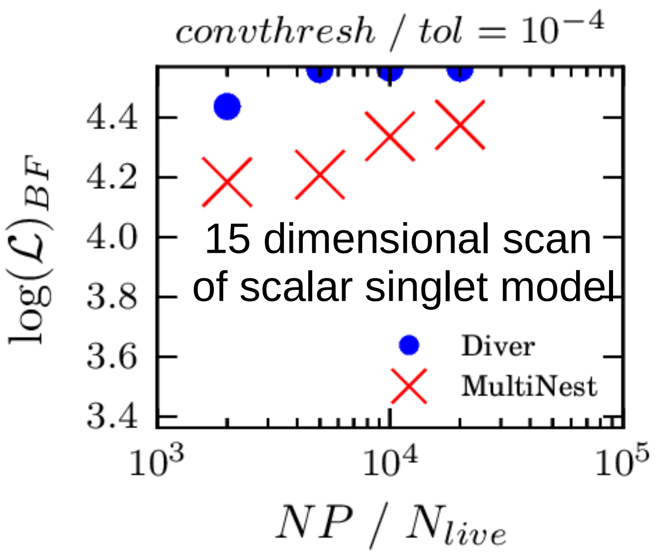
- Fast definition of new datasets and theoretical models
- Plug and play scanning, physics and likelihood packages
- Extensive model database – not just SUSY
- Extensive observable/data libraries
- Many statistical and scanning options (Bayesian & frequentist)
- *Fast* LHC likelihood calculator
- Massively parallel
- Fully open-source

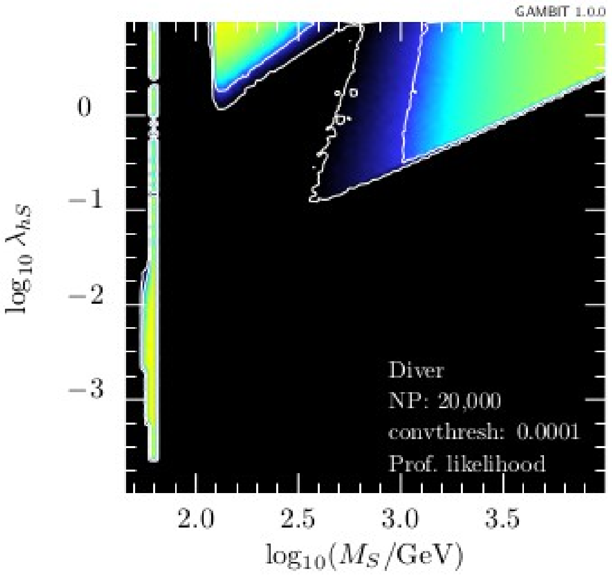
ATLAS	F. Bernlochner, A. Buckley, P. Jackson, M. White
LHCb	M. Chrząszcz, N. Serra
Belle-II	F. Bernlochner, P. Jackson
Fermi-LAT	J. Conrad, J. Edsjö, G. Martinez, P. Scott
CTA	C. Balázs, T. Bringmann, M. White
CMS	C. Rogan
IceCube	J. Edsjö, P. Scott
XENON/DARWIN	B. Farmer, R. Trotta
Theory	P. Athron, C. Balázs, S. Bloor, T. Bringmann, J. Cornell, J. Edsjö, B. Farmer, A. Fowlie, T. Gonzalo, J. Harz, S. Hoof, F. Kahlhoefer, S. Krishnamurthy, A. Kvellestad, F.N. Mahmoudi, J. McKay, A. Raklev, R. Ruiz, P. Scott, R. Trotta, A. Vincent, C. Weniger, M. White, S. Wild

31 Members in 9 Experiments, 12 major theory codes, 11 countries

Global

- Complete global statistical fit framework
- Can be Bayesian, Frequentist or other (random, grid, etc)
- Interfaced to the best + fastest scanners available:
  Multinest, MCMC, Diver (new differential evolution scanner)

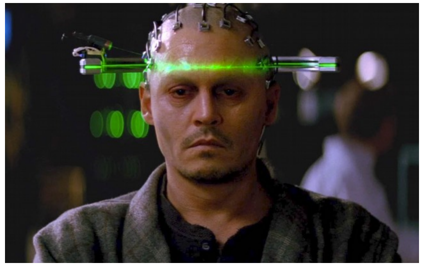




Publication ready plots available using *pippi* plotting code on the GAMBIT HDF5 output

Global and Modular

- **ColliderBit:** collider observables including Higgs + SUSY Searches from ATLAS, CMS, LEP
- **DarkBit:** dark matter observables (relic density, direct & indirect detection)
- **FlavBit:** including $g-2$, $b \to s\gamma$, B decays (new channels), angular obs., theory unc., LHCb likelihoods
- **SpecBit:** generic BSM spectrum object, providing RGE running, masses, mixings
- **DecayBit:** decay widths for all relevant SM and BSM particles

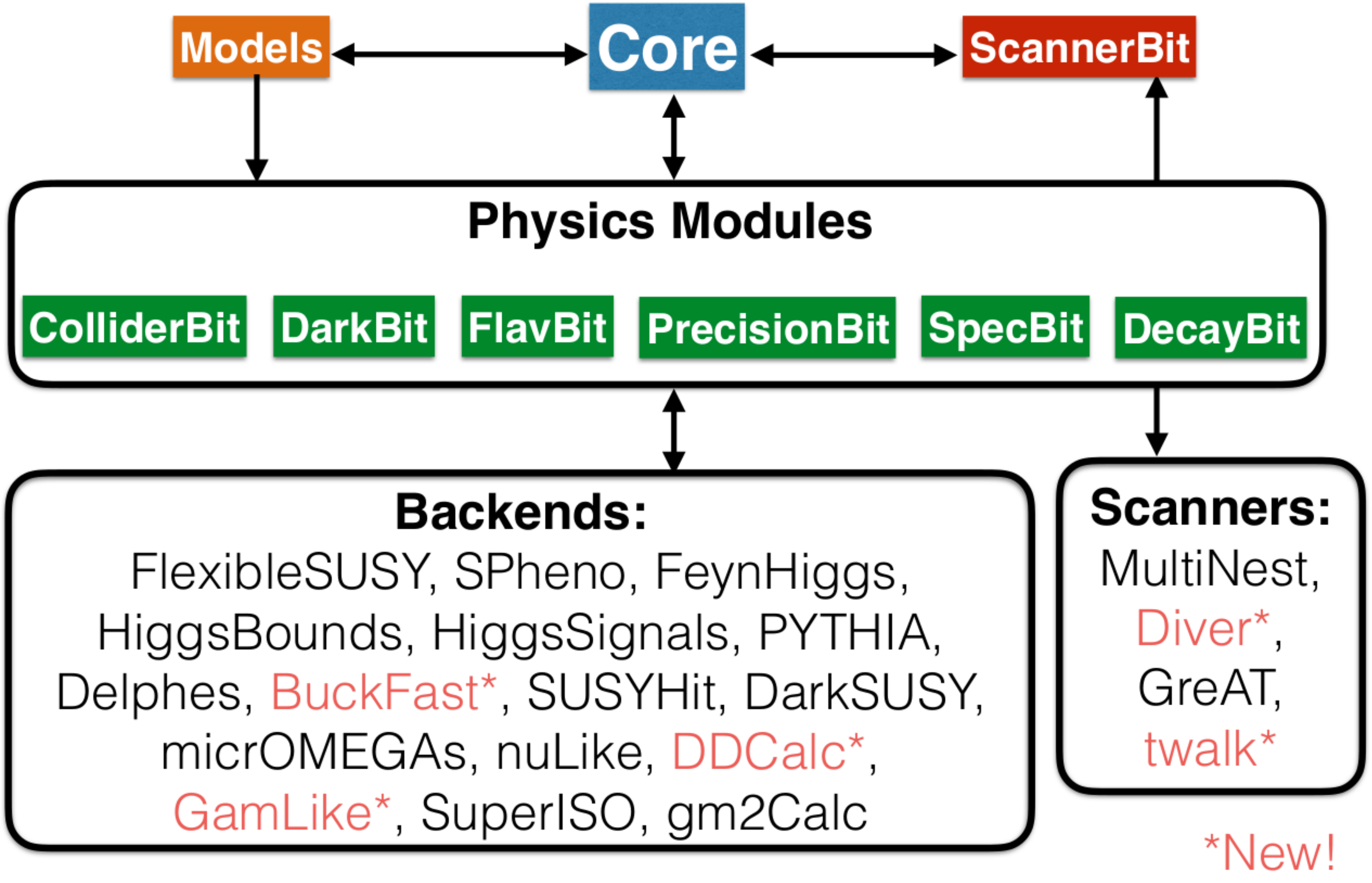


- **PrecisionBit:** precision EW tests (mostly via interface to FeynHiggs or SUSY-POPE)
- **ScannerBit:** manages stats, sampling and optimisation

What's in a module?

- Module functions (actual bits of GAMBIT C++ code)
- These can depend on other module functions
- Or can they can depend on *backends*(external codes)
- Adding new things is ***easy*** (detailed manual)
- Hooking up new backends or swapping them is ***easy***
- Module functions are **tagged** according to what they can calculate $\rightarrow$ plug and play!

GAMBIT code structure



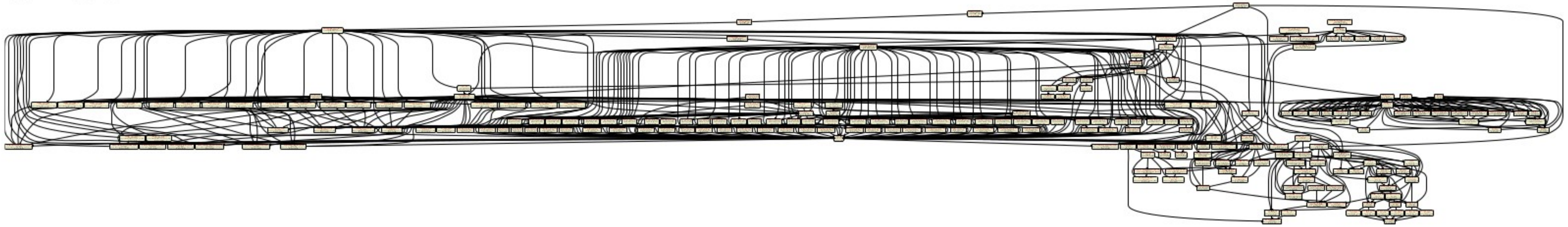
How does GAMBIT work?

- You specify what to calculate and how (yaml input file)
- GAMBIT checks to see which functions can do it
- A dependency resolver stitches things together in the right order, and calculations are also ordered by speed
- GAMBIT performs the scan and writes output
- Pippi makes the plots
- You(r student) write(s) the paper

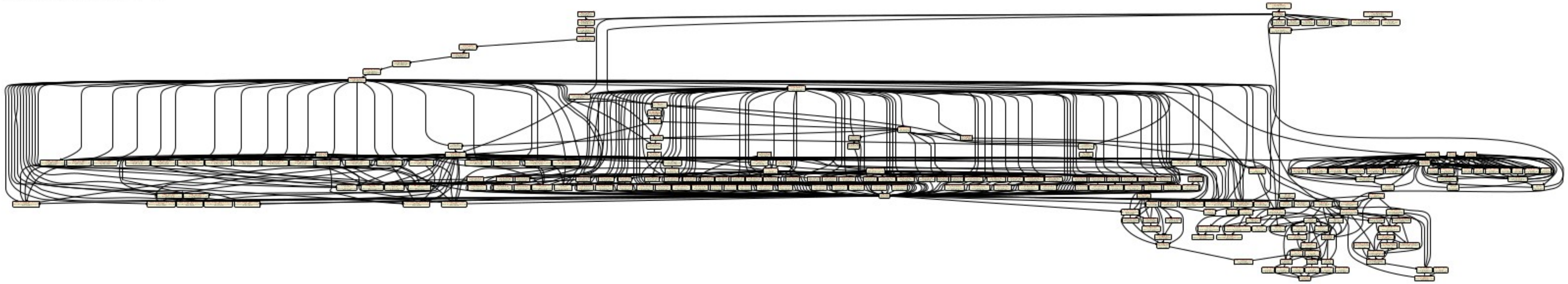
Cornell University Library
arXiv.org

Dependency resolution in action

CMSSM:



MSSM7:



A peek inside a yaml file (more later...)

specify parameters, ranges, priors

select the scanner

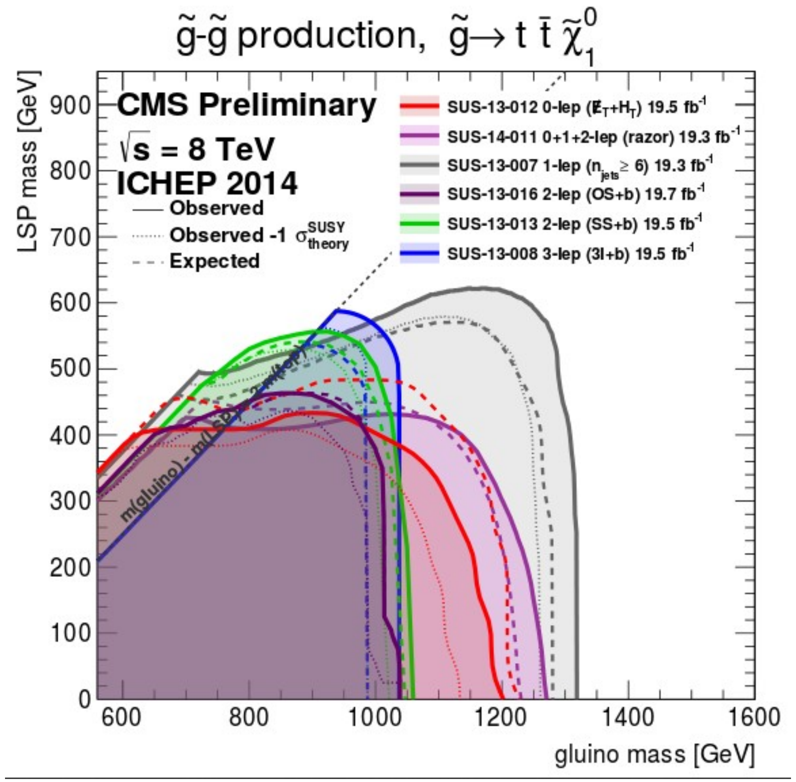
select likelihood components and other observables to calculate

define generic rules for how to fill dependencies

define generic rules for options to be passed to module functions

```yaml
Parameters:
  StandardModel_SLHA2: !import StandardModel_SLHA2_default
  MSSM25atQ: !import LesHouches.in.MSSM_1.yaml

Priors:
    # none: all parameters fixed in this example.

Scanner:

  use_scanner: toy_mcmc

  scanners:
    toy_mcmc:
      plugin: toy_mcmc
      point_number: 2000
      output_file:  output
      like:  Likelihood

ObsLikes:

  # Test DecayBit
  - purpose:      Test
    capability:   decay_rates
    type:         DecayTable

  # 79-string IceCube likelihood
  - capability: IceCube_likelihood
    purpose: Likelihood
    function: IC79_loglike

Rules:

  - capability: MSSM_spectrum
    function: get_MSSMatQ_spectrum
    options:
      invalid_point_fatal: true
```

LHC limits: the problem



ColliderBit

- Handles LHC and LEP limits
- LEP: complete recast of sparticle xsec limits
- SUSY & Exotic LHC search limits from real-time MC simulation
- LHC resonance search limits from HiggsBounds+HiggsSignals
- Future: new resonance limits (beyond NW?)
- Future: interface to Rivet for LHC analyses

Model independent LHC limits

- Custom parallelised Pythia MC + custom detector sim
- Can generate 20,000 events on 12 cores in < 5 s
- Then apply Poisson likelihood with nuisance parameters for systematics
- Combine analyses using best expected exclusion
- The best you can do without extra public info from the experiments. CMS are getting better at this:

  https://cds.cern.ch/record/2242860/files/NOTE2017_001.pdf

ColliderBit likelihood

- We use a Poissonian likelihood marginalized over a rescaling parameter ξ to account for systematic uncertainties:

$$\mathcal{L}(n|s,b) = \int_0^\infty \frac{[\xi(s+b)]^n \, e^{-\xi(b+s)}}{n!} P(\xi) \mathrm{d}\xi$$

$$P(\xi|\sigma_\xi) \approx \frac{1}{\sqrt{2\pi}\sigma_\xi} \frac{1}{\xi} \exp\left[-\frac{1}{2}\left(\frac{\ln \xi}{\sigma_\xi}\right)^2\right]$$

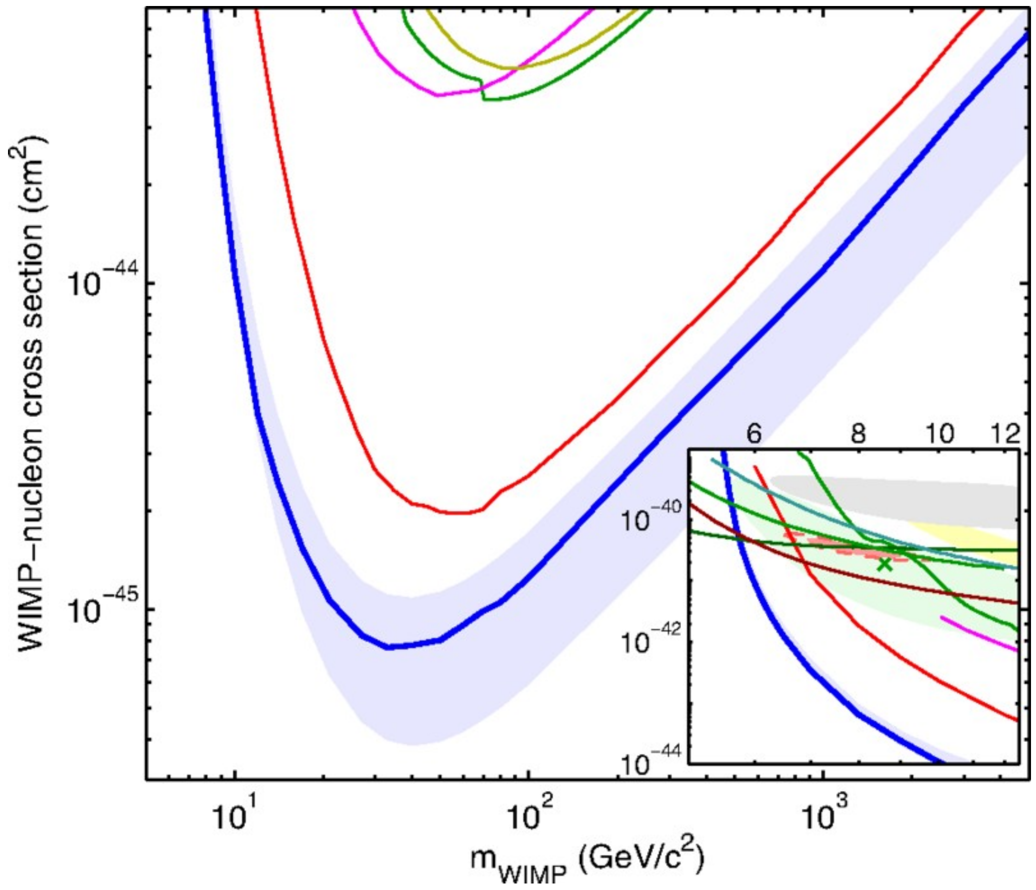
where

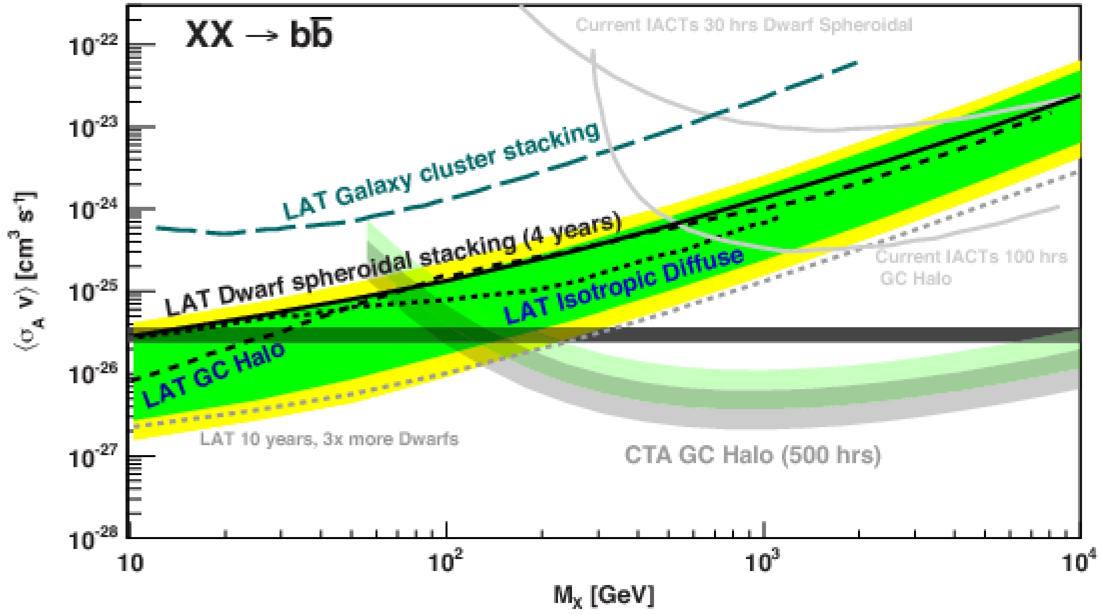
$$\sigma_\xi^2 = \sigma_s^2 + \sigma_b^2$$

- *n*, *s* and *b* are for signal region *expected* to give the strongest limit
- Currently available analyses (all 8 TeV):
  - ATLAS SUSY searches (0 lepton*, 0-1-2 lepton stop, b jets + MET, 2 lepton EW, 3 lepton EW)
  - CMS DM searches (top pair + MET, mono-b, mono-jet)
  - CMS multilepton SUSY search

*13 TeV also available

Astro limits: the problem

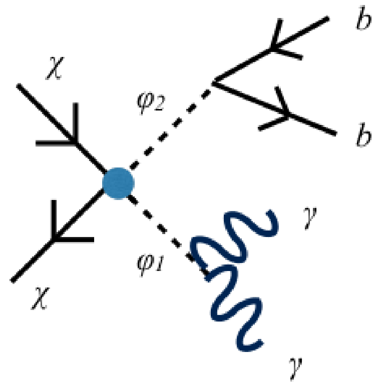


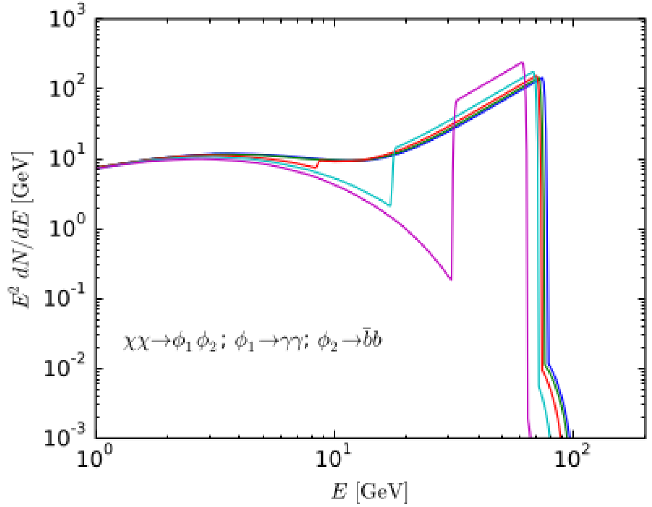


DarkBit: indirect detection

Gamma rays:

- Theoretical spectra calculated using branching fractions and tabulated gamma-ray yields
- Non-SM final state particles and Higgs are decayed on the fly with cascade Monte Carlo
- gamLike (gamlike.hepforge.org): New standalone code with likelihoods for DM searches from Fermi-LAT (dwarf spheroidals, galactic centre) and H.E.S.S. (galactic halo)



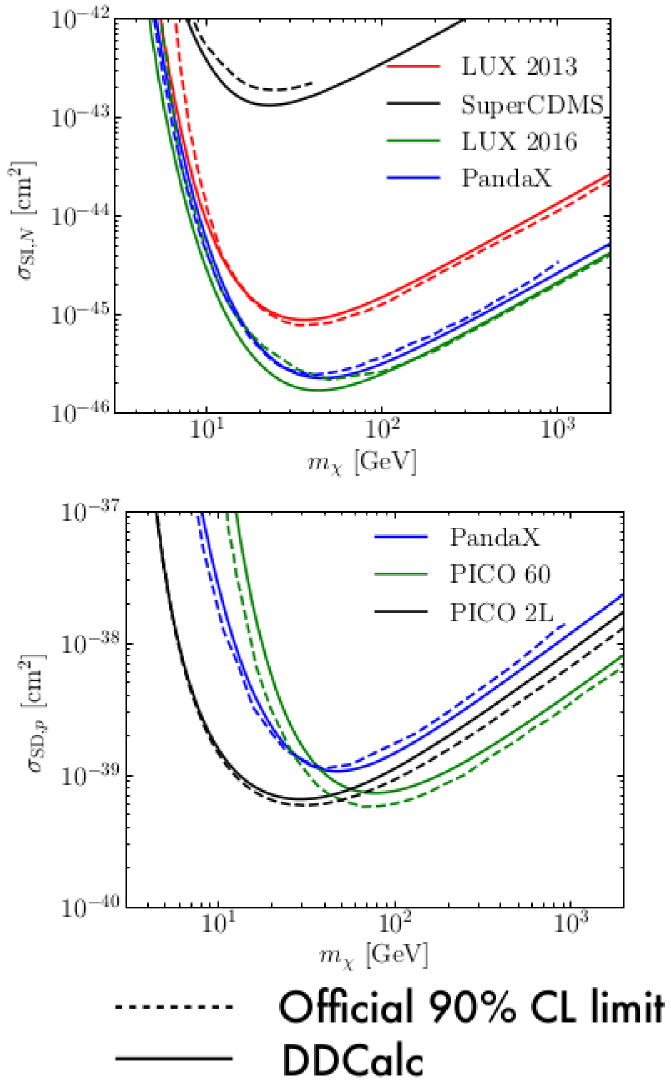


Solar neutrinos:

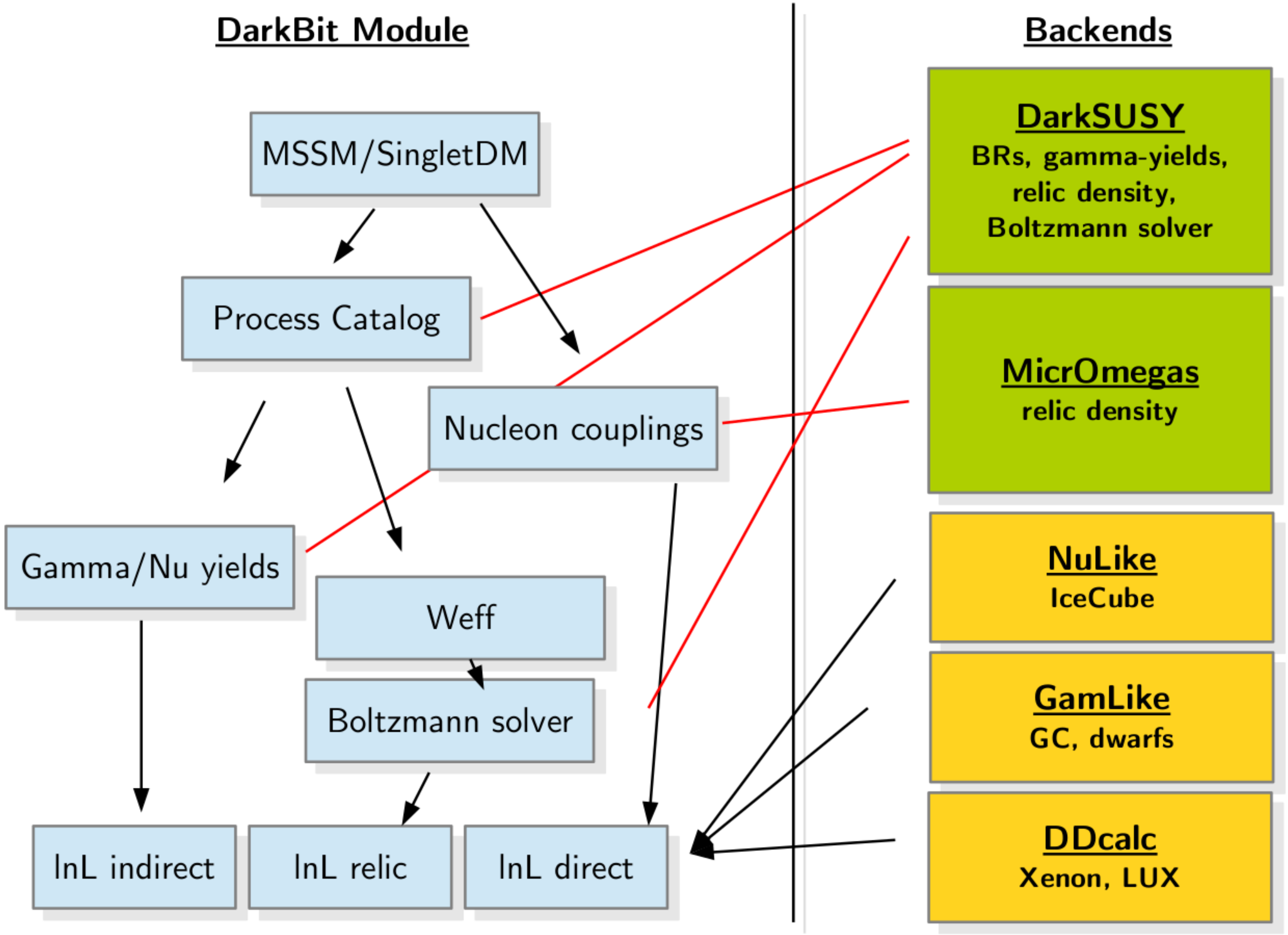
- Yields from DM annihilation in sun calculated by DarkSUSY. IceCube likelihoods contained in nulike (nulike.hepforge.org) standalone code.

DarkBit: direct detection

- In parallel with GAMBIT, we introduce *DDCalc* (ddcalc.hepforge.org), a tool to calculate event rates and complete likelihood functions for direct detection experiments taking into account:
  - A mix of both spin-independent and dependent contributions to the scattering rate.
  - Halo parameters (local density, DM velocity dispersion, etc.) chosen by the user.
- We currently have implemented likelihoods for Xenon(1T, 100), LUX, PandaX, SuperCDMS, PICO(60, 2L), and SIMPLE



DarkBit



- Event level neutrino telescope and gamma ray likelihoods!
- First principles treatment of direct search limits → easily extendable to non-trivial operators
- Very large range of experiments included (includes future, e.g. CTA)

FlavBit

- Models a series of experimental flavour anomalies
- Theoretical predictions currently based on SuperIso
- Theoretical and experimental uncertainties are carefully considered for each observable (including correlations)

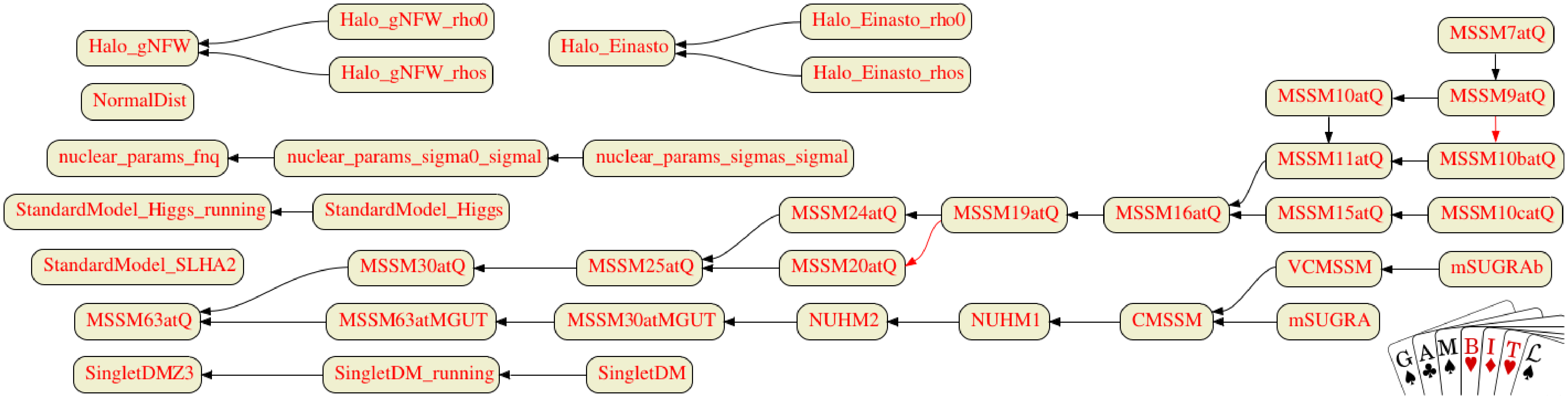
FlavBit likelihoods (more to come...)

- $B \to D^{(*)}\mu\nu$ (PDG, take correlated theoretical uncertainties from SuperIso study)
- $R_{D^{(*)}} \equiv \mathcal{B}(B \to D^{(*)}\tau\nu_\tau)/\mathcal{B}(B \to D^{(*)}\ell\nu_\ell)$ (HFAG, take correlated experimental uncertainties)
- $B^\pm \to \ell\nu_\ell$ (PDG value)
- $D^\pm \to \mu\nu_\mu$, $D_s^\pm \to \tau\nu_\tau$, $D_s^\pm \to \mu\nu_\mu$ (PDG values, theoretical correlations considered)
- Angular observables of $B^0 \to K^{*0}\mu^+\mu^-$ in q^2 bins (LHCb) (experimental correlations within each bin + theory correlations)
- $B_s^0 \to \mu^+\mu^-$ (latest LHCb result)
- $B^0 \to \mu^+\mu^-$ (combined LHCb and CMS)
- $B \to X_s\gamma$ for $E_\gamma > 1.6$ GeV (BaBar and Belle average by Misiak et al)
- ΔM_s: (average of CDF and LHCb, theory error is an order of magnitude greater than experimental uncertainty)

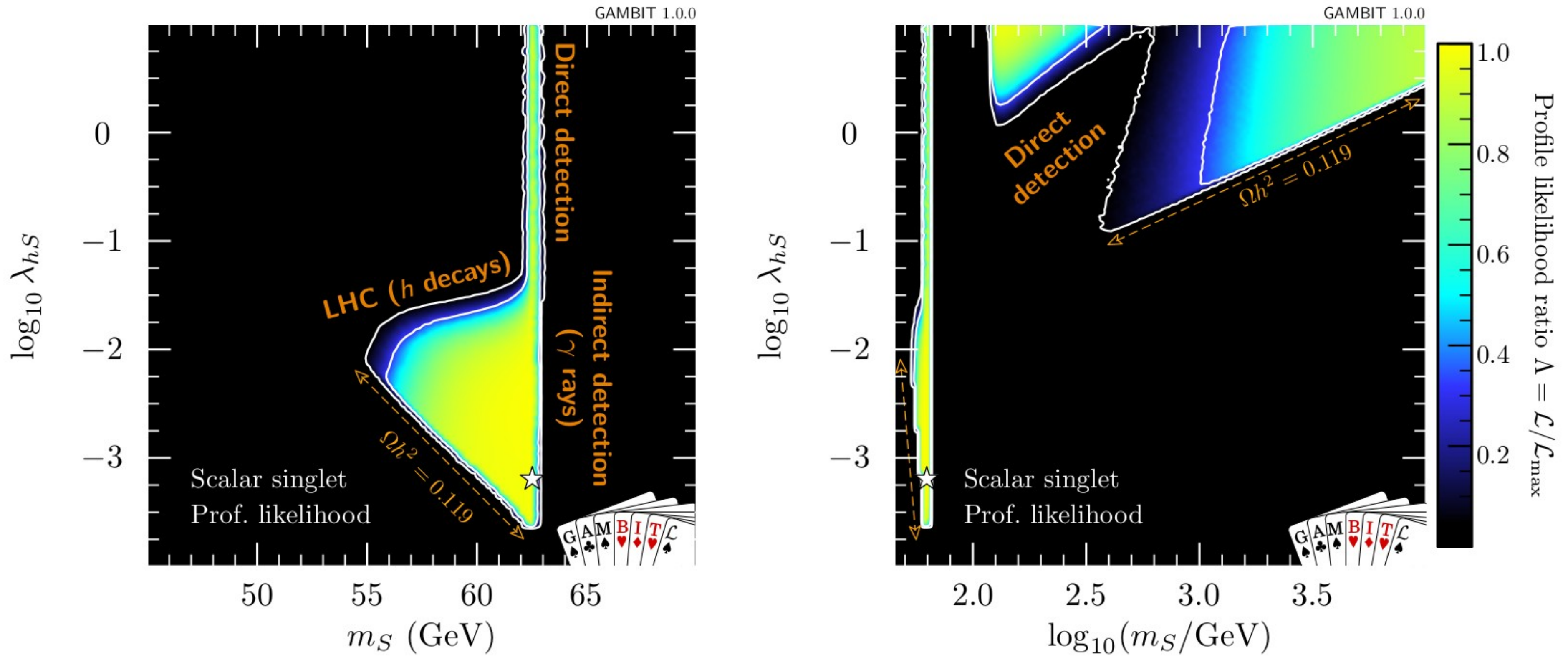
$$\frac{1}{\Gamma}\frac{\mathrm{d}^3(\Gamma+\bar\Gamma)}{\mathrm{d}\cos\theta_\ell\,\mathrm{d}\cos\theta_K\,\mathrm{d}\phi} = \frac{9}{32\pi}\Big[\frac{3}{4}(1-F_\mathrm{L})\sin^2\theta_K + F_\mathrm{L}\cos^2\theta_K + \frac{1}{4}(1-F_\mathrm{L})\sin^2\theta_K\cos 2\theta_\ell - F_\mathrm{L}\cos^2\theta_K\cos 2\theta_\ell + S_3\sin^2\theta_K\sin^2\theta_\ell\cos 2\phi + S_4\sin 2\theta_K\sin 2\theta_\ell\cos\phi + S_5\sin 2\theta_K\sin\theta_\ell\cos\phi + \frac{4}{3}A_\mathrm{FB}\sin^2\theta_K\cos\theta_\ell + S_7\sin 2\theta_K\sin\theta_\ell\sin\phi + S_8\sin 2\theta_K\sin 2\theta_\ell\sin\phi + S_9\sin^2\theta_K\sin^2\theta_\ell\sin 2\phi\Big]$$

Global and Modular BSM

- Models are defined by their parameters and relations to each other
- Models can inherit from parent models, easy translation between relations
- We have so far scanned SUSY + Higgs portal + axion + two Higgs doublet models



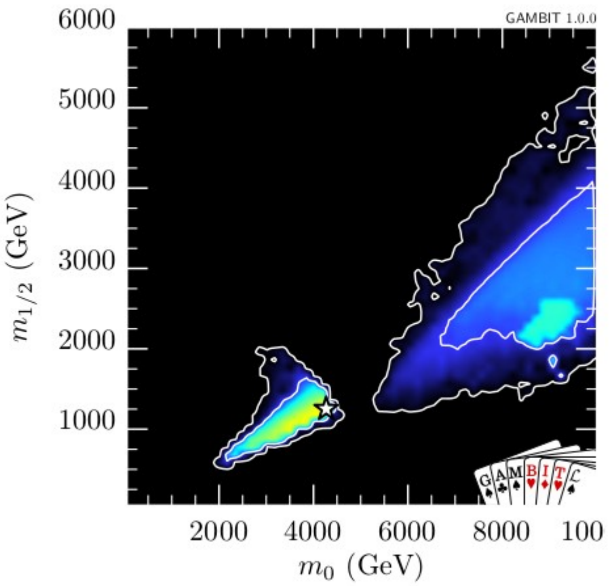
Global and Modular BSM Inference: Scalar singlet DM

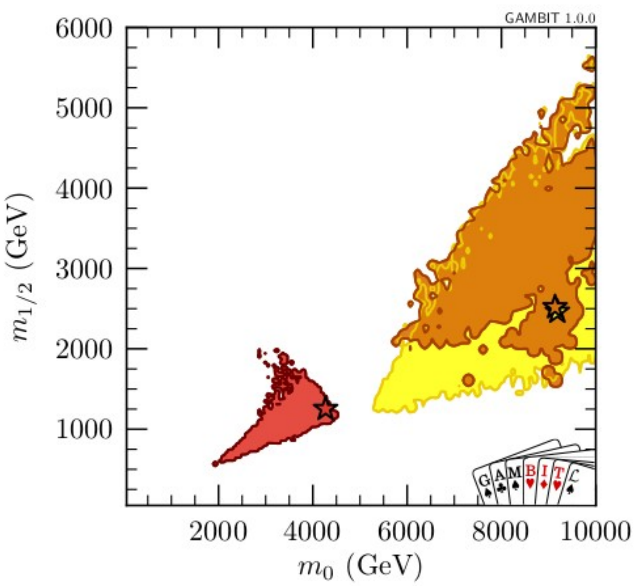


$$\mathcal{L} = \frac{1}{2}\mu_S^2 S^2 + \frac{1}{2}\lambda_{hS} S^2 |H|^2 + \frac{1}{4}\lambda_S S^4 + \frac{1}{2}\partial_\mu S \partial^\mu S$$

(m_s, λ_{hS} + 13 nuisances)

Global and Modular BSM Inference: CMSSM

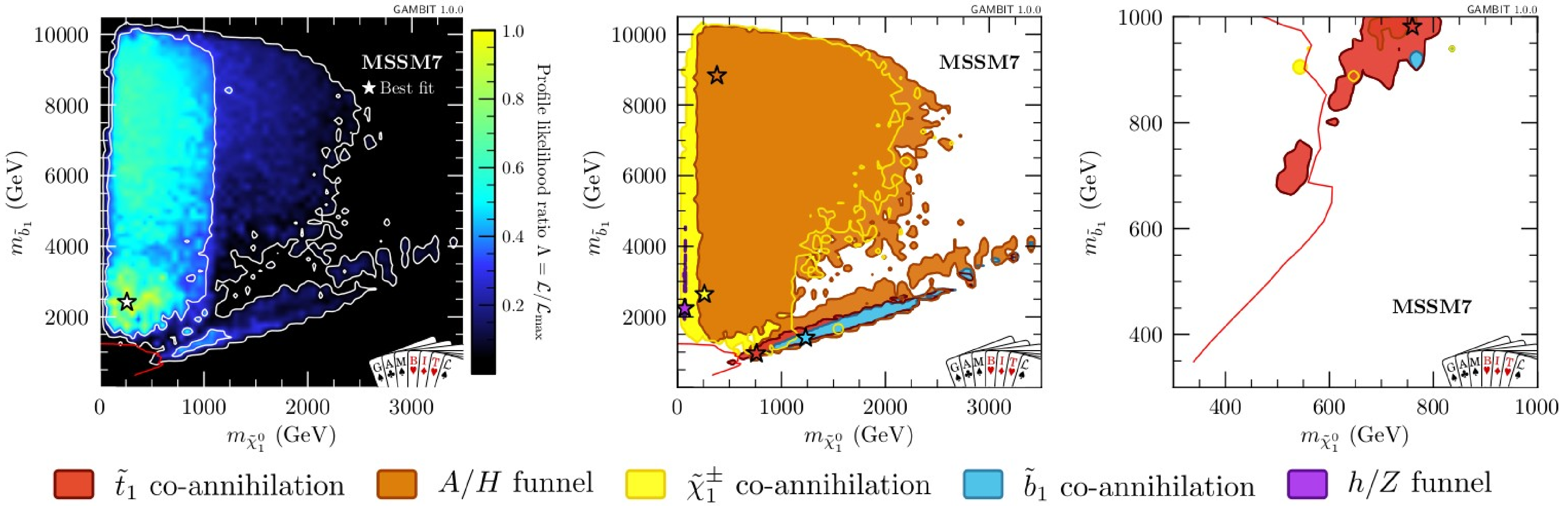




- m_0, $m_{\frac{1}{2}}$, A_0, $\tan\beta$ + 5 nuisances
- H/A^0 funnel, $\chi^{\pm}$ co-annihilation, $\tilde{t}$ co-annihilation
- **$\tilde{\tau}$ co-annihilation now ruled out**
- Includes LUX 2016, Panda-X + direct simulation of LHC Run 1 & Run 2 limits.

(also have NUHM1 and NUHM2 results)

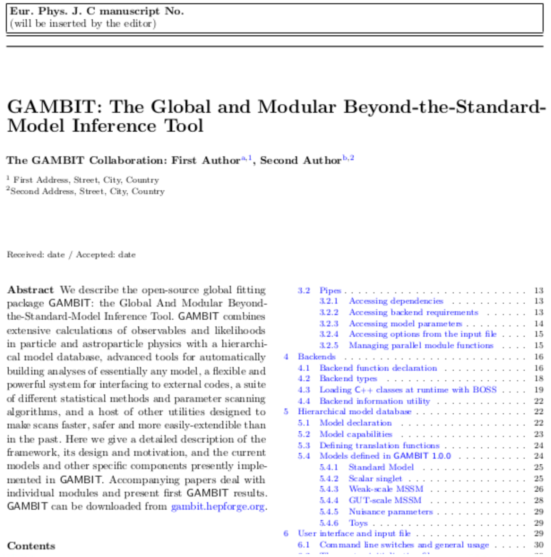
Global and Modular BSM Inference: MSSM7



- $m_{\tilde{f}}$, M_2, A_u, A_d, m_{Hu}, m_{Hd}, $\tan\beta$ + 5 nuisances
- H/A^0 funnel, h/Z funnel, $\chi^{\pm}$ co-annihilation, $\tilde{t}/\tilde{b}$ co-annihilation
- Includes LUX 2016, Panda-X + direct simulation of LHC Run 1 & Run 2 limits.

Global and Modular BSM Inference Tool

- GAMBIT has just been released as an ***open source public tool***
- 9 papers published in EPJC (design, manual + first physics results)
- Feature article in *Physics World* March 2017 issue if you want a gentler introduction
- See **gambit.hepforge.org** for more info



Eur. Phys. J. C manuscript No.
(will be inserted by the editor)

GAMBIT: The Global and Modular Beyond-the-Standard-Model Inference Tool

The GAMBIT Collaboration: First Author[a,1], Second Author[b,2]

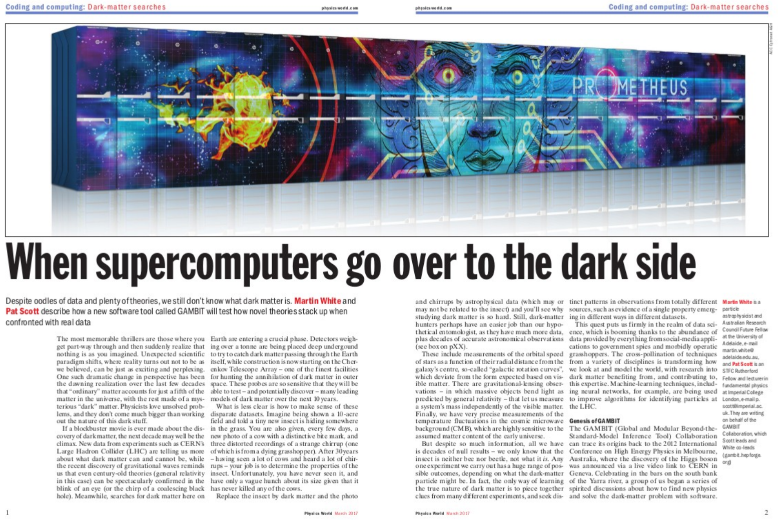
[1] First Address, Street, City, Country
[2] Second Address, Street, City, Country

Received: date / Accepted: date

Abstract We describe the open-source global fitting package GAMBIT: the Global And Modular Beyond-the-Standard-Model Inference Tool. GAMBIT combines extensive calculations of observables and likelihoods in particle and astroparticle physics with a hierarchical model database, advanced tools for automatically building analyses of essentially any model, a flexible and powerful system for interfacing to external codes, a suite of different statistical methods and parameter scanning algorithms, and a host of other utilities designed to make scans faster, safer and more easily-extendible than in the past. Here we give a detailed description of the framework, its design and motivation, and the current models and other specific components presently implemented in GAMBIT. Accompanying papers deal with individual modules and present first GAMBIT results. GAMBIT can be downloaded from gambit.hepforge.org.

Contents

3.2 Pipes . . . 13
3.2.1 Accessing dependencies . . . 13
3.2.2 Accessing backend requirements . . . 13
3.2.3 Accessing model parameters . . . 14
3.2.4 Accessing options from the input file . . . 15
3.2.5 Managing parallel module functions . . . 15
4 Backends . . . 16
4.1 Backend function declaration . . . 16
4.2 Backend types . . . 18
4.3 Loading C++ classes at runtime with BOSS . . . 19
4.4 Backend information utility . . . 22
5 Hierarchical model database . . . 22
5.1 Model declaration . . . 22
5.2 Model capabilities . . . 23
5.3 Defining translation functions . . . 24
5.4 Models defined in GAMBIT 1.0.0 . . . 24
5.4.1 Standard Model . . . 25
5.4.2 Scalar singlet . . . 25
5.4.3 Weak-scale MSSM . . . 26
5.4.4 GUT-scale MSSM . . . 28
5.4.5 Nuisance parameters . . . 29
5.4.6 Toys . . . 29
6 User interface and input file . . . 29
6.1 Command line switches and general usage . . . 30



Coding and computing: Dark-matter searches — physicsworld.com

When supercomputers go over to the dark side

Despite oodles of data and plenty of theories, we still don't know what dark matter is. Martin White and Pat Scott describe how a new software tool called GAMBIT will test how novel theories stack up when confronted with real data

What’s next for Gambit?

- More models: 2HDM, axions, RH neutrinos, …
- More ColliderBit analyses, i.e. 13 TeV coverage [done, but private]
- Simplified likelihood anayses [done, private]
- Improved event simulation: CalcHEP, MadGraph, NLO total cross-sections
- New modules for cosmology and neutrino physics
- Gambit Universal Models (GUM): Mathematica → likelihoods

Tutorial exercises

Two hands-on tutorials to introduce you to the basic features of collider-oriented GAMBIT fits, and the user interface

- Flavour physics Wilson coefficient fit
- Single-point or scan of CMSSM parameters

Installation should already be in hand, via Docker.
But in case we changed something, update again:

- docker pull agbuckley/gambit-tutorial
  docker run -it --rm agbuckley/gambit-tutorial

Tutorial 1: Wilson Coefficients

$$\mathcal{H}_{\text{eff}} = -\frac{4G_F}{\sqrt{2}} V_{tb}V_{ts}^* \sum_{i=1}^{10} \left(C_i(\mu)\mathcal{O}_i(\mu) + C_i'(\mu)\mathcal{O}_i'(\mu) \right)$$

Wilson Coefficients

$$\mathcal{O}_1 = (\bar{s}\gamma_\mu T^a P_L c)(\bar{c}\gamma^\mu T^a P_L b)$$

$$\mathcal{O}_2 = (\bar{s}\gamma_\mu P_L c)(\bar{c}\gamma^\mu P_L b)$$

$$\mathcal{O}_3 = (\bar{s}\gamma_\mu P_L b)\sum_q(\bar{q}\gamma^\mu q)$$

$$\mathcal{O}_4 = (\bar{s}\gamma_\mu T^a P_L b)\sum_q(\bar{q}\gamma^\mu T^a q)$$

$$\mathcal{O}_5 = (\bar{s}\gamma_{\mu_1}\gamma_{\mu_2}\gamma_{\mu_3} P_L b)\sum_q(\bar{q}\gamma^{\mu_1}\gamma^{\mu_2}\gamma^{\mu_3} q)$$

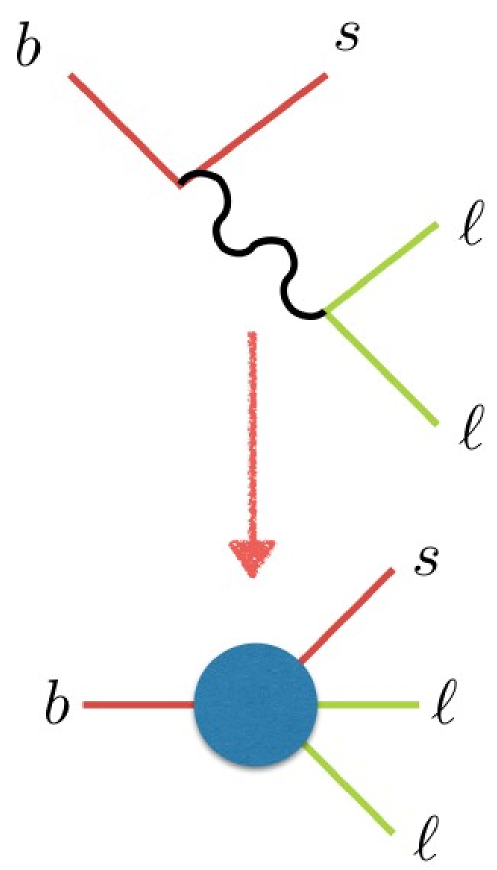
$$\mathcal{O}_6 = (\bar{s}\gamma_{\mu_1}\gamma_{\mu_2}\gamma_{\mu_3} T^a P_L b)\sum_q(\bar{q}\gamma^{\mu_1}\gamma^{\mu_2}\gamma^{\mu_3} T^a q)$$

$$\mathcal{O}_7 = \frac{e}{(4\pi)^2} m_b (\bar{s}\sigma^{\mu\nu} P_R b) F_{\mu\nu}$$

$$\mathcal{O}_8 = \frac{g}{(4\pi)^2} m_b (\bar{s}\sigma^{\mu\nu} T^a P_R b) G^a_{\mu\nu}$$

$$\mathcal{O}_9 = \frac{e^2}{(4\pi)^2} (\bar{s}\gamma^\mu P_L b)(\bar{\ell}\gamma_\mu \ell)$$

$$\mathcal{O}_{10} = \frac{e^2}{(4\pi)^2} (\bar{s}\gamma^\mu P_L b)(\bar{\ell}\gamma_\mu\gamma_5 \ell)$$



See arxiv.org/pdf/1705.07933.pdf for more detail

Tutorial 1: Wilson Coefficients

- 2D Wilson coefficient fit

$$\Delta C_x \equiv C_{x,BSM} - C_{x,SM}$$

- Free parameters: ΔC_7 `Re_DeltaC7`

  ΔC_{10} `Re_DeltaC10`

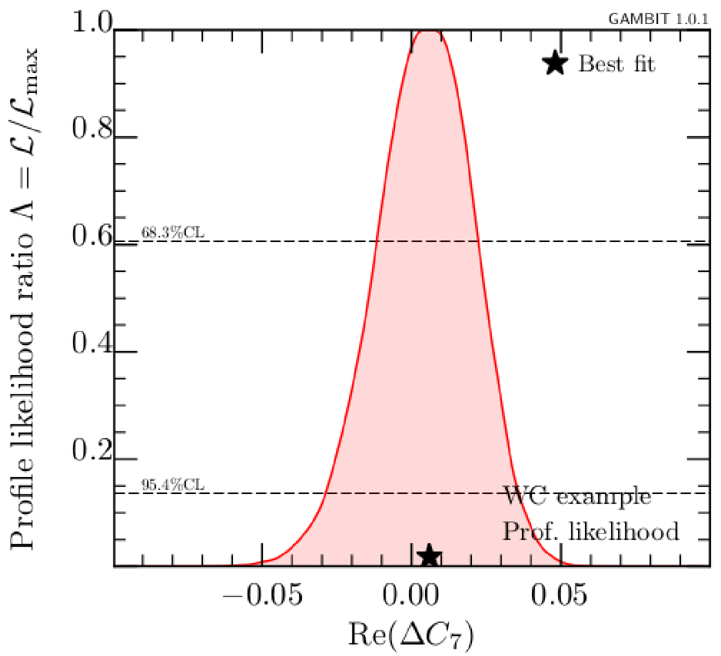
- Observables: $BR(B \to X_s \gamma)$ `b2sgamma`

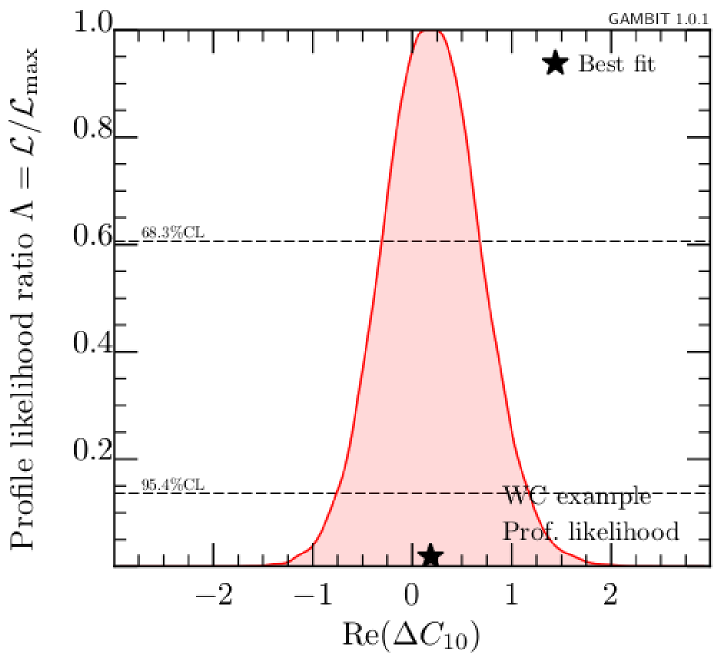
  $BR(B_d \to \mu^+ \mu^-)$

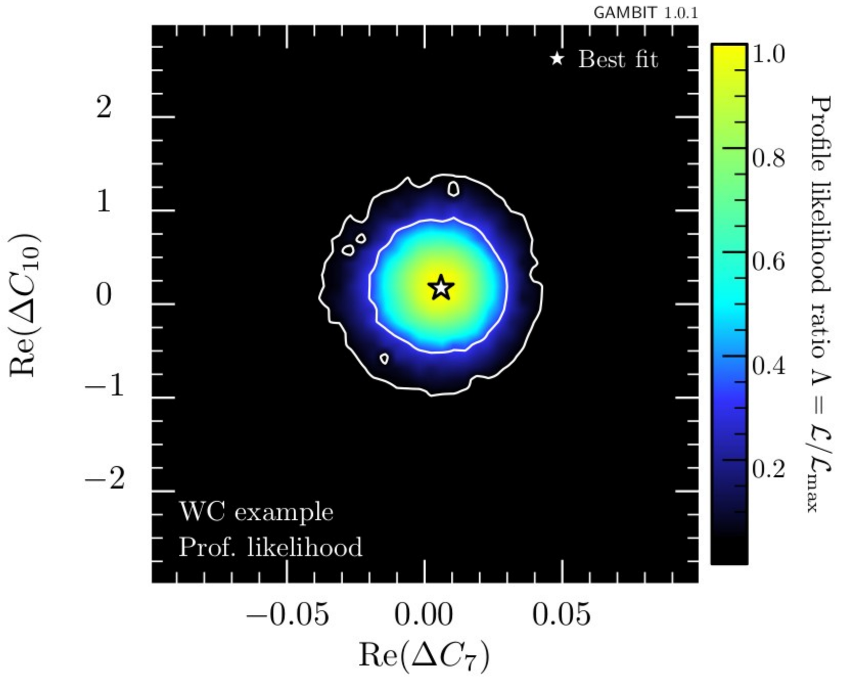
  $BR(B_s \to \mu^+ \mu^-)$ `b2ll`

- Follow the steps in: WC_tutorial_commands.txt

Tutorial 1: Results should look like this







- Feel free to ask for assistance with plotting and interpreting the *.pip file

Tutorial 2: CMSSM file

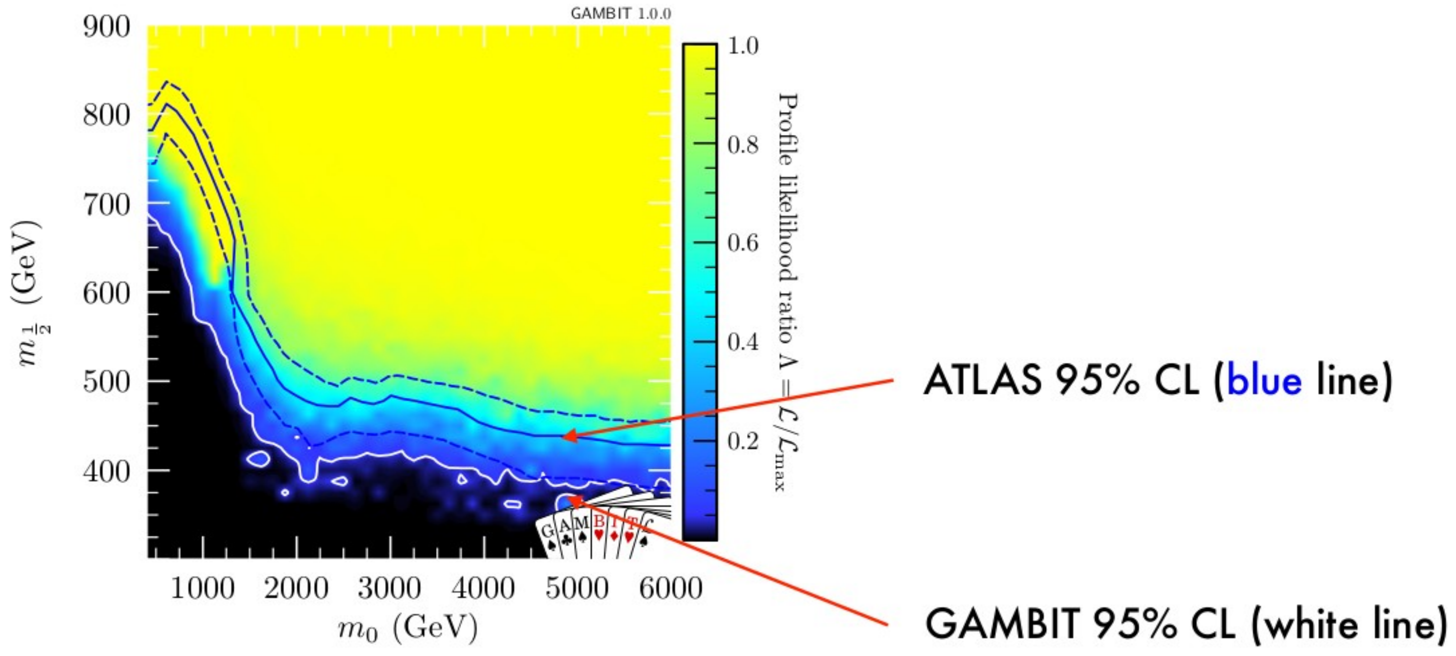
- Put the file ColliderBit_CMSSM_tutorial.yaml in your GAMBIT directory
- Run it using:

```
./gambit -f ColliderBit_CMSSM_tutorial.yaml
```

- After a few minutes, GAMBIT will spit out a likelihood value the ATLAS 8 TeV 0 lepton analysis, compare with:

95% CL exclusion →

$$\Delta \ln \mathcal{L} = -3.0$$



Tutorial 2: Questions

- How can you change the number of generated events?
- How can you add more LHC analyses?
- How can you run both ATLAS and CMS analyses?
- How would you scan the CMSSM rather than run one point?
- How would you go about scanning the MSSM rather than the CMSSM?
- How would you add astrophysical likelihoods?

See the yaml_files/ directory for hints and examples