WHIZARD

Wolfgang Kilian

University of Siegen

MC4BSM, IHEP, Beijing
July 2016

WHIZARD: Overview

Scope

WHIZARD is a stand-alone program for elementary processes at high-energy colliders: scattering and particle decays

- integrate cross sections and decay widths (perturbative, partonic)
- amplitude code is generated and executed on the fly
- calculate observables, generate event samples
- hadronization via external code

Context

Program exists since 1999

Active authors and developers:

WK (U Siegen), T. Ohl (Würzburg), J. Reuter (DESY)

S. Brass, B. Chokoufe, C. Fleper, M. Sekulla, S. Shim, C. Weiss, Z. Zhao

Current version:

2.2.8 ($\Rightarrow$ 2.3.0)

For the User

- Installation centrally or locally on any Linux or Mac (`autotools`)

```
./configure [options]
make
make install
```

- Look-and-feel like standard Linux/Unix apps

```
./whizard [options] FILE
```

- User works in directories at his choice, no structure imposed

Programming Languages

- Main program code is written in modern Fortran (F2008).
  Compilers: `gfortran`, `ifort`, `nagfor`
- Glue code uses `Make`, Makefiles generated on-the-fly
- Algebra is done in `OCaml` (compiler free for all platforms)
  $\Rightarrow$ Sub-package `OMega`
- Some interface to external code in `C/C++`
- Can use precompiled libraries (`LHAPDF`, `HepMC`, ...)

Input

- Models precompiled or generated (FeynRules, SARAH, [UFO])
- User provides all other information via input file, which is actually a script
  - Model selection, data, parameters, process setup
  - Compute observables for in/out particles: cuts, weight for integration
  - Integrate once, or loop and scan over parameters
  - Generate and write event samples, select format
  - Analyze event samples and plot results

  Programming language (DSL): Sindarin

Output

1. Numbers (cross sections, decay widths, other observables)
2. Distributions (plots)
3. Event samples in various formats
   (fixed-order or showered, partonic or hadronic)

Variants

- Docker package (VM) with all prerequisites
- Library version (C-compatible API) for inclusion in other programs
- Interactive mode for simple tasks
- GUI (`Javascript/node.js`) as server/client model version via Web browser

  $\Rightarrow$ end of this talk

WHIZARD Details: Collider Setup

Initial state: just look at process definition

```
process foo = "e+", "e-" ⇒ "mu+", "mu-"
```

. . . or write explicitly

Decay = single beam

Scattering = pair of beams

Explicit definition is useful if there is some beam structure

Decay:

```
beams_pol_density = @(-1)
beams = Z
```

Scattering:

```
$lhapdf_file = "CT14nlo"
beams = p, p => lhapdf
```

Supported: polarization, momentum, angle, any (meaningful) convolution of beam spectra and structure functions

e^+e^- Collider Beam Structure

ILC, CLIC, CEPC, FCC-ee

1. Beamstrahlung = classical beam-beam interaction, collective effect

2. ISR = resummed soft/collinear QED bremsstrahlung for initial-state electron-electron interaction

e^+e^- Collider: Beamstrahlung

Options for handling beamstrahlung (as supported by WHIZARD)

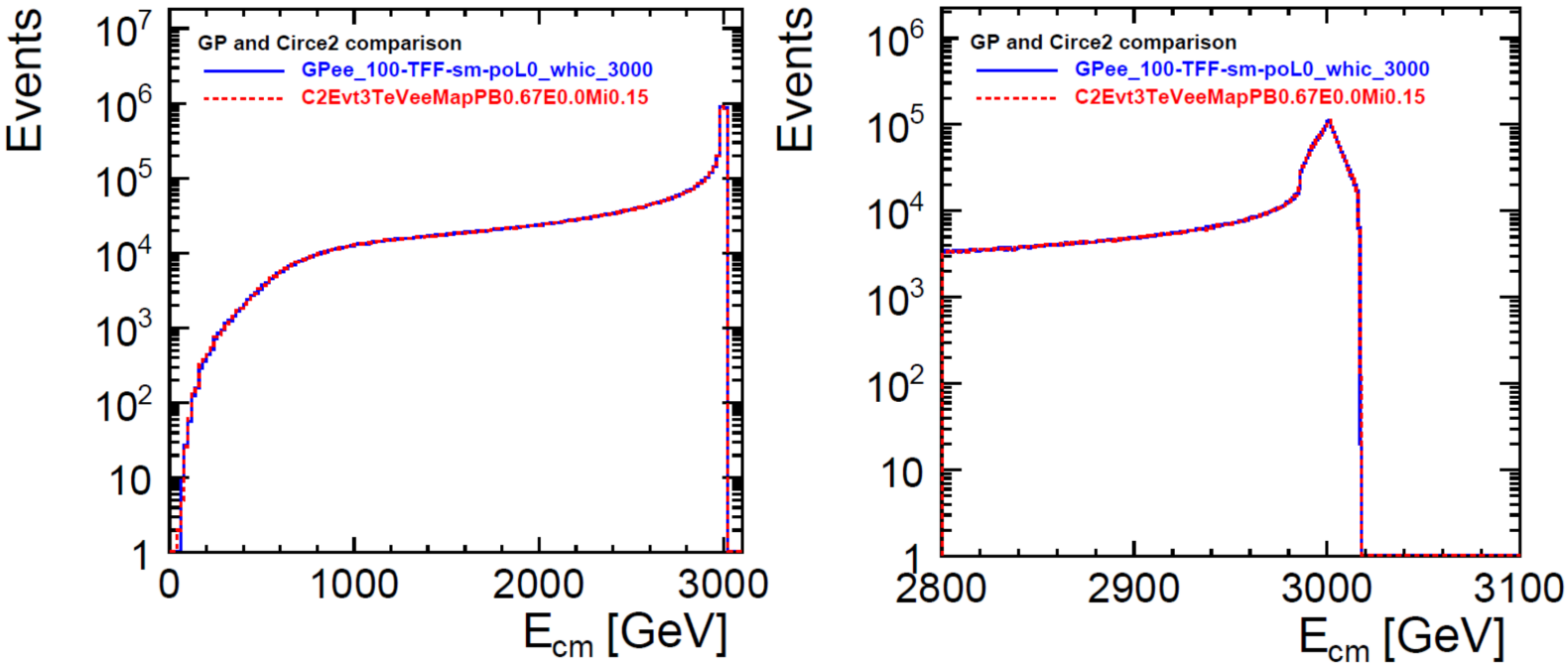
1. Beam-energy spread
   ⇒ Gaussian distribution

2. `CIRCE1`: parameterized beam spectrum
   ⇒ Factorized, few parameters in fit

3. Beam-events file: use result of GuineaPig simulation directly
   ⇒ Finite number of pre-simulated beam events

4. `CIRCE2`: generator takes binned and interpolated result of detailed GuineaPig simulation
   ⇒ Precise description

e e Beam Spectrum 3TeV



Left: dN/d√s distribution Guineapig events (blue) and Circe2 generated events using a power function mapping (red); 20 GeV bins.
Right: zoom into the high √s region; 1 GeV bins; looks good.
Check the distribution ratios.

MODEL TYPE	with CKM matrix	trivial CKM
QED with e, μ, τ, γ	–	`QED`
QCD with d, u, s, c, b, t, g	–	`QCD`
Standard Model	`SM_CKM`	`SM`
SM with anomalous gauge couplings	`SM_ac_CKM`	`SM_ac`
SM with anomalous top couplings	`SMtop_CKM`	`SMtop`
SM for e^+e^- top threshold	–	`SM_tt_threshold`
SM ext. for VV scattering	—	`SSC / SSC2`
SM ext. for unitarity limits in VV	—	`SM_ul`
SM with Higgs singlet	—	`HSExt`
2HDM	`2HDM_CKM`	`2HDM`
MSSM	`MSSM_CKM`	`MSSM`
MSSM with gravitinos	—	`MSSM_Grav`
NMSSM	`NMSSM_CKM`	`NMSSM`
extended SUSY models	—	`PS/E/SSM`
Littlest Higgs	—	`Littlest`
Littlest Higgs with ungauged $U(1)$	—	`Littlest_Eta`
Littlest Higgs with T parity	—	`Littlest_Tpar`
Simplest Little Higgs (anomaly-free)	—	`Simplest`
Simplest Little Higgs (universal)	—	`Simplest_univ`
3-site model	—	`Threeshl`
UED	—	`UED`
SM with Z'	—	`Zprime`
SM with gravitino and photino	—	`GravTest`
Augmentable SM template	—	`Template`

New additions to the Model list

- EFT for the SM: complete $D = 6$ Lagrangian (Warsaw Basis)

  S. Shim

- General UFO File Support (`OMega`): now available for MC4BSM

  T. Ohl

- Beyond EFT: electroweak interactions at high energy
  $\Rightarrow$ (Super)LHC/FCC-pp, ILC and CLIC

  C. Fleper, M. Sekulla

Electroweak Interations Beyond the SM

Genuine Higgs and Anomalous Gauge (Self-)Interactions

- Leading effects of New Physics expected in $D = 8$ operators

Energy dependence of EFT (with $D = 8$):

$\Rightarrow$ EFT must be extrapolated $\Rightarrow$ model of strong EW asymptotics

$\Rightarrow$ Meaningful only if unitarity is imposed: smooth transition

T-matrix unitarization scheme with off-shell continuation: not handled by standard (automatic) Feynman rules approach

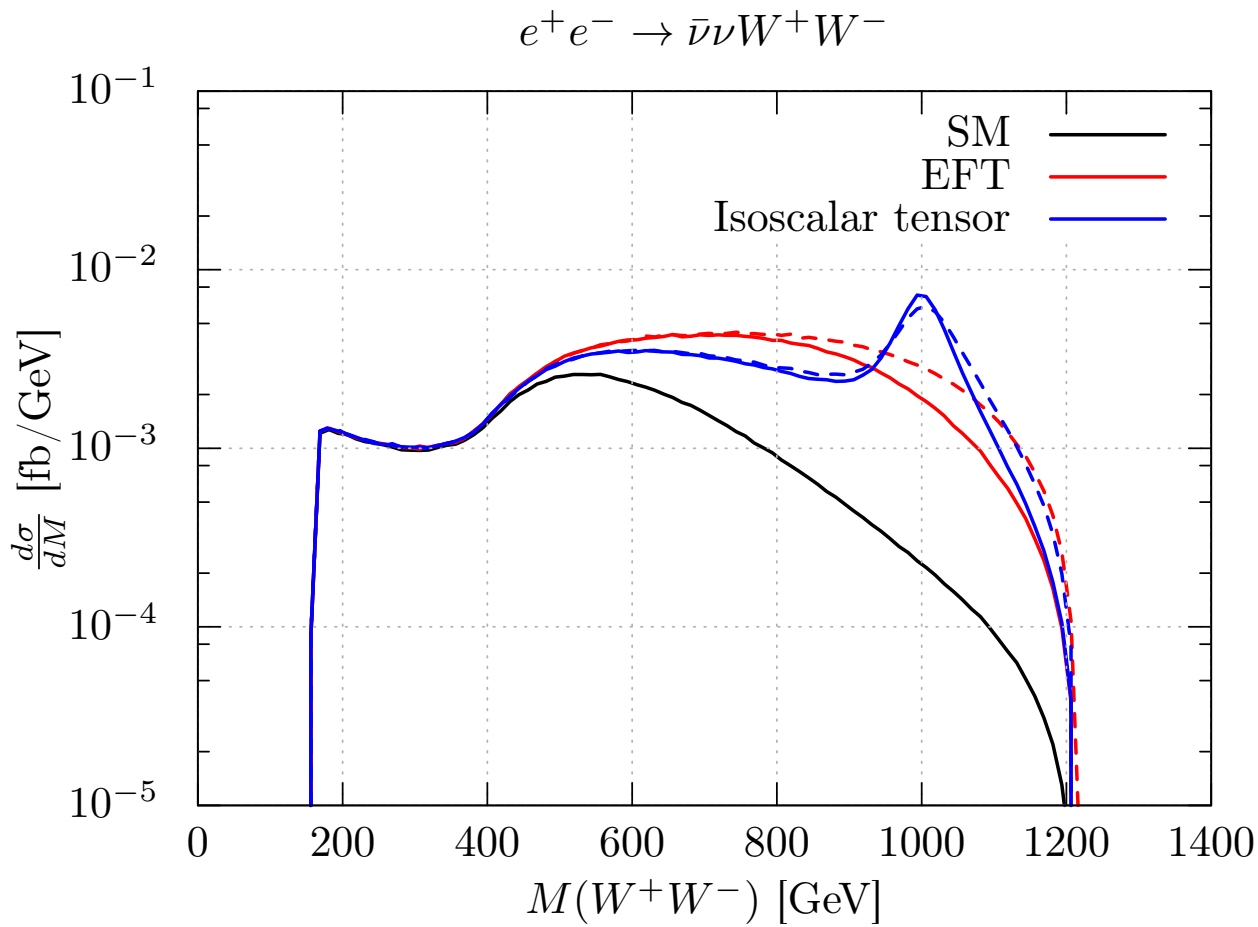
$\Rightarrow$ Implemented manually in WHIZARD model

Extrapolated EFT is just one strongly interacting model.

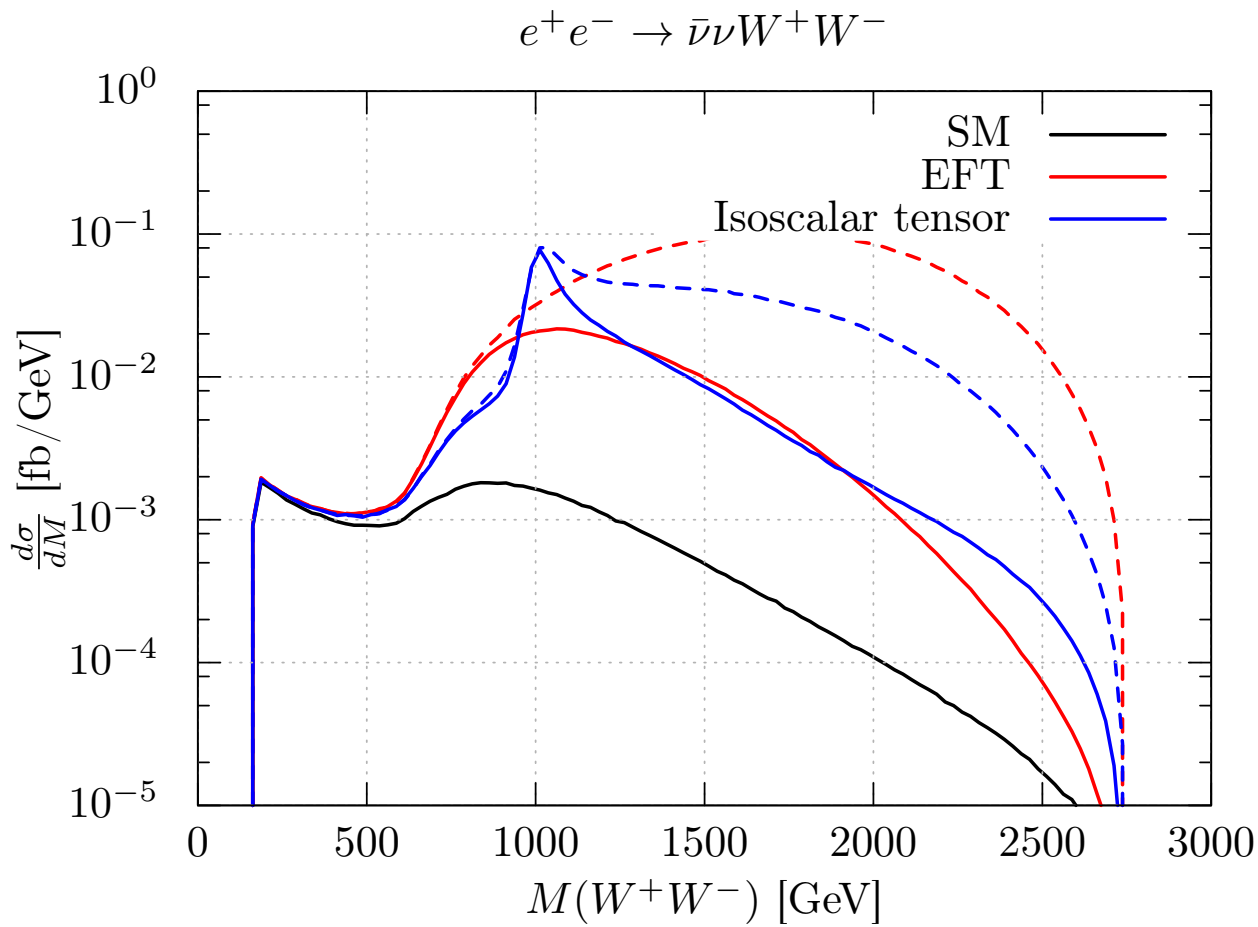
Add in More:

- $\Rightarrow$ Incorporate resonances in $VV \to VV$ topology ($S = 0, 2$)
- $\Rightarrow$ Spin-2 required some theoretical work to do it right (and generic)
- $\Rightarrow$ Includes weakly interacting models (2HDM, ...)
- $\Rightarrow$ Also requires unitarity restoration

CLIC @ 1.4 TeV: Resonance with $M = 1.0$ TeV



CLIC @ 3.0 TeV: Resonance with $M = 1.0$ TeV



Processes

```
process foobar =
  "e+", "e-" => nue, nuebar, b, bbar, q, qbar, "mu+", "mu-"
```

- Explicit, exclusive
- optionally constrained
- optionally with sum over flavors `alias q = u:d:s:c`
- optionally sum over subprocesses ⇒ inclusive, e.g. decays

Generator: `OMega` (automatic recursive tree-level amplitudes)

NLO ⇒ J. Reuter's talk

Phase Space and Integration

Phase Space

Multi-Channel: Select dominant singularity structures, parameterize via resonance mappings etc.

(Selection is automatic, heuristics for determining channel importance)

Integration

Twofold adaptive: VAMP

1. Each channel gets a weight which is iteratively adapted
2. Each channel is binned (VEGAS-like), binning is iteratively adapted

Warmup iterations yield integration grid and channel-weight distribution optimized for the particular process, to be used in event generation

Parallel Integration

OpenMP

Make use of multi-core processors: parallel evaluation of helicity configurations

MPI

Make use of multi-processor clusters: parallel sampling of phase space
MPI 3.0 asynchronous message passing
$\Rightarrow$ under validation S. Brass

Event Handlers

WHIZARD uses the twofold-adapted phase space to generate unweighted event samples. Further processing:

1. Particle decays (cascades):
   - using WHIZARD's own decay processes (explicit or automatic)
   - full control over polarization transfer (uncorrelated, classical, quantum correlation)
2. Shower and Matching $\Rightarrow$ Jürgen's talk
3. Hadronization: optionally call PYTHIA internally
4. Event output: file formats StdHEP, LHEF, HepMC, LCIO, ASCII

Example

```
model = QCD
alias q = u:d
alias Q = U:D
process twojet = g, g => q, Q

sqrts = 13 TeV
beams = p, p => pdf_builtin

cuts =
  all Pt > 100 GeV [q:Q]

simulate (twojet) {
   n_events = 10000
   sample_format = lhef
}
```

Plots, observables

```
histogram ptplot (0 GeV, 100 GeV, 5 GeV)

simulate (twojets) {
  n_events = 10000
  analysis =
    record ptplot (eval Pt [q])
}
compile_analysis
```

More Analysis

SINDARIN

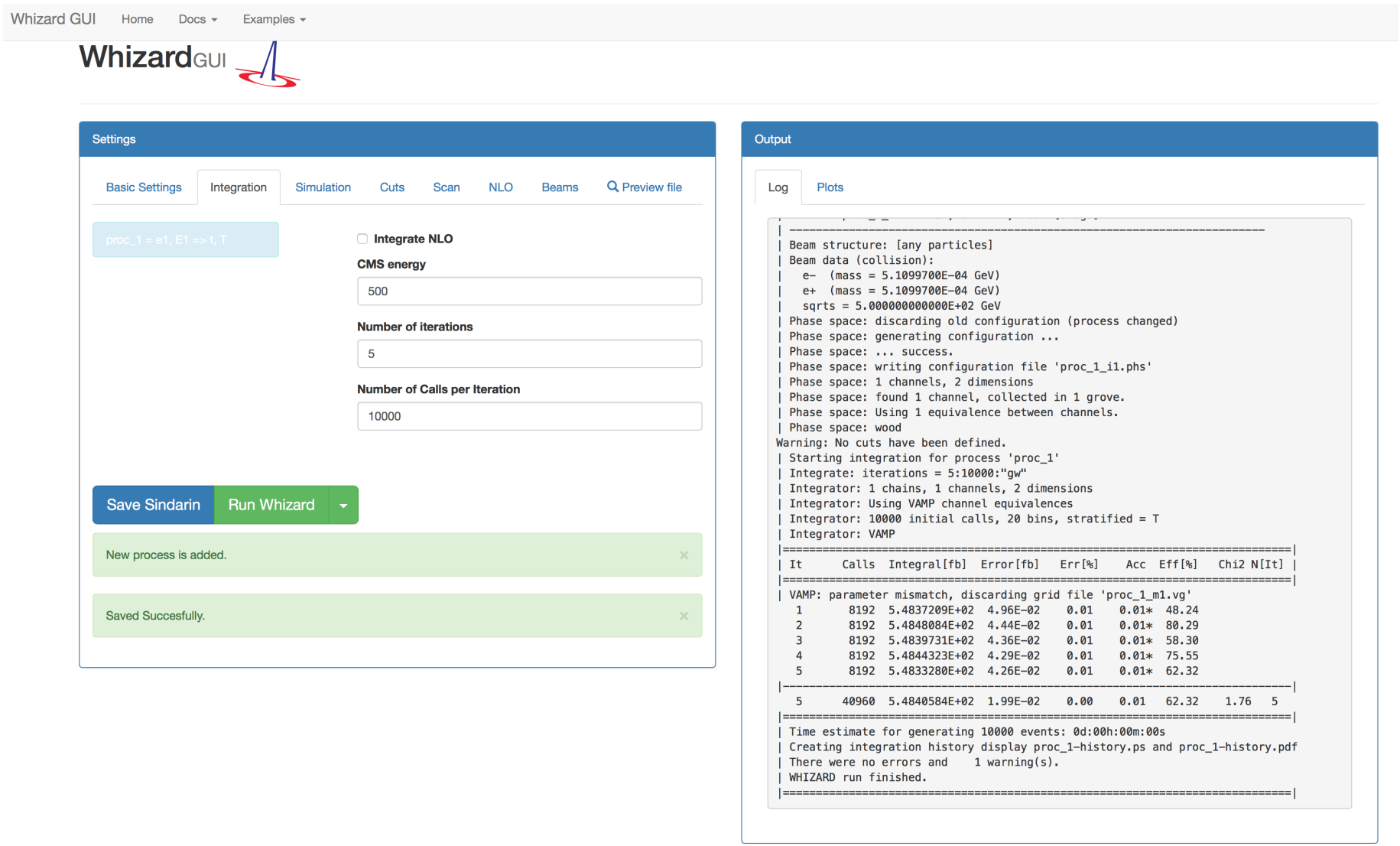
- unified syntax for cuts, observables, histograms, plots embedded in steering code
- Can work on own event samples or on event files
- Resulting plots use MetaPost $\Rightarrow$ integrates seamlessly with LaTeX
- `FastJet` for jet algorithms in cuts/analysis

Also in Sindarin:

- Logicals, strings, integers, reals, complex
- If-then-else branching
- Parameter scans/loops (also nested)

. . . more to come.

WHIZARD GUI


Whizard GUI
Home
Docs
Examples
WhizardGUI
Settings
Basic Settings
Integration
Simulation
Cuts
Scan
NLO
Beams
Preview file
proc_1 = e1, E1 => t, T
Integrate NLO
CMS energy
500
Number of iterations
5
Number of Calls per Iteration
10000
Save Sindarin
Run Whizard
New process is added.
Saved Succesfully.
Output
Log
Plots
| Beam structure: [any particles]
| Beam data (collision):
| e- (mass = 5.1099700E-04 GeV)
| e+ (mass = 5.1099700E-04 GeV)
| sqrts = 5.000000000000E+02 GeV
| Phase space: discarding old configuration (process changed)
| Phase space: generating configuration ...
| Phase space: ... success.
| Phase space: writing configuration file 'proc_1_i1.phs'
| Phase space: 1 channels, 2 dimensions
| Phase space: found 1 channel, collected in 1 grove.
| Phase space: Using 1 equivalence between channels.
| Phase space: wood
Warning: No cuts have been defined.
| Starting integration for process 'proc_1'
| Integrate: iterations = 5:10000:"gw"
| Integrator: 1 chains, 1 channels, 2 dimensions
| Integrator: Using VAMP channel equivalences
| Integrator: 10000 initial calls, 20 bins, stratified = T
| Integrator: VAMP
| It Calls Integral[fb] Error[fb] Err[%] Acc Eff[%] Chi2 N[It]
| VAMP: parameter mismatch, discarding grid file 'proc_1_m1.vg'
1 8192 5.4837209E+02 4.96E-02 0.01 0.01* 48.24
2 8192 5.4848084E+02 4.44E-02 0.01 0.01* 80.29
3 8192 5.4839731E+02 4.36E-02 0.01 0.01* 58.30
4 8192 5.4844323E+02 4.29E-02 0.01 0.01* 75.55
5 8192 5.4833280E+02 4.26E-02 0.01 0.01* 62.32
5 40960 5.4840584E+02 1.99E-02 0.00 0.01 62.32 1.76 5
| Time estimate for generating 10000 events: 0d:00h:00m:00s
| Creating integration history display proc_1-history.ps and proc_1-history.pdf
| There were no errors and 1 warning(s).
| WHIZARD run finished.

Summary: Recent Developments in WHIZARD

- Further refinements for ILC/CLIC beam spectra
- UFO support
- Models: EFT, unitary extrapolation, resonances, $t\bar{t}$ threshold
- Parallel evaluation
- GUI
- NLO, threshold resummation and matching: $\Rightarrow$ Jürgen's talk

whizard.hepforge.org