SHERPA-2.2.5: overview, developments and usage

Marek Schönherr

Universität Zürich

CERN 21 Jun 2018

Contents

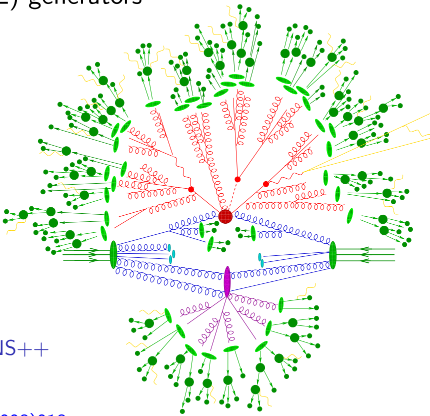
Overview

1. Overview
2. Recent results
3. Usage
4. Running SHERPA
5. Conclusions

The SHERPA event generator framework

JHEP02(2009)007

- Two multi-purpose Matrix Element (ME) generators
  AMEGIC++ JHEP02(2002)044, EPJC53(2008)501
  COMIX JHEP12(2008)039, PRL109(2012)042001
- Two Parton Shower (PS) generators
  CSSHOWER JHEP03(2008)038
  DIRE EPJC75(2015)461
- A multiple interaction simulation
  à la PYTHIA AMISIC++ hep-ph/0601012
- A cluster fragmentation module
  AHADIC++ EPJC36(2004)381
- A hadron and τ decay package HADRONS++
- A higher order QED generator using
  YFS-resummation PHOTONS++ JHEP12(2008)018



Sherpa's traditional strength is the perturbative part of the event

LO, NLO, NNLO, LOPS, NLOPS, NNLOPS, MEPS, MENLOPS, MEPS@NLO

Acronyms and nomenclature

Fixed order calculations

- matrix elements only, implies fixed multiplicities
- no parton shower, no non-perturbative physics, no particle level

⇒ LO, NLO, NNLO

Parton shower matched calculations

- combination of fixed order calculation and parton shower for one multiplicity
- particle level predictions, no multijet observables

⇒ LOPS, NLOPS, NNLOPS

Multijet merged calculations

- combination of parton shower matched calculations for increasing final state multiplicities (mostly jets)
- particle level predictions, multijet observables

⇒ MEPS(@LO), MEPS@NLO (special case MENLOPS)

SHERPA-2.2.5

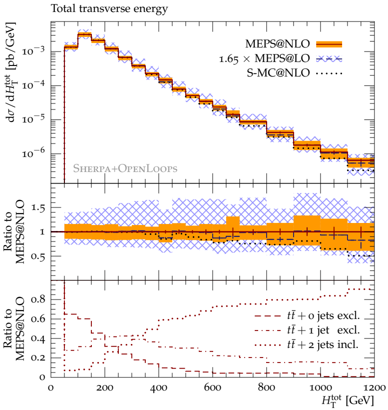
- SHERPA-2.2.5 released Apr '18
- contains bugfixes for all known bugs of SHERPA-2.2.4
- `UFO` support for BSM physics
- two parton showers: CSSHOWER (default), DIRE
- on-the-fly scale and PDF variations for ME part in
  - LO, NLO
  - LOPS, NLOPS (S-MC@NLO)
  - MEPS, MENLOPS, MEPS@NLO

  $\rightarrow$ use named weights in HEPMC (av. since HEPMC-2.06)
- full scale & PDF variations including correlated with parton shower and for NNLO/NNLOPS in SHERPA-2.3.0
- allow to force HEPMC event record into pure tree structure, lost information available through disconnected vertices
- default PDF: `NNPDF30_nnlo_as_0118`

Recent results

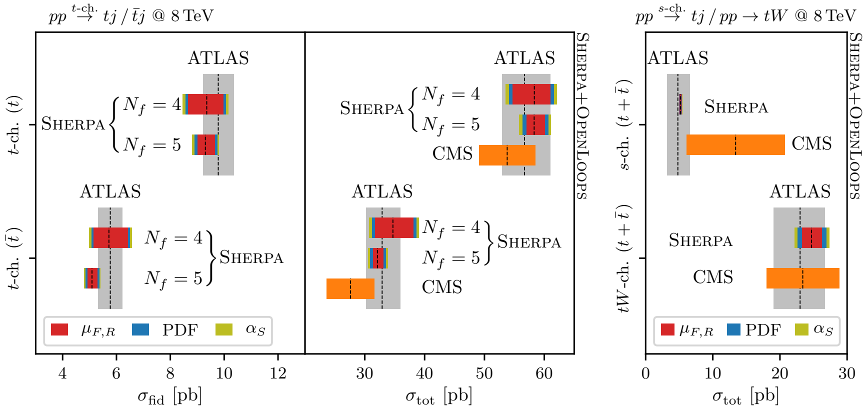
Top physics I

Höche, Krauss, Maierhöfer, Pozzorini, MS, Siegert arXiv:1401.7971



Top physics II

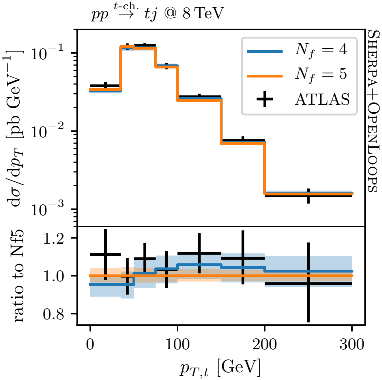
Bothmann, Krauss, MS arXiv:1711.02568

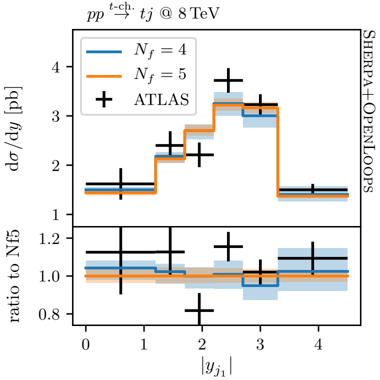


- good description of incl. and fiducial xsec with $N_f = 4$ and $N_f = 5$

Top physics II

Bothmann, Krauss, MS arXiv:1711.02568





- good description of incl. and fiducial xsec with $N_f = 4$ and $N_f = 5$

Higgs physics I

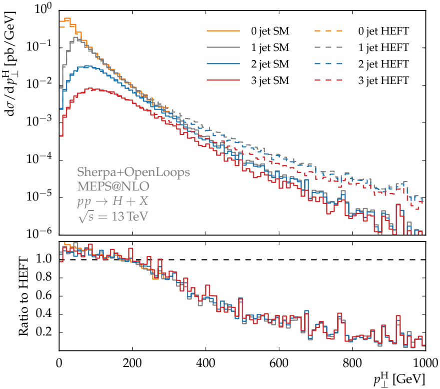
Buschmann,Goncalves,Kuttimalai,MS,Krauss,Plehn JHEP02(2015)038
Kuttimalai,Krauss,Maierhöfer,MS LH'15 arXiv:1605.04692
Kuttimalai,Krauss,Maierhöfer,MS for YR4

$pp \to H +$ jets production (ggF)

- correction factor/weight

$$r_t^{(n)} = \frac{|\mathcal{M}^{(n)}(m_t)|^2}{|\mathcal{M}^{(n)}(m_t \to \infty)|^2}$$

- loops from OPENLOOPS
- construct MEPS@NLO from reweighted S-MC@NLO
- factorised approach for unknown top mass dependence in V_n, otherwise exact NLO mass dependence



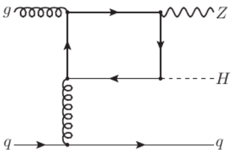
$$\mathrm{d}\sigma_n = \mathrm{d}\Phi_n\, r_t^{(n)} \left[\mathrm{B}_n + \mathrm{V}_n + \int \mathrm{d}\Phi_1\, \mathrm{D}_n\right] \widetilde{\mathrm{PS}}_n + \mathrm{d}\Phi_{n+1} \left[r_t^{(n+1)}\mathrm{R}_n - r_t^{(n)}\mathrm{D}_n\right]$$

Higgs physics II

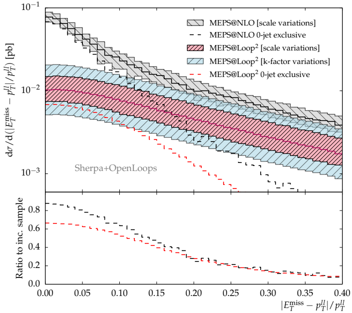
Goncalves,Krauss,Kuttimalai,Maierhöfer PRD92(2015)7,073006

$pp \to Z\,H$ + jets production

- MEPS@NLO for $q\bar{q}$
  MEPS@LOOP^2 for gg
- care for $qg \to ZHq$:



  $\to$ part of NLO ZHj
  $\to$ in loop-induced as gauge inv. subset of NNLO ZHj
- loops from OPENLOOPS



$pp \to Z[\to \ell\ell\,]H[\to \text{inv}]$ + jets

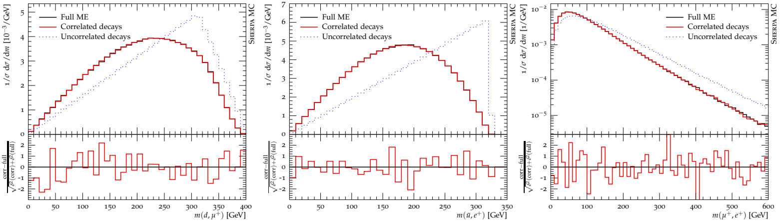
BSM physics

Höche,Kuttimalai,Schumann,Siegert EPJC75(2015)3,135

- full support for UFO model Degrande et.al. CPC183(2012)1201
- Lorentz structures automatically built,
  colour structures mapped on SM/MSSM-like
- automatic identification of all 1→2 and 1→3 decay channels of every unstable particle in the model
  → calculation of all decay widths (LO)
- per default all decay channel used
  → inclusive production
  → mechanism to select individual channels,
  cross section optionally adjusted accordingly
- spin-correlated decay chains of arbitrary length
  using spin density matrices Richardson JHEP11(2001)029,
  Knowles CPC58(1990)271

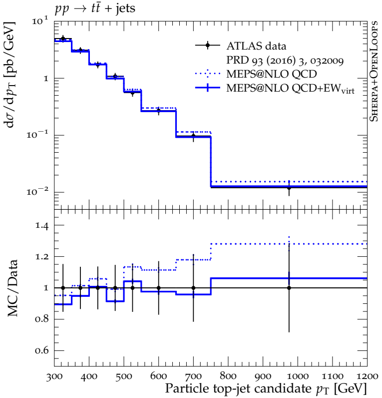
BSM physics

Höche,Kuttimalai,Schumann,Siegert EPJC75(2015)3,135



- simple three-step example:
  $pp \to \tilde{u}[\to d\chi_1^+[\to \chi_1^0 W^+[\to \mu^+\nu_\mu]]]\ \tilde{u}^*[\to \bar{u}\chi_2^0[\to e^+\tilde{e}^-[\to e^-\chi_1^0]]]$
- use truncated showers for QCD radiation off intermediate particles
- QED correction for each decay in YFS soft-photon resummation

Electroweak corrections



Gütschow, Lindert, MS arXiv:1803.00950

- $t\bar{t} + 0, 1j$@NLO QCD+EW$_\mathrm{virt}$ $+ 2, 3, 4j$@LO
- include approx. virtual corrections
- additional LO multiplicities inherit electroweak corrections through MENLOPS differential K-factor

  Höche, Krauss, MS, Siegert arXiv:1009.1127
- improved description of data

Usage

Usage

SHERPA is steered through one run card. The run card is divided into sections to facilitate readability.

`(processes)`	definition of all process to be calculated multiple processes possible
`(selectors)`	phase space cuts on matrix elements apply to core process in multijet merging
`(run)`	all other switches general steering, matrix element generators, parton showers, beam setup, multiple interactions, hadronisation, hadron decays, QED corrections, analysis, etc.

(Almost) all relevant switches to steer SHERPA are documented in the Manual, including details on meaning and defaults.

– online: `http://sherpa.hepforge.org/doc/SHERPA-MC-2.2.5.html`
– shipped: `<prefix>/share/doc/SHERPA-MC/Sherpa.html`

General structure

Switches: `KEY VALUE`

- standard input structure, some switches read in array of values
- = iterpreted as space

Tags: `TAG:=STRING`

- replaces all occurences of `TAG` in run card with `STRING`

Term: computable string, e.g. `sqr(QCUT/E_CMS)`

- some switches take terms as input, ie. parameters are parsed through SHERPA's `Algebra_Interpreter`

End-of-line: `;`
Comments: `#, %`
Command line arguments: `KEY=VALUE`, `TAG:=STRING`

- needs to be one string, added to `(run)` section, takes precedence over file

Special syntax for `(processes)` and `(selector)` sections.

General settings

```
(run){
  % general setting
  EVENTS 1M; ERROR 0.99;
```

number of events & integration error target

```
  % scales, tags for scale variations
  FSF:=1.; RSF:=1.; QSF:=1.;
  SCALES METS{FSF*MU_F2}{RSF*MU_R2}{QSF*MU_Q2};

  % tags for process setup
  NJET:=4; LJET:=2,3,4; QCUT:=20.;

  % me generator settings
  ME_SIGNAL_GENERATOR Comix Amegic LOOPGEN0 LOOPGEN1;
  EVENT_GENERATION_MODE PartiallyUnweighted;
  LOOPGEN0:=Internal;
  LOOPGEN1:=BlackHat;

  % collider setup
  BEAM_1 2212; BEAM_ENERGY_1 6500.;
  BEAM_2 2212; BEAM_ENERGY_2 6500.;
}(run)
```

General settings

```
(run){
  % general setting
  EVENTS 1M; ERROR 0.99;

  % scales, tags for scale variations
  FSF:=1.; RSF:=1.; QSF:=1.;
  SCALES METS{FSF*MU_F2}{RSF*MU_R2}{QSF*MU_Q2};

  % tags for process setup
  NJET:=4; LJET:=2,3,4; QCUT:=20.;

  % me generator settings
  ME_SIGNAL_GENERATOR Comix Amegic LOOPGEN0 LOOPGEN1;
  EVENT_GENERATION_MODE PartiallyUnweighted;
  LOOPGEN0:=Internal;
  LOOPGEN1:=BlackHat;

  % collider setup
  BEAM_1 2212; BEAM_ENERGY_1 6500.;
  BEAM_2 2212; BEAM_ENERGY_2 6500.;
}(run)
```

number of events & integration error target

tags for scale prefactors
scale definition $\{\mu_F\}\{\mu_R\}\{\mu_Q\}$
$\rightarrow$ use default METS (CKKW)

General settings

```
(run){
  % general setting
  EVENTS 1M; ERROR 0.99;
```
number of events & integration error target

```
  % scales, tags for scale variations
  FSF:=1.; RSF:=1.; QSF:=1.;
  SCALES METS{FSF*MU_F2}{RSF*MU_R2}{QSF*MU_Q2};
```
tags for scale prefactors

scale definition $\{\mu_F\}\{\mu_R\}\{\mu_Q\}$

$\rightarrow$ use default METS (CKKW)

```
  % tags for process setup
  NJET:=4; LJET:=2,3,4; QCUT:=20.;
```
multijet merging setup

max. jet multi, NLO multiplicities, Q_{cut}

```
  % me generator settings
  ME_SIGNAL_GENERATOR Comix Amegic LOOPGEN0 LOOPGEN1;
  EVENT_GENERATION_MODE PartiallyUnweighted;
  LOOPGEN0:=Internal;
  LOOPGEN1:=BlackHat;

  % collider setup
  BEAM_1 2212; BEAM_ENERGY_1 6500.;
  BEAM_2 2212; BEAM_ENERGY_2 6500.;
}(run)
```

General settings

```
(run){
  % general setting
  EVENTS 1M; ERROR 0.99;
```

number of events & integration error target

```
  % scales, tags for scale variations
  FSF:=1.; RSF:=1.; QSF:=1.;
  SCALES METS{FSF*MU_F2}{RSF*MU_R2}{QSF*MU_Q2};
```

tags for scale prefactors
scale definition $\{\mu_F\}\{\mu_R\}\{\mu_Q\}$
$\rightarrow$ use default METS (CKKW)

```
  % tags for process setup
  NJET:=4; LJET:=2,3,4; QCUT:=20.;
```

multijet merging setup
max. jet multi, NLO multiplicities, Q_{cut}

```
  % me generator settings
  ME_SIGNAL_GENERATOR Comix Amegic LOOPGEN0 LOOPGEN1;
  EVENT_GENERATION_MODE PartiallyUnweighted;
  LOOPGEN0:=Internal;
  LOOPGEN1:=BlackHat;
```

matrix element generators
& event generation mode

```
  % collider setup
  BEAM_1 2212; BEAM_ENERGY_1 6500.;
  BEAM_2 2212; BEAM_ENERGY_2 6500.;
}(run)
```

General settings

```
(run){
  % general setting
  EVENTS 1M; ERROR 0.99;
```

number of events & integration error target

```
  % scales, tags for scale variations
  FSF:=1.; RSF:=1.; QSF:=1.;
  SCALES METS{FSF*MU_F2}{RSF*MU_R2}{QSF*MU_Q2};
```

tags for scale prefactors
scale definition $\{\mu_F\}\{\mu_R\}\{\mu_Q\}$
$\rightarrow$ use default METS (CKKW)

```
  % tags for process setup
  NJET:=4; LJET:=2,3,4; QCUT:=20.;
```

multijet merging setup
max. jet multi, NLO multiplicities, Q_{cut}

```
  % me generator settings
  ME_SIGNAL_GENERATOR Comix Amegic LOOPGEN0 LOOPGEN1;
  EVENT_GENERATION_MODE PartiallyUnweighted;
  LOOPGEN0:=Internal;
  LOOPGEN1:=BlackHat;
```

matrix element generators
& event generation mode

```
  % collider setup
  BEAM_1 2212; BEAM_ENERGY_1 6500.;
  BEAM_2 2212; BEAM_ENERGY_2 6500.;
}(run)
```

collider setup

Particle containers

SHERPA introduces particle containers to facilite easy process declaration and minimise the risk of accidentally omitting partonic channels in multi-particle calculations. Containers help speed up calculations by helping SHERPA to map processes and calculate them simultaneously. Thus, containers must only contain particles of identical masses and thus identical phase space.

There are a few pre-defined containers:

90	leptons (massless ℓ)
91	neutrinos
92	fermions (massless ℓ, q and $\bar{q}$, neutrinos)
93	jets (g, massless q and $\bar{q}$)
94	quarks (massless q and $\bar{q}$)

User defined containers:

```
PARTICLE_CONTAINER <id> <name> <list of particles>
```

Process setup

```
(processes){
  Process 93 93 -> 13 -13 93{NJET};
  Order (*,2);
  NLO_QCD_Mode MC@NLO {LJET};
  ME_Generator Amegic {LJET};
  RS_ME_Generator Comix {LJET};
  Loop_Generator LOOPGEN0 {2};
  Loop_Generator LOOPGEN1 {3,4};
  End process;
}(processes)
```

$pp \to \mu^- \mu^+ + n\,\text{jets}$

Process setup

```
(processes){
  Process 93 93 -> 13 -13 93{NJET};
  Order (*,2);
  NLO_QCD_Mode MC@NLO {LJET};
  ME_Generator Amegic {LJET};
  RS_ME_Generator Comix {LJET};
  Loop_Generator LOOPGEN0 {2};
  Loop_Generator LOOPGEN1 {3,4};
  End process;
}(processes)
```

$pp \to \mu^-\mu^+ + n\,\text{jets}$

$\mathcal{O}(\alpha_s^n\alpha)$ $n = 0, 1, 2, ..$

Process setup

```
(processes){
  Process 93 93 -> 13 -13 93{NJET};
  Order (*,2);
  NLO_QCD_Mode MC@NLO {LJET};
  ME_Generator Amegic {LJET};
  RS_ME_Generator Comix {LJET};
  Loop_Generator LOOPGEN0 {2};
  Loop_Generator LOOPGEN1 {3,4};
  End process;
}(processes)
```

$pp \to \mu^-\mu^+ + n$ jets
$\mathcal{O}(\alpha_s^n \alpha)$ $n = 0, 1, 2, ..$
final state multis to be computed at NLO QCD

Process setup

```
(processes){
  Process 93 93 -> 13 -13 93{NJET};
  Order (*,2);
  NLO_QCD_Mode MC@NLO {LJET};
  ME_Generator Amegic {LJET};
  RS_ME_Generator Comix {LJET};
  Loop_Generator LOOPGEN0 {2};
  Loop_Generator LOOPGEN1 {3,4};
  End process;
}(processes)
```

$pp \to \mu^-\mu^+ + n$ jets
$\mathcal{O}(\alpha_s^n\alpha)$ $n = 0, 1, 2, ..$
final state multis to be computed at NLO QCD

which loop generator to use for which process

Phase space cuts

Many processes need phase space cuts in order to be well defined. They are set up in the `(selector)` section. Note: In multijet merged samples, the additional cuts on higher multiplicities are automatically applied after specifying the merging scale.

One particle selectors:
`<sel> <pdgid> <min> <max>`
e.g. `PT`, `Rapidity`, `PseudoRapidity`, etc.

Two particle selectors:
`<sel> <pdgid1> <pdgid2> <min> <max>`
e.g. `PT2`, `DeltaR`, `Mass`, etc.

Various jet algorithms. Interface to Fastjet available, only QCD partons fed into jet algorithms.

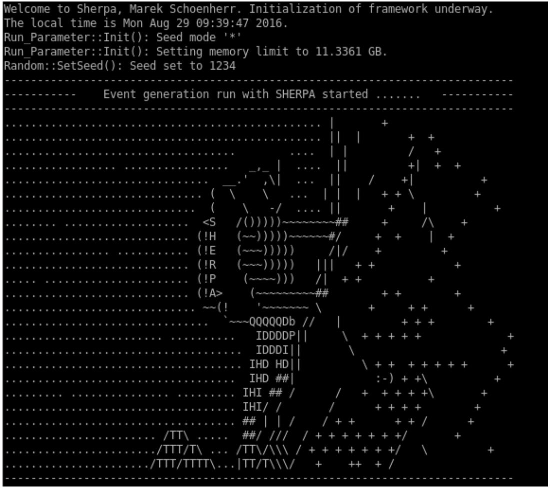
Process setup

```
(selector){
  Mass 13 -13 66 E_CMS
}(selector)
```

$66\,\text{GeV} < m_{\mu\mu} < E_{\text{CMS}}$

Running SHERPA

Running SHERPA



SHERPA generally runs in three stages:

1) generate process libraries
2) integrate partonic cross sections
3) generate events

Steps 1) and 2) only have to be performed once.

Start up and logo

Running SHERPA

```
SVN branch branches/rel-2-2-1, revision 28665.

Beam_Spectra_Handler :
   type = Monochromatic*Monochromatic
   for    P+  ((6500,0,0,6500))
   and    P+  ((6500,0,0,-6500))
PDF set 'NNPDF30NNLO' loaded for beam 1 (P+).
PDF set 'NNPDF30NNLO' loaded for beam 2 (P+).
Initialized the ISR: (SF)*(SF)
One_Running_AlphaS::One_Running_AlphaS() {
  Setting \alpha_s according to PDF
  perturbative order 2
  \alpha_s(M_Z) = 0.118
}
One_Running_AlphaS::One_Running_AlphaS() {
  Setting \alpha_s according to PDF
  perturbative order 2
  \alpha_s(M_Z) = 0.118
}
```

Version information
Beams & PDFs
α_s info

Running SHERPA

List of Particle Data

IDName	kfc	MASS[<kfc>]	WIDTH[<kfc>]	STABLE[<kfc>]	MASSIVE[<kfc>]	ACTIVE[<kfc>]	YUKAWA[<kfc>]
d	1	0.01	0	1	0	1	0
u	2	0.005	0	1	0	1	0
s	3	0.2	0	1	0	1	0
c	4	1.42	0	1	0	1	0
b	5	4.8	0	1	0	1	0
t	6	173.21	2	0	1	1	173.21
e-	11	0.000511	0	1	0	1	0
ve	12	0	0	1	0	1	0
mu-	13	0.105	0	1	0	1	0
vmu	14	0	0	1	0	1	0
tau-	15	1.777	2.26735e-12	0	0	1	0
vtau	16	0	0	1	0	1	0
G	21	0	0	1	0	1	0
P	22	0	0	1	0	1	0
Z	23	91.1876	2.4952	0	1	1	91.1876
W+	24	80.385	2.085	0	1	1	80.385
h0	25	125	0	0	1	1	125

List of Particle Containers

IDName	kfc	Constituents
l	90	{e-,e+,mu-,mu+,tau-,tau+}
v	91	{ve,veb,vmu,vmub,vtau,vtaub}
j	93	{d,db,u,ub,s,sb,c,cb,b,bb,G}
Q	94	{d,db,u,ub,s,sb,c,cb,b,bb}
r	99	{d,db,u,ub,s,sb,c,cb,b,bb,G}

Particles of the model and their properties containers

Running SHERPA

```
HEFT::InitEFTVertices() {
  ggh coupling is (5.15039e-05,7.95745e-07) [ \alpha_s = 0.118 ]
  ggh coupling is (-2.65449e-05,-4.10124e-07) [ 1/\alpha = 128.802 ]
}
Hadron_Init::Init(): Initializing kf table for hadrons.
Initialized the Fragmentation_Handler.
Initialized the Soft_Collision_Handler.
CS_Shower::CS_Shower(): Set respect Q2 mode 0
CS_Shower::CS_Shower(): Set color setter mode 0
CS_Shower::CS_Shower(): Set respect Q2 mode 0
CS_Shower::CS_Shower(): Set color setter mode 0
Initialized the Shower_Handler.
+-----------------------------------------+
|   X   X   X XXXX  XXX  XXX  XXX         |
|  X X  XX XX X    X      X  X     X   X  |
| X   X X X X XXX  X XXX  X  X    XXX XXX |
| XXXXX X   X X    X   X  X  X     X   X  |
| X   X X   X XXXX  XXX  XXX  XXX         |
+-----------------------------------------+
| please cite: JHEP 0202:044,2002         |
+-----------------------------------------+
ME_Generator_Base::SetPSMasses(): Massive PS flavours for Comix: (c,cb,b,bb,e-,e+,mu-,mu+,tau-,tau+)
+----------------------------------+
|                                  |
|      CCC  OOO  M   M I X   X     |
|     C    O   O MM MM I  X X      |
|     C    O   O M M M I   X       |
|     C    O   O M   M I  X X      |
|      CCC  OOO  M   M I X   X     |
|                                  |
+==================================+
|  Color dressed  Matrix Elements  |
|     http://comix.freacafe.de     |
|   please cite  JHEP12(2008)039   |
+----------------------------------+
ME_Generator_Base::SetPSMasses(): Massive PS flavours for Amegic: (c,cb,b,bb,e-,e+,mu-,mu+,tau-,tau+)
Amegic::Initialize(): Set gauge 1.
ME_Generator_Base::SetPSMasses(): Massive PS flavours for Internal: (c,cb,b,bb,e-,e+,mu-,mu+,tau-,tau+)
ME_Generator_Base::SetPSMasses(): Massive PS flavours for Internal: (c,cb,b,bb,e-,e+,mu-,mu+,tau-,tau+)
Matrix_Element_Handler::BuildProcesses(): Looking for processes ME_Generator_Base::SetPSMasses(): Massive
,cb,b,bb,e-,e+,mu-,mu+,tau-,tau+)
. done ( 30 MB, 0s / 0s ).
Matrix_Element_Handler::InitializeProcesses(): Performing tests . done ( 30 MB, 0s / 0s ).
Initialized the Matrix_Element_Handler for the hard processes.
Initialized the Beam_Remnant_Handler.
```

further model info

shower initialisation

matrix element generator initialisation

infos on which particles massive in parton shower, but not in matrix element

Running SHERPA

```
Library_Loader::LoadLibrary(): Failed to load library 'libProc_P2_2.so'.
Library_Loader::LoadLibrary(): Failed to load library 'libProc_P2_2.so'.
Library_Loader::LoadLibrary(): Failed to load library 'libProc_P2_2.so'.
Library_Loader::LoadLibrary(): Failed to load library 'libProc_P2_2.so'.
Library_Loader::LoadLibrary(): Failed to load library 'libProc_P2_2.so'.
........... done ( 37 MB, 1s / 0s ).
Sherpa: Matrix_Element_Handler::InitializeProcesses throws normal exit:
   Source code created. Run './makelibs' to compile.
-------------------------------------------------------------------------
Please cite the publications listed in 'Sherpa_References.tex'.
  Extract the bibtex list by running 'get_bibtex Sherpa_References.tex'
  or email the file to 'slaclib2@slac.stanford.edu', subject 'generate'.
-------------------------------------------------------------------------
```

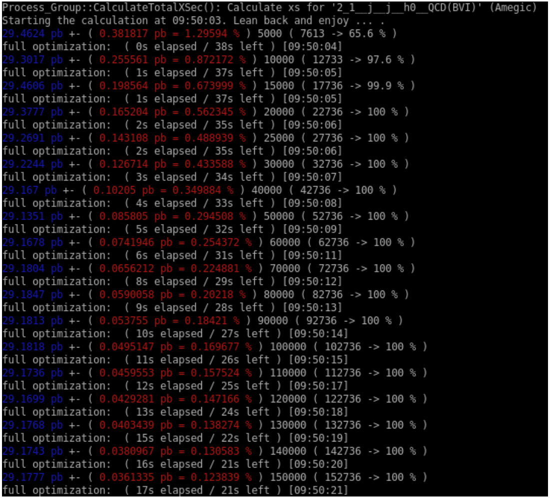
Step 1) finishes with a normal exit, prompting to compile the libraries.

Running SHERPA

```
Hadron_Decay_Map::Read:   Initializing HadronDecays.dat. This may take some time.
Initialized the Hadron_Decay_Handler, Decay model = Hadrons
ME_Generator_Base::SetPSMasses(): Massive PS flavours for Amisic: (c,cb,b,bb,e-,e+,mu-,mu+,tau-,tau
Simple_Chain::InitializeProcessList(): Init processes ..........................................
......................................................................................... do
Grid_Creator::ReadInGrid(): Reading grid .......................................................
........................................................................... done.
Simple_Chain::CalculateTotal(): Result is {
   \sigma_{hard} = 39.5319 mb
   at PT_{min} = 4.68991 GeV
}
Profile_Function_Base::CalculateOMean(2.32106): Results are {
   k           = 7.14376
   <\tilde{O}> = 0.324907
}
Initialized the Multiple_Interactions_Handler (MI_Handler).
Initialized the Soft_Photon_Handler.
Variations::InitialiseParametersVector(0 variations){
  Named variations:
}
```

Step 2) will commence to load the libraries, then initialise the multiple interactions

Running SHERPA



```
Process_Group::CalculateTotalXSec(): Calculate xs for '2_1__j__j__h0__QCD(BVI)' (Amegic)
Starting the calculation at 09:50:03. Lean back and enjoy ... .
29.4624 pb +- ( 0.381817 pb = 1.29594 % ) 5000 ( 7613 -> 65.6 % )
full optimization:  ( 0s elapsed / 38s left ) [09:50:04]
29.3017 pb +- ( 0.255561 pb = 0.872172 % ) 10000 ( 12733 -> 97.6 % )
full optimization:  ( 1s elapsed / 37s left ) [09:50:05]
29.4606 pb +- ( 0.198564 pb = 0.673999 % ) 15000 ( 17736 -> 99.9 % )
full optimization:  ( 1s elapsed / 37s left ) [09:50:05]
29.3777 pb +- ( 0.165204 pb = 0.562345 % ) 20000 ( 22736 -> 100 % )
full optimization:  ( 2s elapsed / 35s left ) [09:50:06]
29.2691 pb +- ( 0.143108 pb = 0.488939 % ) 25000 ( 27736 -> 100 % )
full optimization:  ( 2s elapsed / 35s left ) [09:50:06]
29.2244 pb +- ( 0.126714 pb = 0.433588 % ) 30000 ( 32736 -> 100 % )
full optimization:  ( 3s elapsed / 34s left ) [09:50:07]
29.167 pb +- ( 0.10205 pb = 0.349884 % ) 40000 ( 42736 -> 100 % )
full optimization:  ( 4s elapsed / 33s left ) [09:50:08]
29.1351 pb +- ( 0.085805 pb = 0.294508 % ) 50000 ( 52736 -> 100 % )
full optimization:  ( 5s elapsed / 32s left ) [09:50:09]
29.1678 pb +- ( 0.0741946 pb = 0.254372 % ) 60000 ( 62736 -> 100 % )
full optimization:  ( 6s elapsed / 31s left ) [09:50:11]
29.1804 pb +- ( 0.0656212 pb = 0.224881 % ) 70000 ( 72736 -> 100 % )
full optimization:  ( 8s elapsed / 29s left ) [09:50:12]
29.1847 pb +- ( 0.0590058 pb = 0.20218 % ) 80000 ( 82736 -> 100 % )
full optimization:  ( 9s elapsed / 28s left ) [09:50:13]
29.1813 pb +- ( 0.053755 pb = 0.18421 % ) 90000 ( 92736 -> 100 % )
full optimization:  ( 10s elapsed / 27s left ) [09:50:14]
29.1818 pb +- ( 0.0495147 pb = 0.169677 % ) 100000 ( 102736 -> 100 % )
full optimization:  ( 11s elapsed / 26s left ) [09:50:15]
29.1736 pb +- ( 0.0459553 pb = 0.157524 % ) 110000 ( 112736 -> 100 % )
full optimization:  ( 12s elapsed / 25s left ) [09:50:17]
29.1699 pb +- ( 0.0429281 pb = 0.147166 % ) 120000 ( 122736 -> 100 % )
full optimization:  ( 13s elapsed / 24s left ) [09:50:18]
29.1768 pb +- ( 0.0403439 pb = 0.138274 % ) 130000 ( 132736 -> 100 % )
full optimization:  ( 15s elapsed / 22s left ) [09:50:19]
29.1743 pb +- ( 0.0380967 pb = 0.130583 % ) 140000 ( 142736 -> 100 % )
full optimization:  ( 16s elapsed / 21s left ) [09:50:20]
29.1777 pb +- ( 0.0361335 pb = 0.123839 % ) 150000 ( 152736 -> 100 % )
full optimization:  ( 17s elapsed / 21s left ) [09:50:21]
```

Integration of partonic matrix elements

Running SHERPA

```
29.1702 pb +- ( 0.0280843 pb = 0.0962774 % ) 210000 ( 212736 -> 100 % )
full optimization:  ( 24s elapsed / 14s left ) [09:50:29]
29.1659 pb +- ( 0.027157 pb = 0.0931122 % ) 220000 ( 222736 -> 100 % )
full optimization:  ( 25s elapsed / 13s left ) [09:50:30]
29.1706 pb +- ( 0.0262998 pb = 0.0901588 % ) 230000 ( 232736 -> 100 % )
integration time:  ( 26s elapsed / 12s left ) [09:50:31]
29.1718 pb +- ( 0.0247908 pb = 0.0849821 % ) 250000 ( 252736 -> 100 % )
integration time:  ( 29s elapsed / 9s left ) [09:50:34]
29.1774 pb +- ( 0.0235071 pb = 0.080566 % ) 270000 ( 272736 -> 100 % )
integration time:  ( 31s elapsed / 7s left ) [09:50:36]
29.1715 pb +- ( 0.0223807 pb = 0.0767214 % ) 290000 ( 292736 -> 100 % )
integration time:  ( 33s elapsed / 5s left ) [09:50:38]
29.1733 pb +- ( 0.0213806 pb = 0.0732881 % ) 310000 ( 312736 -> 100 % )
integration time:  ( 36s elapsed / 2s left ) [09:50:41]
2_1__j__j__h0__QCD(BVI) : 29.1733 pb +- ( 0.0213806 pb = 0.0732881 % )  exp. eff: 28.041 %
Process_Group::CalculateTotalXSec(): Calculate xs for '2_2__j__j__h0__j__QCD(RS)' (Comix)
Starting the calculation at 09:50:41. Lean back and enjoy ... .
1.47906 pb +- ( 0.0778611 pb = 5.26423 % ) 5000 ( 7642 -> 65.4 % )
full optimization:  ( 2s elapsed / 2m 59s left ) [09:50:44]
1.46325 pb +- ( 0.0542023 pb = 3.70424 % ) 10000 ( 12770 -> 97.5 % )
full optimization:  ( 5s elapsed / 2m 54s left ) [09:50:46]
1.49925 pb +- ( 0.0432498 pb = 2.88476 % ) 15000 ( 17775 -> 99.9 % )
full optimization:  ( 8s elapsed / 2m 50s left ) [09:50:49]
1.51216 pb +- ( 0.0363024 pb = 2.40069 % ) 20000 ( 22775 -> 100 % )
full optimization:  ( 11s elapsed / 2m 47s left ) [09:50:52]
1.5085 pb +- ( 0.0314168 pb = 2.08266 % ) 25000 ( 27775 -> 100 % )
full optimization:  ( 13s elapsed / 2m 45s left ) [09:50:55]
```

Integration of partonic matrix elements

Running SHERPA

Start of event generation

```
-----------------------------------------------------------
-- SHERPA generates events with the following structure --
-----------------------------------------------------------
Perturbative       : Signal_Processes
Perturbative       : Hard_Decays
Perturbative       : Jet_Evolution:CSS
Perturbative       : Lepton_FS_QED_Corrections:None
Perturbative       : Multiple_Interactions:Amisic
Perturbative       : Minimum_Bias:Off
Hadronization      : Beam_Remnants
Hadronization      : Hadronization:Ahadic
Hadronization      : Hadron_Decays
---------------------------------------------------------
█ Event 700 ( 9s elapsed / 2m 1s left ) -> ETA: Mon Aug 29 10:03
  XS = 30.5692 pb +- ( 0.940913 pb = 3.07 % )
```

Run

```
------------------------------------------------------
-- SHERPA generates events with the following structure --
------------------------------------------------------
Perturbative       : Signal_Processes
Perturbative       : Hard_Decays
Perturbative       : Jet_Evolution:CSS
Perturbative       : Lepton_FS_QED_Corrections:None
Perturbative       : Multiple_Interactions:Amisic
Perturbative       : Minimum_Bias:Off
Hadronization      : Beam_Remnants
Hadronization      : Hadronization:Ahadic
Hadronization      : Hadron_Decays
-----------------------------------------------------
Decay_Channel::GenerateKinematics(Omega(c) --> ss_1 s e+ ve ) warning:
  d\Gamma(x)=1.75909e-12 > max(d\Gamma)=1.081e-12
  Event 10000 ( 134 s total ) = 6.41807e+06 evts/day
In Event_Handler::Finish : Summarizing the run may take some time.
+-------------------------------------------------------+
|                                                       |
|  Total XS is 30.7222 pb +- ( 0.167737 pb = 0.54 % )   |
|                                                       |
+-------------------------------------------------------+
Return_Value::PrintStatistics(): Statistics {
  Generated events: 10000
  New events {
    From "Jet_Evolution:CSS": 3 (39105) -> 0 %
  }
  Retried events {
    From "Beam_Remnants": 5 (10006) -> 0 %
    From "Hadronization:Ahadic": 1 (14009) -> 0 %
    From "Jet_Evolution:CSS": 233 (39105) -> 0.5 %
  }
  Retried phases {
    From "Hadron_Decay_Handler::RejectExclusiveChannelsFromFragmentation": 1996 (0) -> 1996
  }
  Retried methods {
    From "Decay_Channel::GenerateKinematics": 7 (2418822) -> 0 %
  }
}
-----------------------------------------------------------------------
Please cite the publications listed in 'Sherpa_References.tex'.
  Extract the bibtex list by running 'get_bibtex Sherpa_References.tex'
  or email the file to 'slaclib2@slac.stanford.edu', subject 'generate'.
-----------------------------------------------------------------------
Time: 2m 18s on Mon Aug 29 10:03:05 2016
 (User: 2m 17s, System: 0s, Children User: 0s, Children System: 0s)
marek@marek-laptop-uzh:~/work/sherpa/rel-2-2-1/Examples/H_in_GluonFusion/LHC_HJets$
```

Finished !!
Display statistics

SHERPA-2.2.5

- a new parton shower DIRE
- vastly extended support for UFO BSM format
- multijet merging for loop induced processes further tested, use as:
  - MEPS@LOOP^2
  - reweight MEPS@NLO Higgs production in HEFT with top mass dependence (approximate in virtual corrections only)
- on-the-fly variations of μ_R, μ_F, α_s and PDF for
  - LO, NLO
  - LOPS, NLOPS (S-MC@NLO)
  - MEPS, MENLOPS, MEPS@NLO
- incorporation of approx. NLO EW corrs in existing NLO QCD MEPS@NLO
- default PDF: `NNPDF30_nnlo_as_0118` including tune of non-perturbative parameters
- **coming in SHERPA-2.3.0:** PS reweighting, full NLO EW

http://sherpa.hepforge.org

Thank you for your attention!

Backup

Reweighting

Parameters

parametric e.g. $\alpha_s(m_Z)$, m_t, PDF

perturbative e.g. NLO, NLL, leading-N_c $\to$ $\boldsymbol{\mu_R}$, $\boldsymbol{\mu_F}$

algorithmic e.g. evolution variable, recoil schemes, matching scheme

Explicit variations

- can be done for any scale or PDF dependence
- functional form can be changed
- separate run (independent calculation) for every variation

On-the-fly variations Bothmann,MS,Schumann arXiv:1606.08753

- can be done for μ_R, μ_F, α_s & PDF dependence of ME & PS
- store in HEPMC weight container using LH'13 naming convention

General settings

EVENTS	<n>	number of events
EVENT_OUTPUT	<format>[<file>]	event output format and file
ANALYSIS	<type>	pass events to analysis framework
ANALYSIS_OUTPUT	<name>	name for analysis output

Example:

```
(run){
  % general settings
  EVENTS 5M; ANALYSIS Rivet;
  ANALYSIS_OUTPUT ana_proc1;
  EVENT_OUTPUT HepMC_GenEvent[evt_proc1];

  ...
}(run)
```

Note: Analysis details defined in separate (analysis) section.
Note: Within ATLAS/CMS software no/limited need for these switches.

Matrix element generator settings

ME_SIGNAL_GENERATOR	<MEG1> <MEG2> ...	list of ME generators
INTEGRATOR	<name>	integrator type to use
ERROR	<accu>	target error for integration
EVENT_GENERATION_MODE	<mode>	weighted or (partially) unweighted events
RESULT_DIRECTORY	<name>	dir to store integration results

Example:

```
(run){
  ...

  % me generator settings
  ME_SIGNAL_GENERATOR Comix Amegic LOOPGEN;
  EVENT_GENERATION_MODE Weighted;
  RESULT_DIRECTORY res_proc1;
  LOOPGEN:=Internal; % BlackHat/OpenLoops

  ...
}(run)
```

Scale setting

SCALES	<scale-setter>	algorithm to set the scales
CORE_SCALE	<scale-setter>	algorithm to set the core scales in scale setters using a cluster history

SHERPA's scale setter take three parameters, μ_F, μ_R, μ_Q (in that order). If less are given, the missing default to μ_F. Computable terms can be given as input, modifying predefined tags if applicable.

Example:

```
(run){
  ...

  % scales, tags for scale variations
  FSF:=1.; RSF:=1.; QSF:=1.;
  SCALES STRICT_METS{FSF*MU_F2}{RSF*MU_R2}{QSF*MU_Q2};

  ...
}(run)
```

Parton shower generator settings

SHOWER_GENERATOR	<psgen>	parton shower generator
CSS_EVOLUTION_SCHEME	<scheme>	evolution variable scheme
CSS_KIN_SCHEME	<scheme>	kinematic recoil scheme
CSS_FS_PT2MIN	<val>	final state infrared cutoff
CSS_IS_PT2MIN	<val>	initial state infrared cutoff
CSS_FS_AS_FAC	<val>	final state α_s scale factor
CSS_IS_AS_FAC	<val>	initial state α_s scale factor
CSS_SCALE_SCHEME	<scheme>	α_s scale scheme in $g \to q\bar{q}$

Example:

```
(run){
  ...

  % parton shower settings
  SHOWER_GENERATOR CSS;
  CSS_EVOLUTION_SCHEME 1;
  CSS_SCALE_SCHEME 1;

  ...
}(run)
```

Beams

BEAM_1	<pdgid>	PDG ID of incoming beam 1
BEAM_2	<pdgid>	PDG ID of incoming beam 2
BEAM_ENERGY_1	<val>	energy of incoming beam 1
BEAM_ENERGY_2	<val>	energy of incoming beam 2

Example:

```
(run){
  ...

  % beam settings
  BEAM_1 2212; BEAM_ENERGY_1 6500.;
  BEAM_2 2212; BEAM_ENERGY_2 6500.;

  ...
}(run)
```

PDFs

PDF_LIBRARY	<lib1> <lib2>	List of libraries to load PDFs from
PDF_SET	<set>	PDF set
PDF_SET_VERSION	<member>	PDF set member
PDF_SET_1/2	<set>	PDF set for beam 1/2
PDF_SET_MPI	<set>	PDF set for multiple interactions

Example:

```
(run){
  ...

  % PDF settings
  PDF_LIBRARY LHAPDFSherpa;
  PDF_SET NNPDF30_nnlo_as_0118;

  ...
}(run)
```

Model settings

MODEL	<model>	model to be used in the calculation
MASS[<id>]	<val>	set mass of particle with PDG ID <id>
WIDTH[<id>]	<val>	set width of particle with PDG ID <id>
MASSIVE[<id>]	<val>	set whether particle with PDG ID <id> is considered massive in the matrix elements
STABLE[<id>]	<val>	set whether particle with PDG ID <id> is considered stable

Example:

```
(run){
  ...

  % model parameters
  MODEL HEFT;
  MASS[6]  173.2; WIDTH[6]  1.5;
  MASS[25] 125.0; WIDTH[25] 0.;

  ...
}(run)
```

Model settings

EW_SCHEME	<scheme>	EW scheme, determines set of input parameters
WIDTH_SCHEME	<val>	how to treat particle widths
YUKAWA_MASSES	<val>	how to relate particle masses and Yukawa couplings
1/ALPHAQED(0)	<val>	value of α in Thomson limit
ALPHAQED_DEFAULT_SCALE	<scale>	scale at which to evaluate α

Example:

```
(run){
  ...

  % model parameters
  EW_SCHEME 3;
  WIDTH_SCHEME CMS;
  YUKAWA_MASSES Running;

  ...
}(run)
```

Hard decays

HARD_DECAYS	<on>	switch perturbative decays on/off
HDH_STATUS[<decay>]	<mode>	selectively disable/force individual decays
HDH_WIDTH[<decay>]	<val>	override computed width (LO) for decays
HDH_BR_WEIGHTS	<on>	apply $\Gamma(\text{active})/\Gamma_{\text{tot}}$ weight
HARD_SPIN_CORRELATIONS	<on>	switch on/off spinn correlations

Example:

```
(run){
  ...

  % settings for hard decays
  HARD_DECAYS On;
  HDH_STATUS[25,22,22] 2;
  HDH_BR_WEIGHTS 0;

  ...
}(run)
```

Non-perturbative physics

MI_HANDLER	<gen>	specify multiple interaction generator
FRAGMENTATION	<gen>	specify fragmentation generator
DECAYMODEL	<gen>	hadron decay generator
DECAYPATH	<path>	path where to find hadron decay database
SOFT_SPIN_CORRELATIONS	<on>	switch on/off spin correlations
MAX_PROPER_LIFETIME	<mm>	maximum proper lifetime in mm for particles to be considered unstable

Example:

```
(run){
  ...

  % non-perturbative settings
  FRAGMENTATION Lund; DECAYMODEL Lund;
  PARJ(21) 0.432; PARJ(41) 1.05; PARJ(42) 1.0;
  PARJ(47) 0.65; MSTJ(11) 5;

  ...
}(run)
```

QED corrections

`ME_QED`	<on>	switch QED corrections to hard ME on/off
`ME_QED_CLUSTERING`	<on>	swtich clustering before QED corrs. to ME to preserve resonances on/off
`ME_QED_CLUSTERING_THRESHOLD`	<val>	set resonance identification threshold in units of resonance width
`YFS_MODE`	<mode>	operation mode of YFS correction
`YFS_IR_CUTOFF`	<val>	infrared cutoff in photon emission

Example:

```
(run){
  ...

  % QED correction settings
  ME_QED On;
  ME_QED_CLUSTERING_THRESHOLD 3.;

  ...
}(run)
```

Matrix elements and process declaration

The processes to be calculated are set up in the (processes) section. It uses a more flexible syntax, tags are also replaced here.

Process	<proc>	partonic process to calculate
Order	(<qcd>,<qed>[,<bsm>])	perturbative orders of process
CKKW	<val>	merging cut
NLO_QCD_Mode	<mode>	fixed order or matched
ME_Generator	<gen>	generator for tree MEs
Loop_Generator	<gen>	generator for loop corrections
RS_ME_Generator	<gen>	generator for real subtracted MEs
Enhance_Factor	<fac>	enhancement factor
Enhance_Observable	<obs>	enhancement observable definition
End process		end of process declations

Multiple processes can be calculated simultaneously.

Matrix elements and process declaration

Example:

```
(processes){
  Process 93 93 -> 25 93{2};
  Order (*,0,1); CKKW sqr(20./E_CMS);
  NLO_QCD_Mode MC@NLO {1,2};
  Loop_Generator Internal;
  Enhance_Factor 0.1 {1};
  Enhance_Factor 10. {3};
  End process;
}(processes);
```

Bracket notation:

particles — $(n+1)$ processes with the multiplicity of the given flavour or container increased up to the given number $(0,1,\ldots,n)$

else — corresponding setting is applied to process with given total final state multiplicity

Phase space cuts

Example:

```
(processes){
  Process 93 93 -> 90 90 93 93;
  Order (2,2);
  End process;
}(processes);

(selector){
  FastjetFinder antikt 2 30. 0.;
  NJetFinder 2 30. 0. 0.4 -1;
  PT   11     25. E_CMS;
  PT  -11     25. E_CMS;
  PT   13     25. E_CMS;
  PT  -13     25. E_CMS;
  Mass 11 -11 60. 120.;
  Mass 13 -13 60. 120.;
}(selector);
```

Complete example

Setup for $pp \to \mu^-\bar{\nu}_\mu + 0,1,2j$@NLO$,3,4,5j$@LO in MEPS@NLO

```
(run){
  % general setting
  EVENTS 1M; ERROR 0.1;

  % scales, tags for scale variations
  FSF:=1.; RSF:=1.; QSF:=1.;
  SCALES STRICT_METS{FSF*MU_F2}{RSF*MU_R2}{QSF*MU_Q2};

  % tags for process setup
  NJET:=5; LJET:=2,3,4; QCUT:=20.;

  % me generator settings
  ME_SIGNAL_GENERATOR Comix Amegic LOOPGEN;
  EVENT_GENERATION_MODE Weighted;
  LOOPGEN:=OpenLoops;

  % collider setup
  BEAM_1 2212; BEAM_ENERGY_1 6500.;
  BEAM_2 2212; BEAM_ENERGY_2 6500.;
}(run)
```

```
(processes){
  Process 93 93 -> 13 -14 93{NJET};
  Order (*,2); CKKW sqr(QCUT/E_CMS);
  NLO_QCD_Mode MC@NLO {LJET};
  ME_Generator Amegic {LJET};
  RS_ME_Generator Comix {LJET};
  Loop_Generator LOOPGEN {LJET};
  End process;
}(processes)

(selector){
  Mass 13 -14 1. E_CMS
}(selector)
```

Complete example

Setup for $pp \to h\,[\to \gamma\gamma] + 0, 1j$@NLO$, 2j$@LO (ggF) with quark mass effects in MEPS@NLO

```
(run){
  % general settings
  EVENTS 5M; ERROR 0.1;

  % tags and settings for scale definitions
  FSF:=1.0; RSF:=1.0; QSF:=1.0;
  SCALES STRICT_METS{FSF*MU_F2}{RSF*MU_R2}{QSF*MU_Q2};

  % tags for process setup
  LJET:=1,2; NJET:=1; QCUT:=20.;

  % tags and settings for ME generators
  ME_SIGNAL_GENERATOR Amegic Internal OpenLoops;
  EVENT_GENERATION_MODE Weighted;

  % collider setup
  BEAM_1 2212; BEAM_ENERGY_1 6500;
  BEAM_2 2212; BEAM_ENERGY_2 6500;

  % finite top mass effects
  KFACTOR GGH;
  OL_IGNORE_MODEL 1;
  OL_PARAMETERS preset 2 allowed_libs pph2,pphj2,pphjj2 psp_tolerance 1.0e-7;

  % settings for hard decays
  HARD_DECAYS On;
  HDH_STATUS[25,22,22] 2;
  HDH_BR_WEIGHTS 0;

  % model parameters
  MODEL HEFT;
  MASS[5]    4.5;
  MASS[6]  173.2; WIDTH[6]  1.5;
  MASS[25] 125.0; WIDTH[25] 0.;
}(run);

(processes){
  Process 93 93 -> 25 93{NJET};
  Order (*,0,1); CKKW sqr(QCUT/E_CMS);
  NLO_QCD_Mode MC@NLO {LJET};
  Loop_Generator Internal;
  Enhance_Function VAR{log(PPerp(p[2]))
  End process;
}(processes);
```

Expandability

Most modules and all selectors, scale setters, K-factors, etc. in SHERPA are loaded dynamically at run time. Thus, SHERPA can be extended in a very simple way.

For example, if you wanted to use a different way of setting the scales you would not need to modify your SHERPA installation. Simply provide a shared library that contains a class that inherits from SHERPA's `Scale_Setter_Base` to implement the respective interface functions etc. a hook and optionally some argument read-in. Then SHERPA only needs to be instructed to load this library at run time and use your new scale setter.

Example:

```
(run){
  ...

  SHERPA_LDADD MyScaleSetter; % loads libMyScaleSetter.so
  SCALES MyScales;

  ...
}(run)
```