MC-generators

Lectures and laboratories

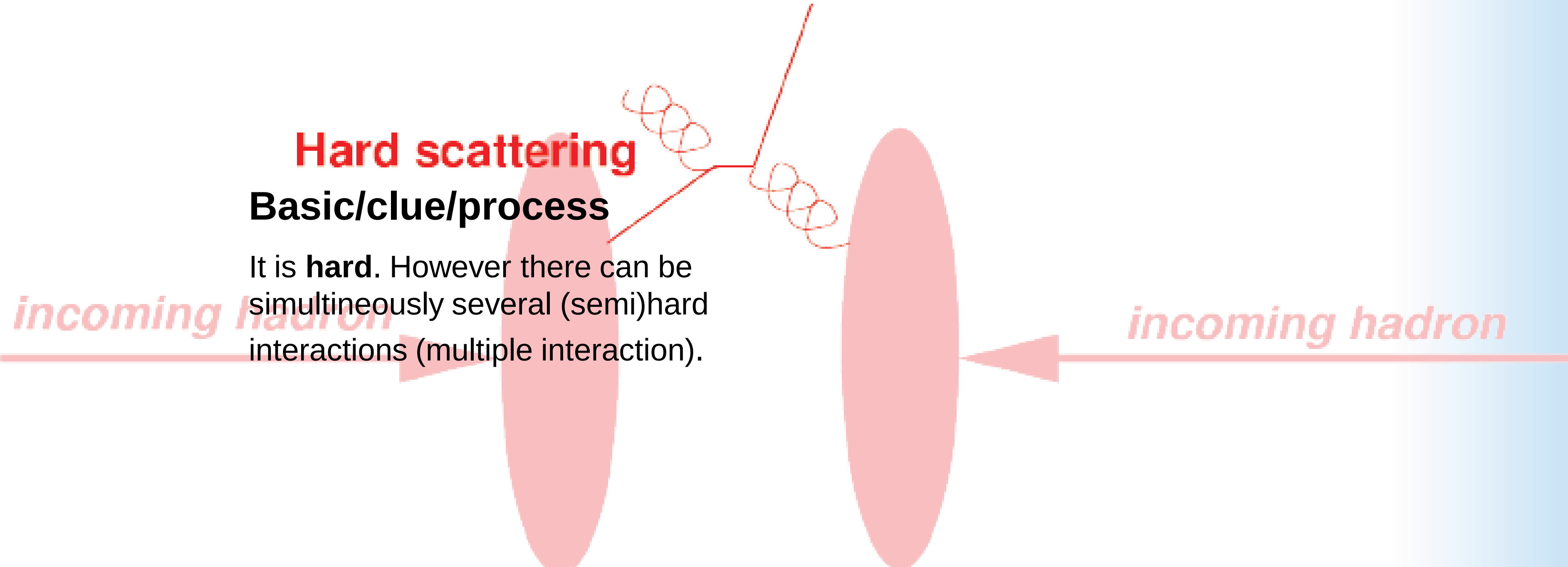
Physics of the process

Physics of the process:

“event topology” (basic terminology)

Physics of the process:

“event topology” (basic terminology)

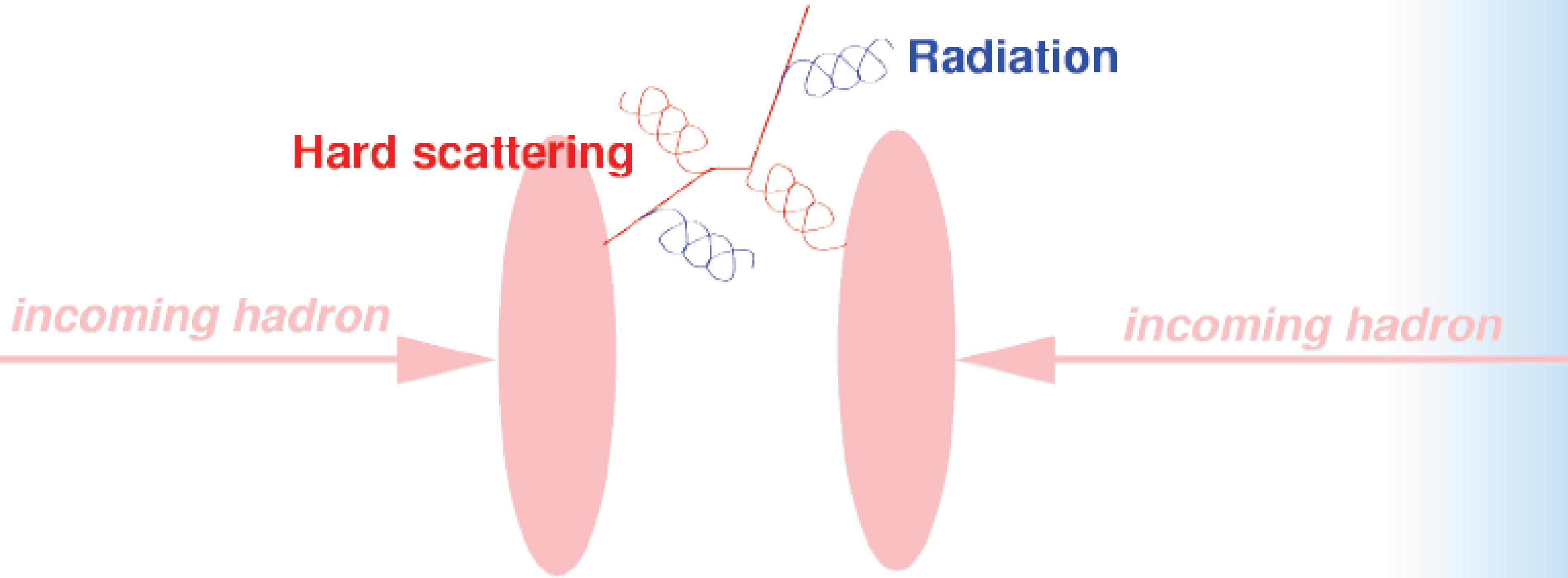


Basic/clue/process

It is **hard**. However there can be simultineously several (semi)hard interactions (multiple interaction).

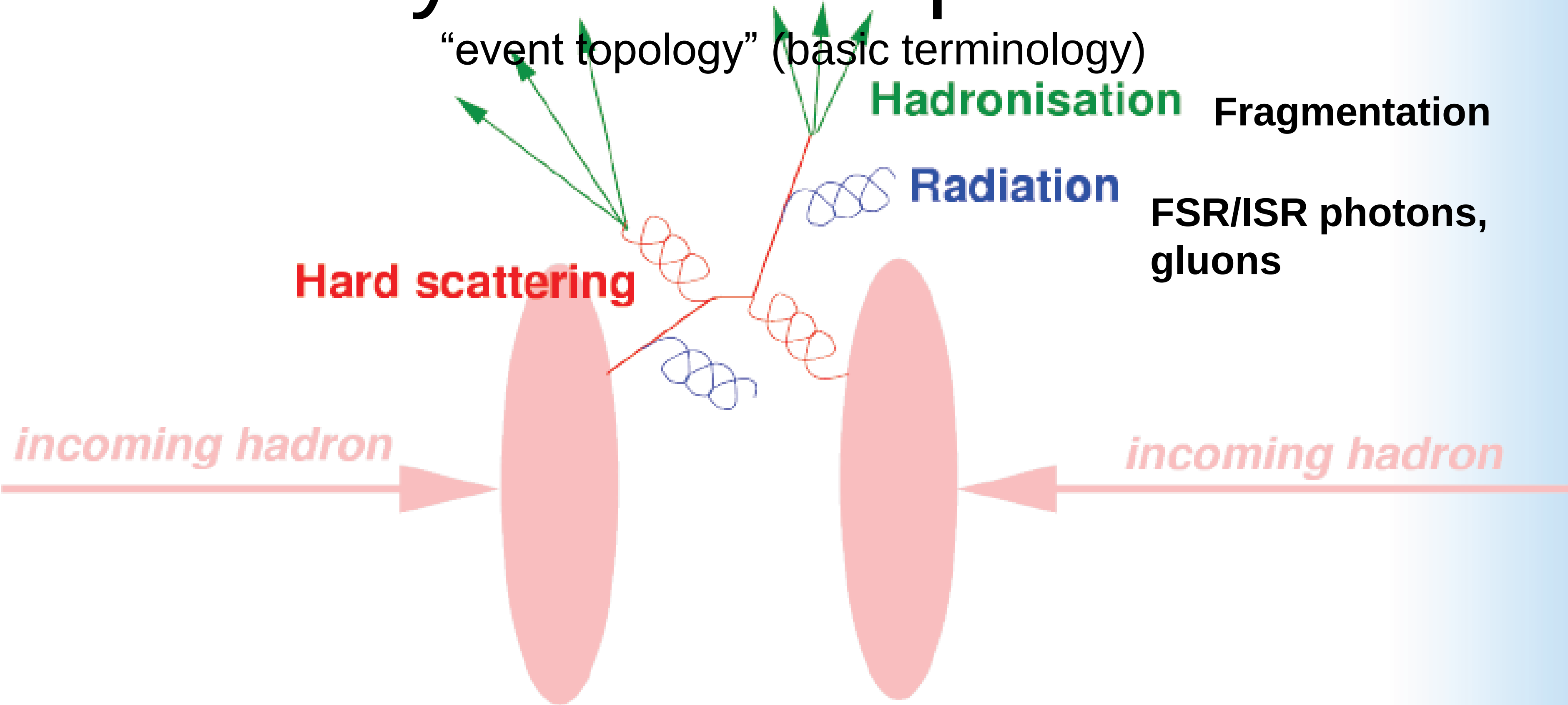
Physics of the process:

“event topology” (basic terminology)



Physics of the process:

"event topology" (basic terminology)

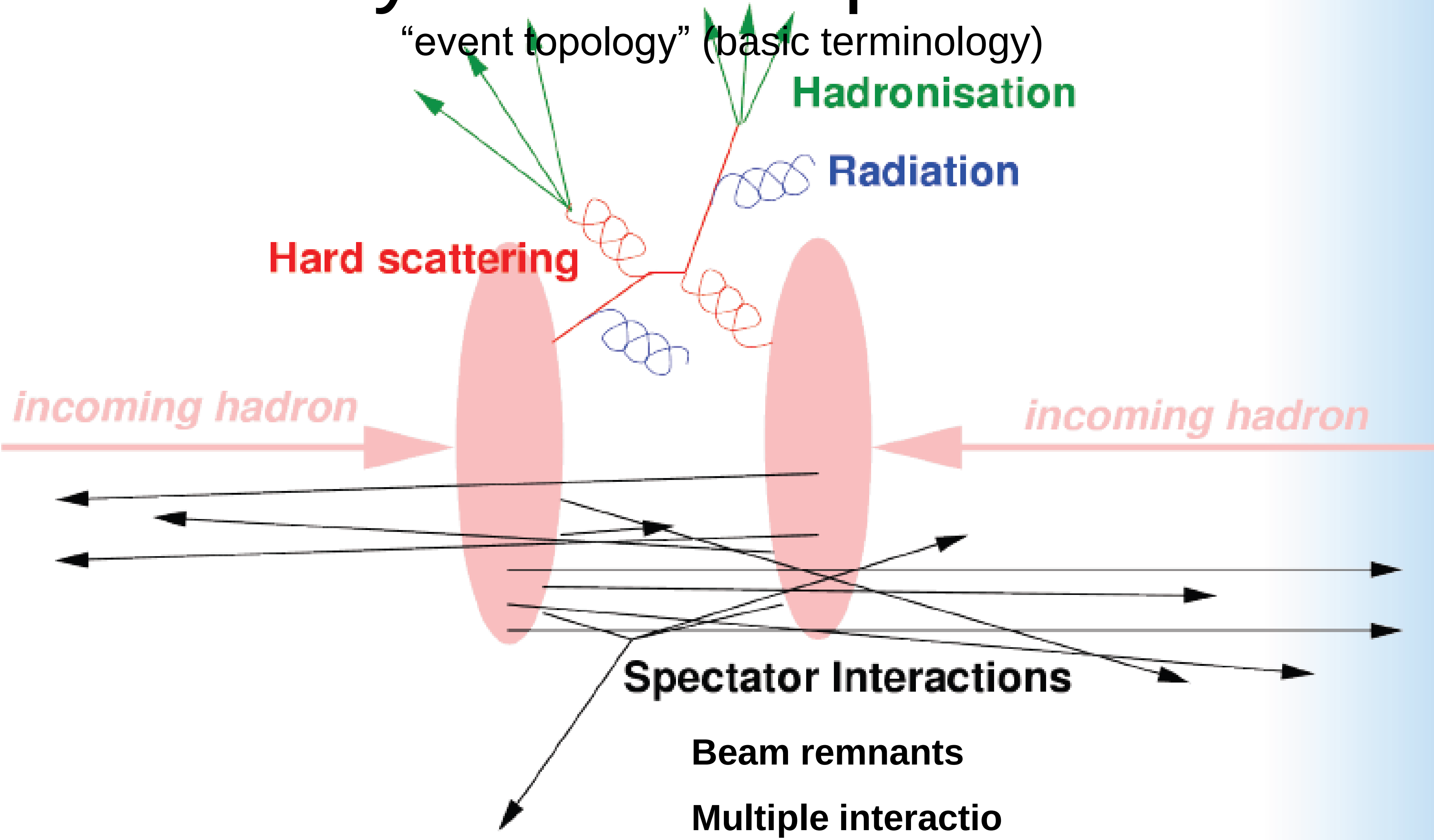


Fragmentation

FSR/ISR photons, gluons

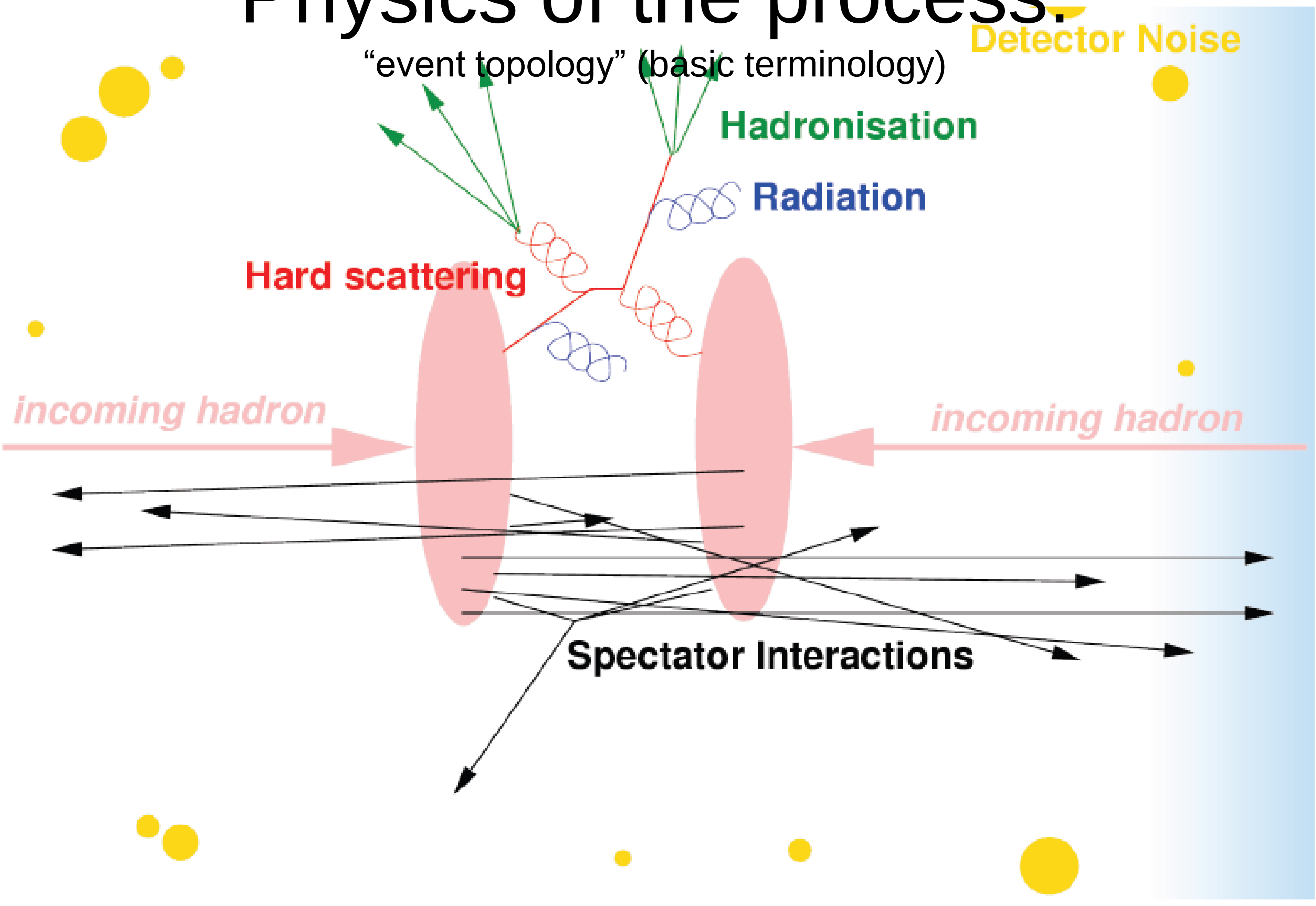
Physics of the process:

"event topology" (basic terminology)



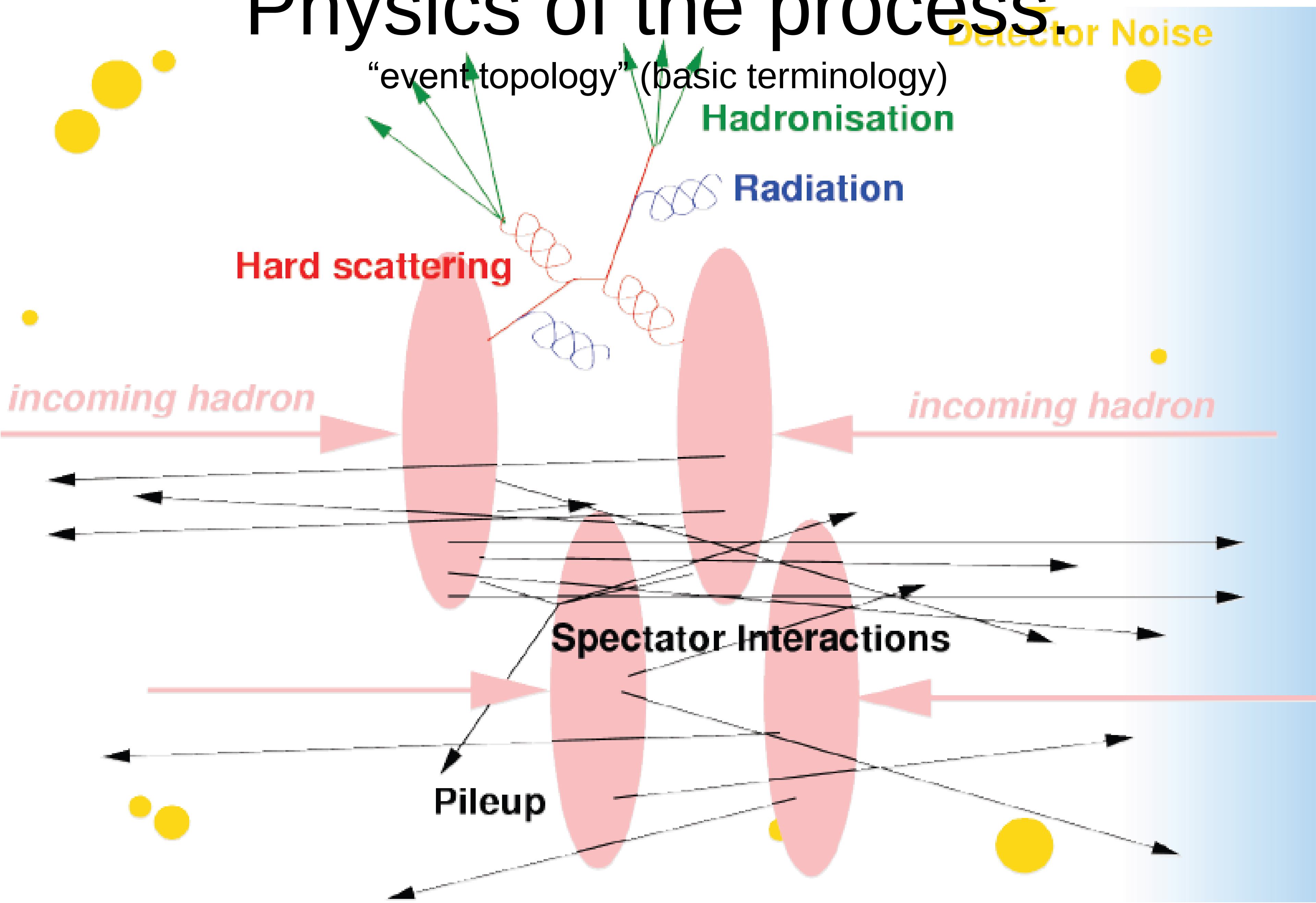
Physics of the process:

“event topology” (basic terminology)



Physics of the process:

"event topology" (basic terminology)



Underlying events

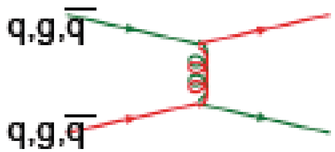
Everything excluding the basic (hard) process:

- **FSR/ISR (?)**
- **beam remnants**
- **pile-up events (several pp-collisions in one bunch crossing)**
- **multiple interaction (several (semi)hard parton-parton collision in one pp-collision)**
- **detector noise**

«Minimum (zero) bias events» contain the most of underlying events

«Hard» collisions

QCD 2→2 processes



$$\mathrm{d}\hat{\sigma}/\mathrm{d}p_\perp^2 \approx 1/p_\perp^4 \text{ for } p_\perp \to 0.$$

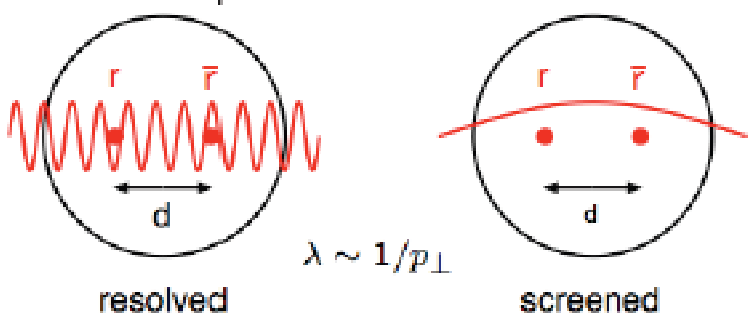
$$\to \frac{\alpha_s^2(p_{\perp 0}^2 + p_\perp^2)}{(p_{\perp 0}^2 + p_\perp^2)^2}$$

Naively breakdown at

$$p_{\perp\mathrm{min}} \simeq \frac{\hbar}{r_\mathrm{p}} \approx \frac{0.2\ \mathrm{GeV}\cdot\mathrm{fm}}{0.7\ \mathrm{fm}} \approx 0.3\ \mathrm{GeV} \simeq \Lambda_\mathrm{QCD}$$

$$\sigma_{hard}(p_{T,min}) = \int_{p_{T,min}^2}^{s/4} \frac{d\sigma}{dp_T^2} dp_T^2$$

Replace r_p by color screening length d



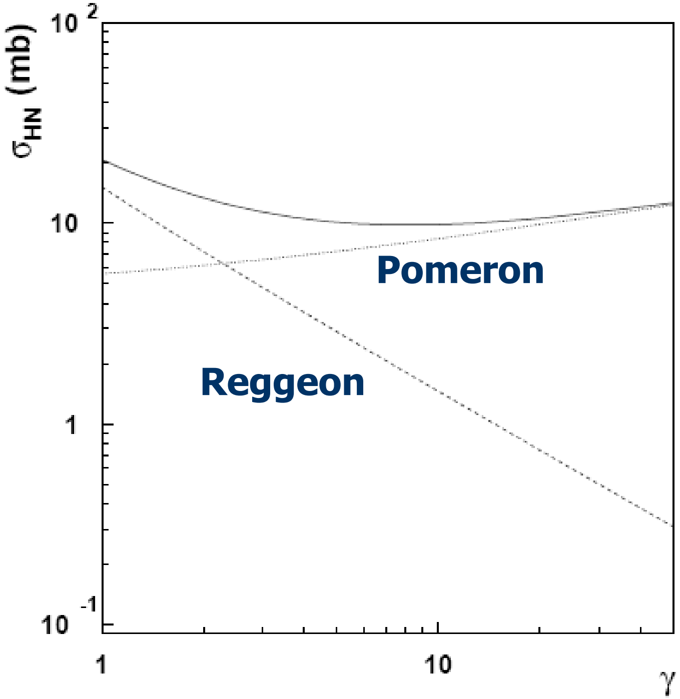
p_{tmin} tuneable

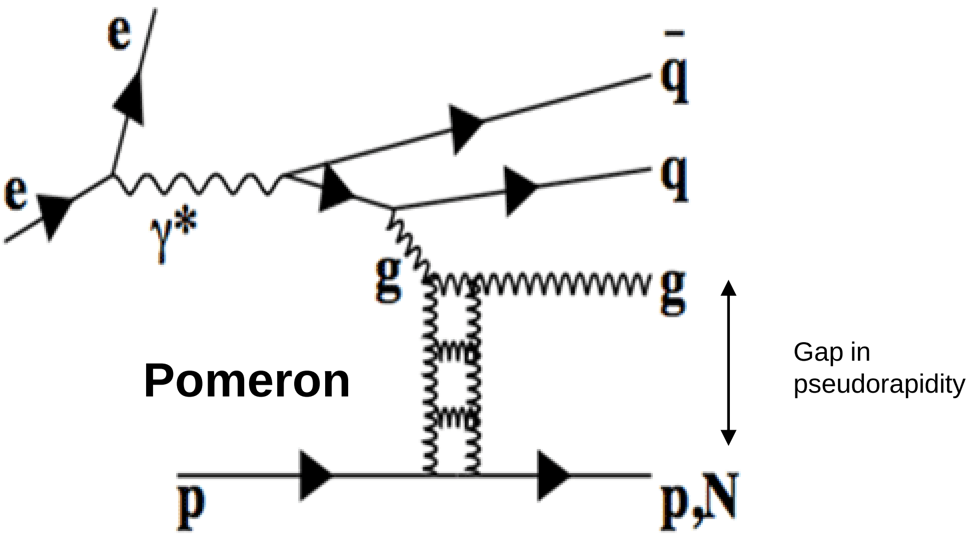
p_{tmin} ~ 2 GeV @ Tevatron

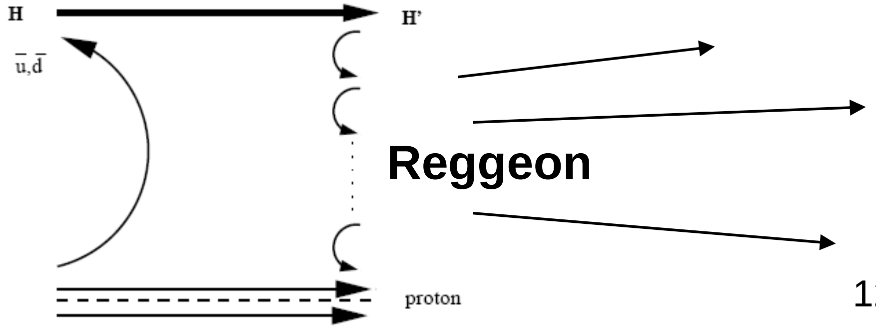
$\langle n \rangle = \sigma_{nd} / \sigma_{hard}$ n is then assumed to be Poisson-distributed

(Multiple interactions)

Diffracrive interactions





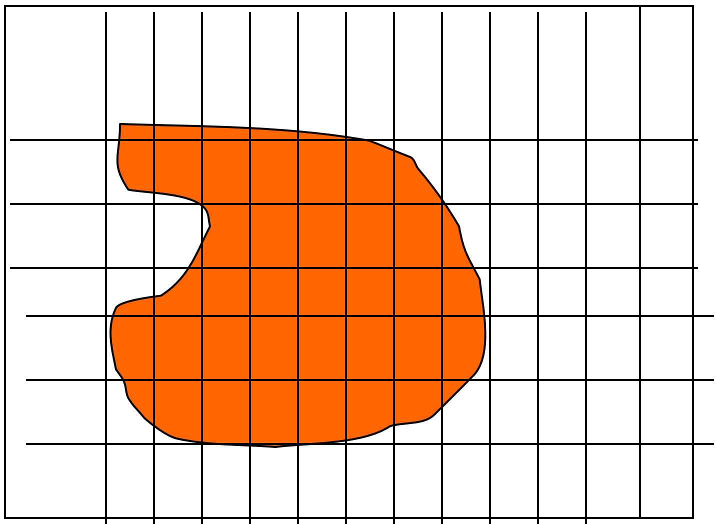


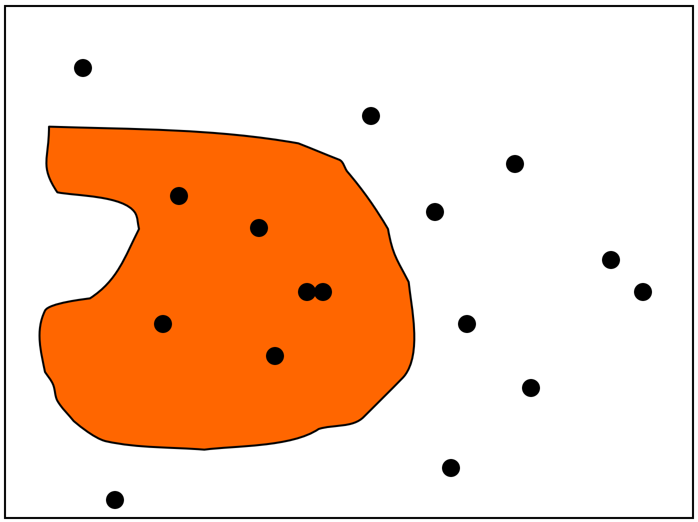
Monte-Carlo method

The simplest example of MC method application: calculation of the area of figure

- Cover the figure by a grid, calculate the number of grid cells which are inside and this gives you the area

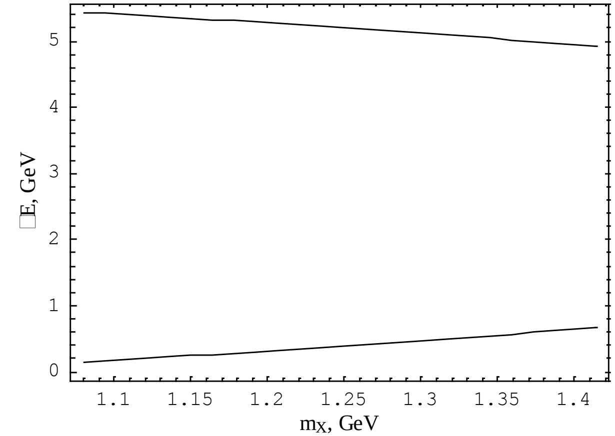
- Shoot at random at the figure. Count the bullets that hit it. The area of then figure is
- **S=(Nhit/Ntotal)*S(rectangle)**





One needs to know in advance the boundaries (maximums) within which the figure (function) is enclosed!

Example of MC-method application



Algorithm (method «**miss-or-hit**»):

1) simulation m_X: $\tilde{m}_X = m_{min} + r_1(m_{max} - m_{min})$

2) разыгрывается ΔE:

$$\Delta\tilde{E} = \Delta E_{min}(m_{min}) + r_2\left(\min\{\Delta E_{min}(m_{max}) + 5\cdot\Delta_{max}, \Delta E_{max}(m_{max})\} - \Delta E_{min}(m_{min})\right)$$

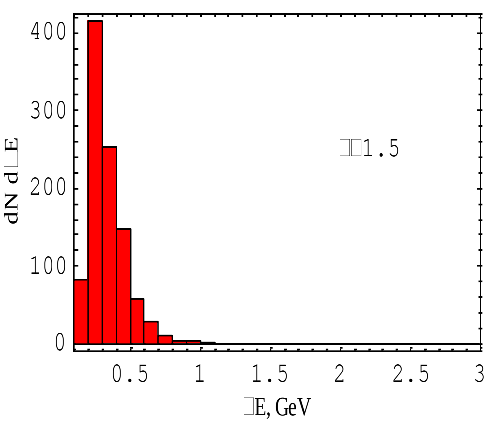
3) check: if

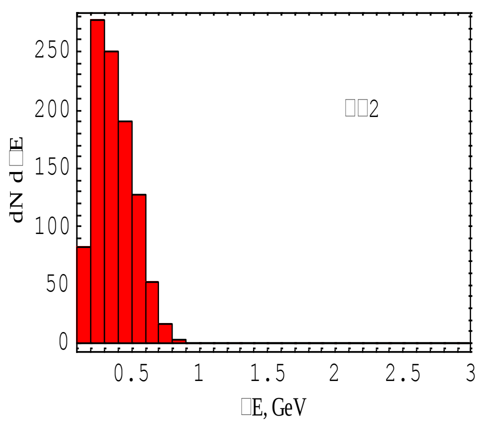
$\Delta\tilde{E} \notin \left[\Delta E_{min}(\tilde{m}_X), \Delta E_{max}(\tilde{m}_X)\right]$ **, то возврат на шаг 1)**

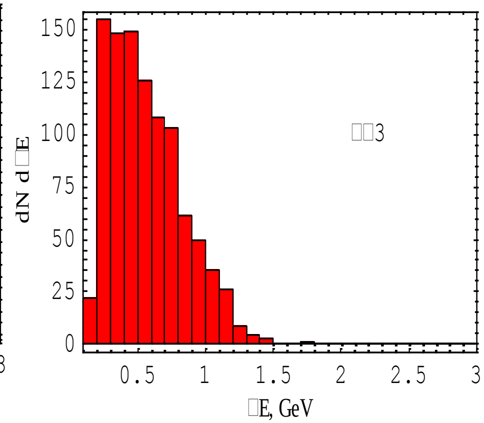
4) evaluation $\frac{dw}{d\Delta E dm_X}(\tilde{m}_X, \Delta\tilde{E})$

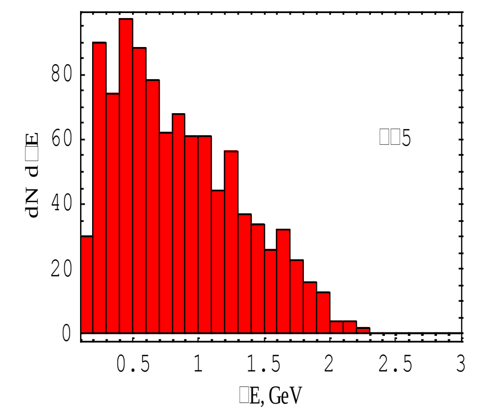
5) simulation $\frac{dw}{d\Delta E dm_X}$ $\frac{d\tilde{w}}{d\Delta E dm_X} = r_3 \frac{dw_{max}}{d\Delta E dm_X}$ **r_i – разные случайные числа от 0 до 1**

6) simulation: if $\frac{d\tilde{w}}{d\Delta E dm_X} < \frac{dw}{d\Delta E dm_X}(m_X, \Delta\tilde{E})$**, то событие принимается.**









- *<ΔE> if found to 20% less, than it gives direct integration => it is necessary to check a quality of MC-generator*
- *m_x is not specified*

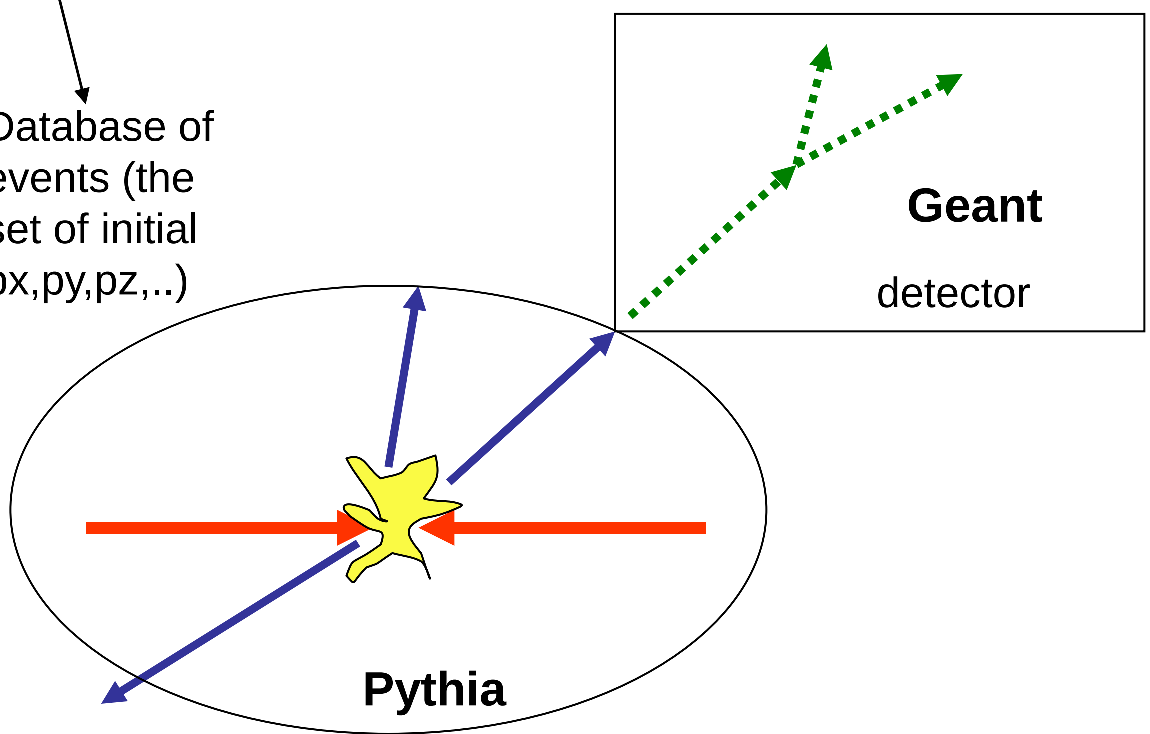
Application of MC-methods in HEP

Simulation (generation) of the basic interaction process (Pythia,…)

Simulation of the processes in detector (Geant)

Cross section

Database of events (the set of initial px,py,pz,..)



MC-generators

Multi-purpse generators

- Pythia
- fHerwig
- Pythia8
- Herwig++
- Sherpa

IsaJet

ME generators

- Alpgen
- Whizard
- AcerMC
- Grappa
- Amegic++
- MadGraph
- Helac/Phegas
- CompHep....

Testing & Tuning tools

- MCTester
- Rivet
- HepMCAnalysisTool
- MatchChecker
- Professor
- Proffit

Special purpose generators

- Tauola
- Photos
- EvtGen

NLO calculators

- MCFM
- NLOJET++
- BlackHat
- Rocket
- CutTools

NLO generators

- MC@NLO
- POWHEG
- SANC
- Grace (@NLO)

alpgen generation of hard multiparton processes in hadronic collisions
blackmax a black-hole event generator, which simulates the experimental signature of microscopic and Planckian black-hole production and evolution at the LHC in the context of brane-world models with low-scale quantum gravity.
cascade full event generator for ep and pp scattering applying kt-factorisation and unintegrated PDFs
charybdis production and decay of black holes in hadron collider experiments
charybdis2 new version of the black hole event generator
hej a High Energy Physics Monte Carlo generator for multi-jet analyses
herwig an event generator for Hadron Emission Reactions With Interfering Gluons (including SUSY processes)
herwig++ new event generator, written in C++ and built on the experience gained with HERWIG
hijing event generator for high energy heavy ion collisions
Hydjet event generator simulating jet production, jet quenching and flow effects in ultrarelativistic heavy ion collisions
isajet event renerator for pp, pbar p, and e+e- Interactions
jimmy a library for multiparton interactions in HERWIG
MCatNLO a Fortran package for combining a Monte Carlo event generator with NLO calculations of rates for QCD processes
MCFM Monte Carlo for FeMtobarn processes
nlojet++ LO and NLO cross section calculation, using the Catani-Seymour dipole subtraction method
Phojet general purpose event generator based on Dual Parton Model
photos package for QED radiative corrections in decays of resonances
photos++ C++ Interface to PHOTOS
Pomwig Herwig for diffractive interactions
pyquen event generator for simulation of rescattering, radiative and collisional energy loss of hard partons in expanding quark-gluon plasma created in ultrarelativistic heavy ion collisions
pythia6 The Lund Monte Carlo - general purpose HEP event generator (written in Fortran)
Pythia8 C++ version of pythia6 MC
Sherpa an event-generation framework for high-energy particle collisions
stagen generator for black hole and graviton production
starlight Monte Carlo generator for two-photon and photon-Pomeron interactions between relativistic nuclei and protons.
tauola generation of tau decays including spin polarization
Toprex parton level generator for top quark production
vincia a shower plugin to PYTHIA 8. It is based on the dipole-antenna picture of QCD and focusses on describing quark and gluon radiation with high precision.
winhac event generator for single W-boson production in hadron collisions

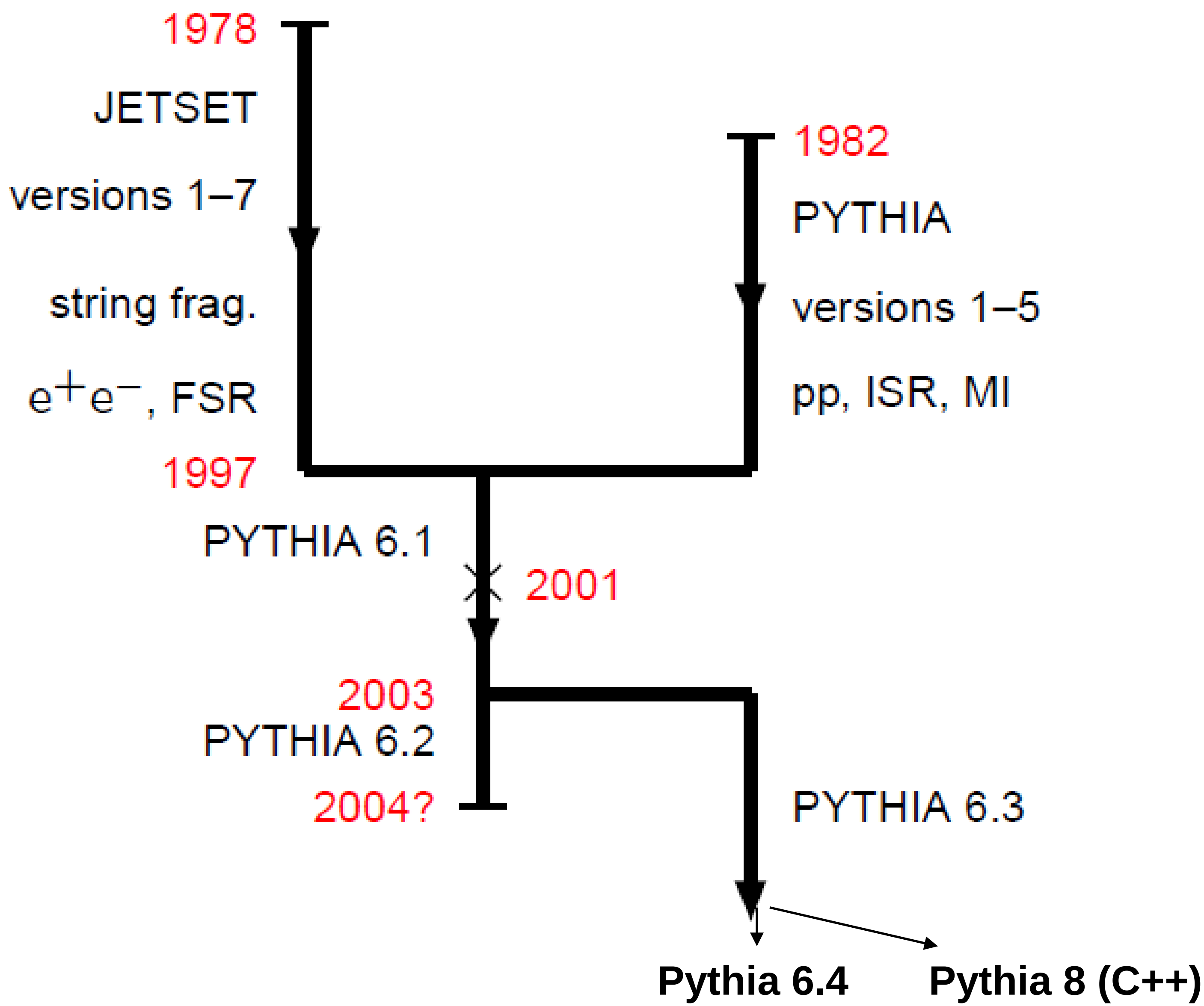
AGILe A Generator Interface Library (& executable) which provide a standard steering interface for Fortran generator codes
CLHEP a Class Library for High Energy Physics (No further development is foreseen, except for bug fixes)
FastJet a fast and efficient implementation of various jet algorithms
JetWeb web-based system for MC event generator validation
LHAPDF the Les Houches Accord PDF Interface
HepMC a C++ Event Record for Monte Carlo Generators
HepPDT particle properties from PDG
HepMCAnalysis a tool for generator validation and comparisons
HZtool robust model-to-data comparisons
MC-tester a universal tool for comparisons of Monte Carlo predictions
Professor A tuning tool for Monte Carlo event generators
Rivet a toolkit for validation of Monte Carlo event generators (an object oriented C++ replacement for the Fortran HZtool package)
ThePEG a toolkit for High Energy Physics Event Generation

Pythia 6.4

http://home.thep.lu.se/~torbjorn/Pythia.html

http://projects.hepforge.org/pythia6/

PYTHIA history



Pythia vs other generators

Kind	Process	PYT	HER	ISA
QCD & related	Soft QCD	$\star$	$\star$	$\star$
	Hard QCD	$\star$	$\star$	$\star$
	Heavy flavour	$\star$	$\star$	$\star$
Electroweak SM	Single $\gamma^*/Z^0/W^\pm$	$\star$	$\star$	$\star$
	$(\gamma/\gamma^*/Z^0/W^\pm/f/g)^2$	$\star$	$\star$	$\star$
	Light SM Higgs	$\star$	$\star$	$\star$
	Heavy SM Higgs	$\star$	$\star$	$\star$
SUSY BSM	$h^0/H^0/A^0/H^\pm$	$\star$	$\star$	$\star$
	SUSY	$\star$	$\star$	$\star$
	$\not{R}$ SUSY	$\star$	$\star$	—
Other BSM	Technicolor	$\star$	—	$(\star)$
	New gauge bosons	$\star$	—	—
	Compositeness	$\star$	—	—
	Leptoquarks	$\star$	—	—
	$H^{\pm\pm}$ (from LR-sym.)	$\star$	—	—
	Extra dimensions	$(\star)$	$(\star)$	$(\star)$

Some shortcomings Pythia

$\star$ Processes usually only in lowest nontrivial order
$\Rightarrow$ need programs that include HO loop corrections to cross sections, alternatively do $(p_\perp, y)$-dependent rescaling by hand?

$\star$ No multijet topologies
$\Rightarrow$ have to trust shower to get it right, alternatively match to HO (non-loop) ME generators

$\star$ Spin correlations often absent or incomplete
e.g. top produced unpolarized, while $\mathrm{t} \to \mathrm{bW}^+ \to \mathrm{b}\ell^+\nu_\ell$ decay correct
$\Rightarrow$ have to use external programs when important

$\star$ New physics scenarios appear at rapid pace
$\Rightarrow$ need to have a bigger class of “one-issue experts” contributing code

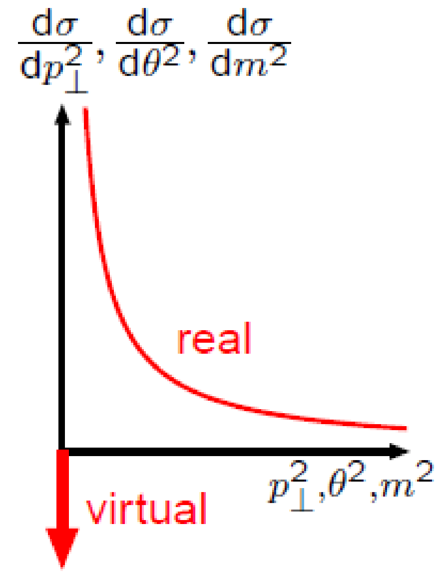
always remember that the program does not represent a dead collection of established truths, but rather one of many possible approaches to the problem of multiparticle production in high-energy physics, at the frontline of current research. Be critical! (Manual on Pythia 6.4)

Physics models implemented in Pythia

Matrix Elements vs. Parton Showers

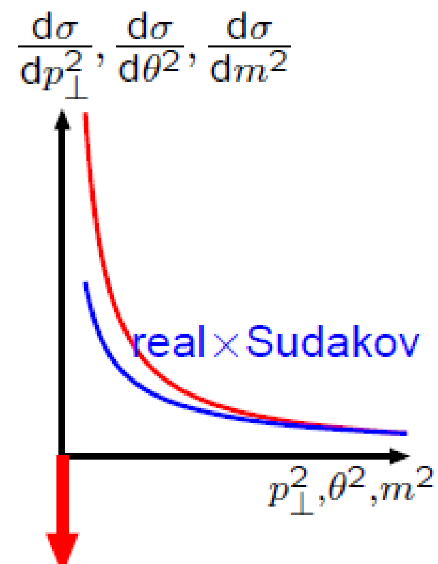
ME : Matrix Elements

+ systematic expansion in α_s (‘*exact*’)
+ powerful for multiparton Born level
+ flexible phase space cuts
– loop calculations very tough
– negative cross section in collinear regions
  $\Rightarrow$ unpredictive jet/event structure
– *no easy match to hadronization*



PS : Parton Showers

– approximate, to LL (or NLL)
– main topology not predetermined
  $\Rightarrow$ inefficient for exclusive states
+ process-generic $\Rightarrow$ simple multiparton
+ Sudakov form factors/resummation
  $\Rightarrow$ sensible jet/event structure
+ *easy to match to hadronization*



Physics models implemented in Pythia

Parton Shower Approach

(already well covered by B.R. Webber, so brief here)

3 common algorithms, each with its advantages and disadvantages:

HERWIG: θ-ordered emissions (ISR & FSR)

ARIADNE: $p_\perp$-ordered emissions (FSR primarily)

PYTHIA: M^2, Q^2-ordered emissions (ISR & FSR)

New in PYTHIA 6.3: $p_\perp$-ordered emissions (ISR & FSR)

Matrix Elements and Parton Showers

Marriage desirable! But how?

Much work ongoing $\Longrightarrow$ no established orthodoxy

Three main areas, in ascending order of complication:

1) Match to lowest-order nontrivial process — merging

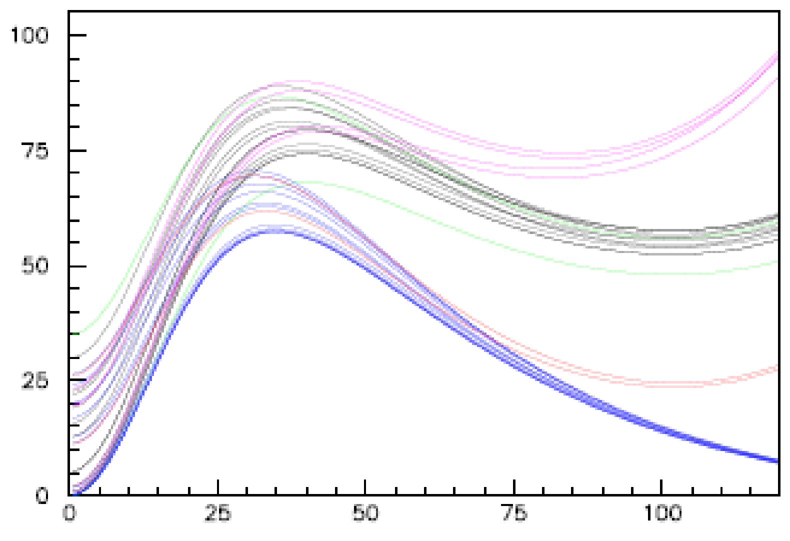
2) Combine leading-order multiparton process — vetoed parton showers

3) Match to next-to-leading order process — MC@NLO

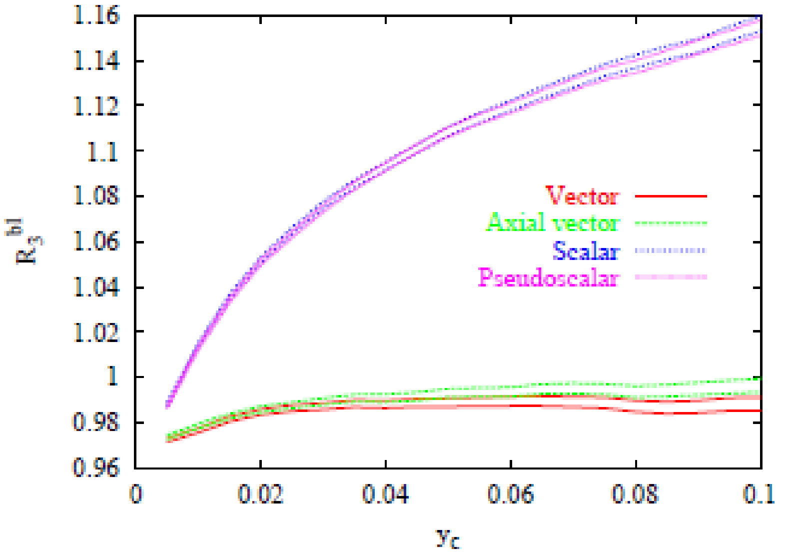
Physics models implemented in Pythia

PYTHIA performs merging with generic FSR $a \to bc$g ME,
in SM: $\gamma^*/\mathrm{Z}^0/\mathrm{W}^\pm \to \mathrm{q}\overline{\mathrm{q}}$, $\mathrm{t} \to \mathrm{bW}^+$, $\mathrm{H}^0 \to \mathrm{q}\overline{\mathrm{q}}$,
and MSSM: $\mathrm{t} \to \mathrm{bH}^+$, $\mathrm{Z}^0 \to \tilde{\mathrm{q}}\overline{\tilde{\mathrm{q}}}$, $\tilde{\mathrm{q}} \to \tilde{\mathrm{q}}'\mathrm{W}^+$, $\mathrm{H}^0 \to \tilde{\mathrm{q}}\overline{\tilde{\mathrm{q}}}$, $\tilde{\mathrm{q}} \to \tilde{\mathrm{q}}'\mathrm{H}^+$,
$\chi \to \mathrm{q}\overline{\tilde{\mathrm{q}}}$, $\chi \to \mathrm{q}\overline{\tilde{\mathrm{q}}}$, $\tilde{\mathrm{q}} \to \mathrm{q}\chi$, $\mathrm{t} \to \tilde{\mathrm{t}}\chi$, $\tilde{\mathrm{g}} \to \mathrm{q}\overline{\tilde{\mathrm{q}}}$, $\tilde{\mathrm{q}} \to \mathrm{q}\tilde{\mathrm{g}}$, $\mathrm{t} \to \tilde{\mathrm{t}}\tilde{\mathrm{g}}$

g emission for different colour, spin and parity:



$R_3^{\mathrm{bl}}(y_c)$: mass effects in Higgs decay:

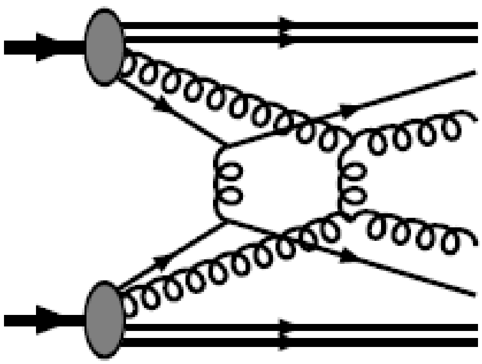


PYTHIA ISR: only $\mathrm{q}\overline{\mathrm{q}} \to \gamma^*/\mathrm{Z}^0/\mathrm{W}^\pm$ and $\mathrm{gg} \to \mathrm{H}^0$ (for $m_\mathrm{t} \to \infty$)
(but K factor not implemented here)

Physics models implemented in Pythia

Multiple Interactions

Consequence of composite nature of hadrons!

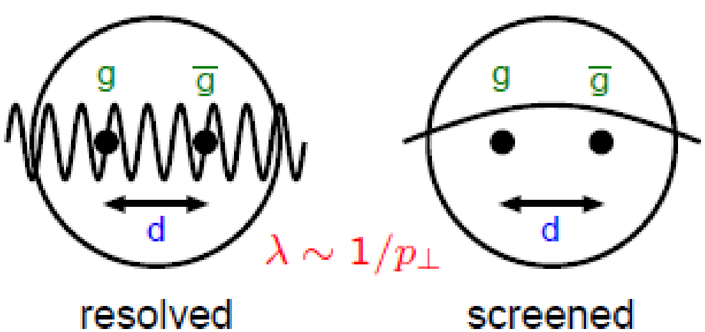


Evidence:

- direct observation: AFS, UA1, CDF
- implied by width of multiplicity distribution + jet universality: UA5
- forward–backward correlations: UA5
- pedestal effect: UA1, H1, CDF

One new free parameter: $p_{\perp\mathrm{min}}$

$$\frac{1}{2}\sigma_{\mathrm{jet}} = \int_{p^2_{\perp\mathrm{min}}}^{s/4} \frac{\mathrm{d}\sigma}{\mathrm{d}p^2_\perp}\,\mathrm{d}p^2_\perp \quad \Longleftarrow \quad \int_0^{s/4} \frac{\mathrm{d}\sigma}{\mathrm{d}p^2_\perp}\,\frac{p^4_\perp}{(p^2_{\perp 0} + p^2_\perp)^2}\,\mathrm{d}p^2_\perp$$



Measure of colour screening length d in hadron:

$p_{\perp\mathrm{min}}\,\langle d \rangle \approx 1 (= \hbar)$

Physics models implemented in Pythia

Hadronization: Lund String Model

In QCD, for large charge separation, field lines seem to be compressed to tubelike region(s) $\Rightarrow$ **string(s)**

by self-interactions among soft gluons in the "vacuum".

(Analogy: vortex lines in type II superconductor)

Gives linear confinement with string tension:

$$F(r) \approx \text{const} = \kappa \approx 1\ \text{GeV/fm} \quad \Longleftrightarrow \quad V(r) \approx \kappa r$$

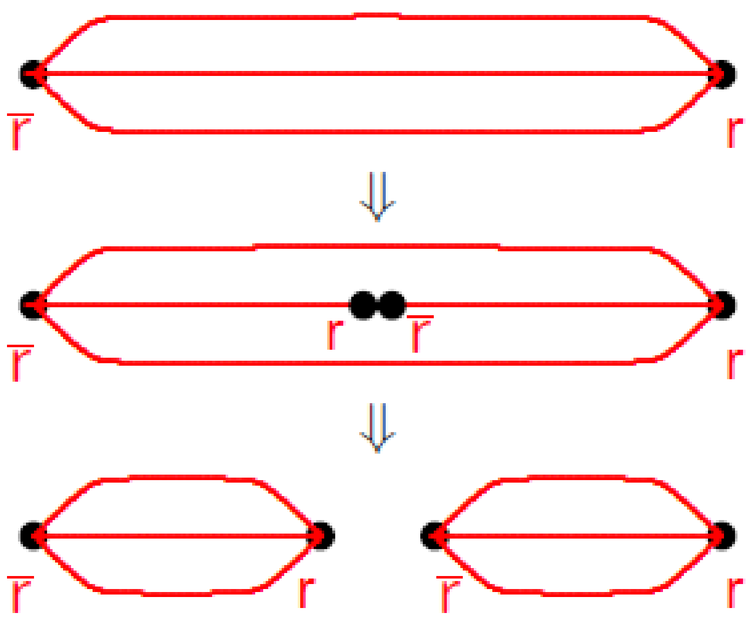
Confirmed e.g. by quenched lattice QCD

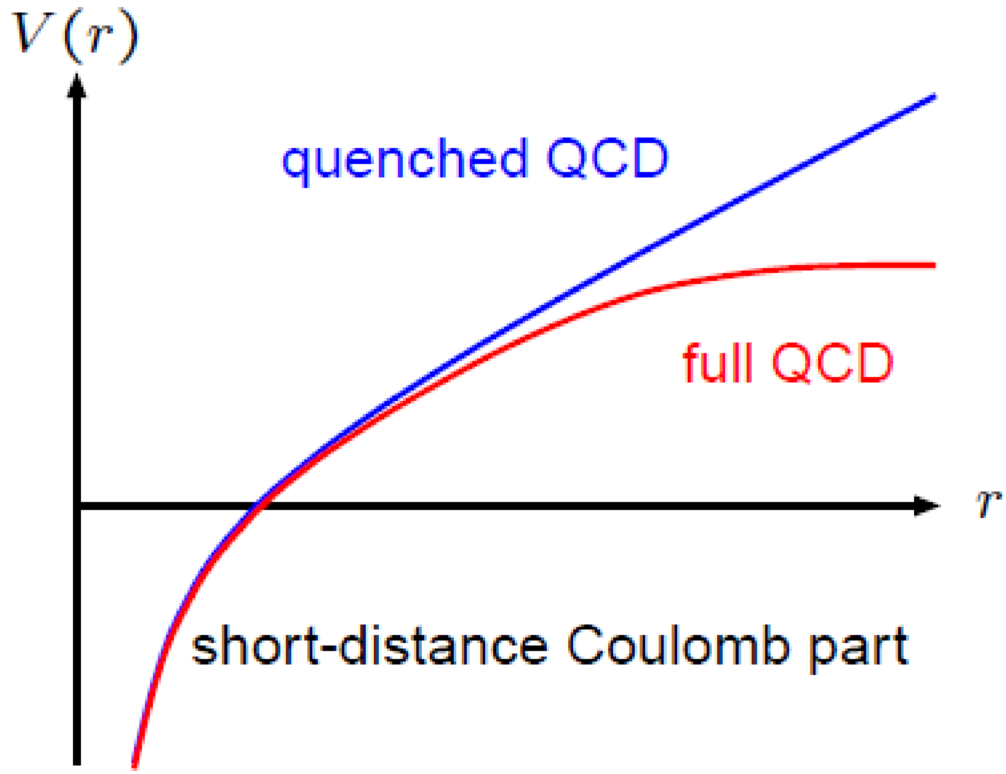
Physics models implemented in Pythia

Real world (??, or at least unquenched lattice QCD)

$\Longrightarrow$ nonperturbative string breakings $gg \ldots \to q\bar{q}$

simplified colour representation:





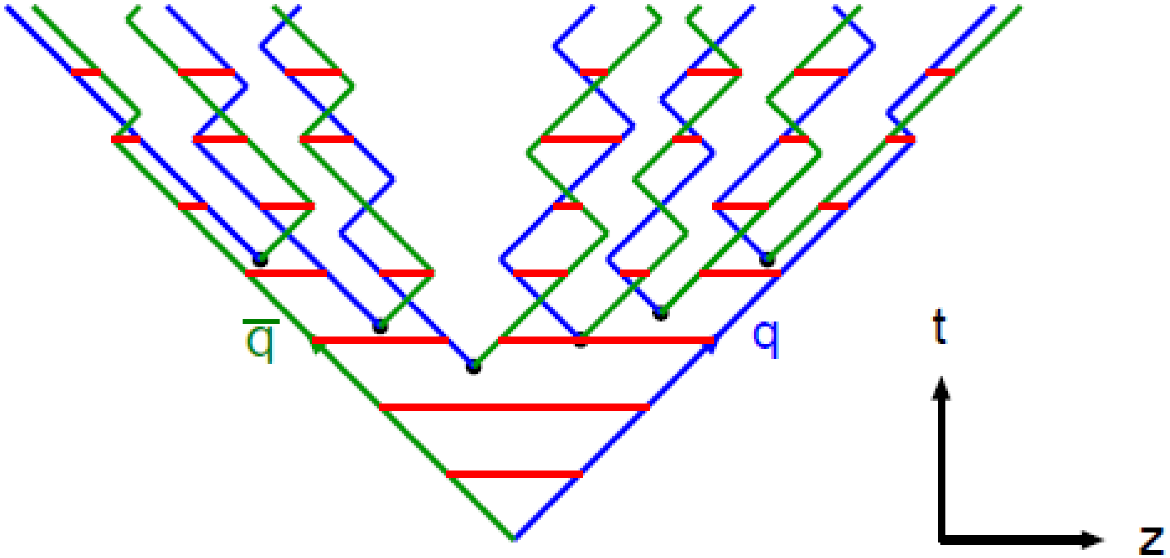
Physics models implemented in Pythia

Repeat for large system $\Rightarrow$ *Lund model*

which neglects Coulomb part:

$$\left|\frac{\mathrm{d}E}{\mathrm{d}x}\right| = \left|\frac{\mathrm{d}p}{\mathrm{d}x}\right| = \left|\frac{\mathrm{d}E}{\mathrm{d}t}\right| = \left|\frac{\mathrm{d}p}{\mathrm{d}t}\right| = \kappa$$

Motion of quarks and antiquarks in a $q\overline{q}$ system:

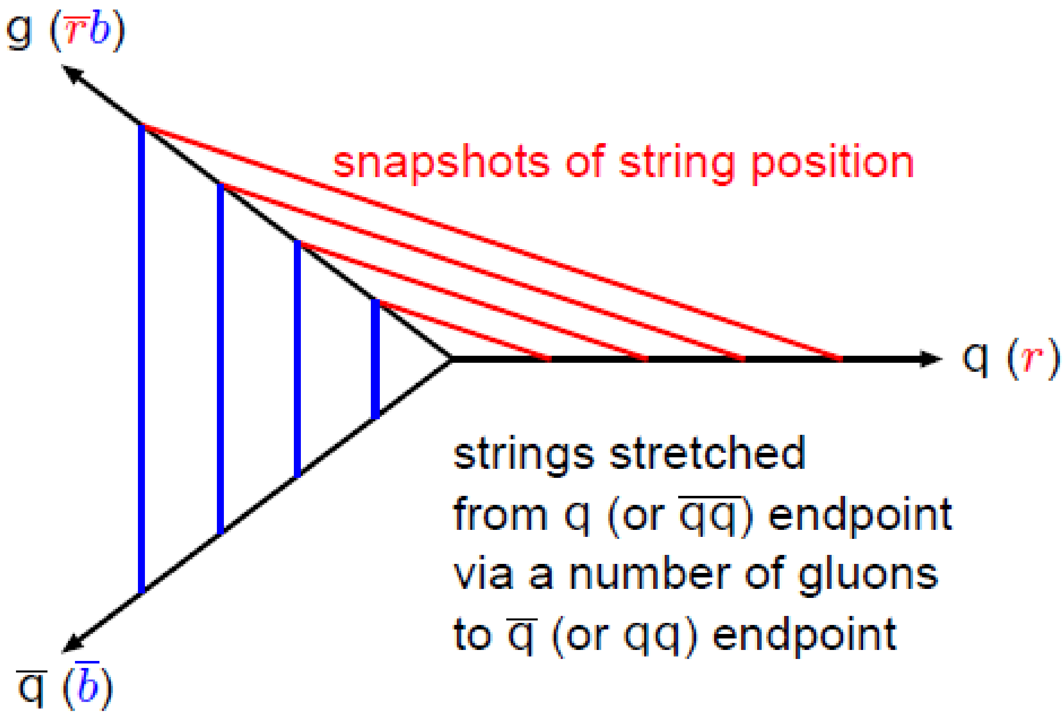


gives simple but powerful picture of hadron production

(with extensions to massive quarks, baryons, . . .)

Physics models implemented in Pythia

The Lund gluon picture — the most characteristic feature:



Gluon = kink on string, carrying energy and momentum.
Force ratio gluon/ quark = 2, cf. QCD $N_C/C_F = 9/4$

- Few parameters to describe energy–momentum structure!
- Many parameters to describe flavour composition!

HERWIG cluster fragmentation: the opposite

Numerous and detailed tests at LEP favour string picture
. . . but much uncertain when moving to hadron colliders.

Physics models implemented in Pythia

Summary (so far)

⋆ Big selection of subprocesses ⋆

but often not enough, so

⋆ Standard interface for including external processes ⋆

⋆ State-of-the-art parton showers ⋆

. . . but much to be done on ME + PS matching

⋆ *The* trend-setting model for underlying events ⋆

⋆ *The* most realistic and successful model for hadronization ⋆

⋆ Extensive documentation: 450 pp manual + update notes ⋆

⋆ Webpage `http://www.thep.lu.se/~torbjorn/Pythia.html` ⋆

Nobody ever got fired for using PYTHIA! (?)

but also

Nobody is perfect!

So, whichever is your favourite generator,

you always need cross-checks!

Physics models implemented in Pythia

[. . .] The Monte Carlo simulation has become the major means of visualization of not only detector performance but also of physics phenomena. So far so good. But it often happens that the physics simulations provided by the Monte Carlo generators carry the authority of data itself. They look like data and feel like data, and if one is not careful they are accepted as if they were data.

[. . .] I am prepared to believe that the computer-literate generation (of which I am a little too old to be a member) is in principle no less competent and in fact benefits relative to us in the older generation by having these marvelous tools. They do allow one to look at, indeed visualize, the problems in new ways. But I also fear a kind of "terminal illness", perhaps traceable to the influence of television at an early age. There the way one learns is simply to passively stare into a screen and wait for the truth to be delivered. A number of physicists nowadays seem to do just this.

J.D. Bjorken

from a talk given at the 75th anniversary celebration of the Max-Planck Institute of Physics, Munich, Germany, December 10th, 1992. As quoted in: Beam Line, Winter 1992, Vol. 22, No. 4

Nomenclature

KF (according to PDG) and KC codes

KF	Name	Printed	KF	Name	Printed
1	d	d	11	e^-	e-
2	u	u	12	ν_e	nu_e
3	s	s	13	μ^-	mu-
4	c	c	14	ν_μ	nu_mu
5	b	b	15	τ^-	tau-
6	t	t	16	ν_τ	nu_tau
7	b′	b'	17	τ'	tau'
8	t′	t'	18	ν'_τ	nu'_tau
9			19		
10			20		

KF	Name	Printed	KF	Name	Printed
21	g	g	31		
22	γ	gamma	32	Z'^0	Z'0
23	Z^0	Z0	33	Z''^0	Z"0
24	W^+	W+	34	W'^+	W'+
25	h^0	h0	35	H^0	H0
26			36	A^0	A0
27			37	H^+	H+
28			38		
29			39	G	Graviton
30			40		
			41	R^0	R0
			42	L_Q	LQ

KC = PYCOMP(KF)

KF = KCHG(KC,4)

Nomenclature

KF (according to PDG) and KC codes

$$\mathrm{KF} = 1000i + 100j + 10k + 2s + 1$$

KF	Name	Printed	KF	Name	Printed
			1114	Δ^-	Delta-
2112	n	n0	2114	Δ^0	Delta0
2212	p	p+	2214	Δ^+	Delta+
			2224	Δ^{++}	Delta++
3112	Σ^-	Sigma-	3114	Σ^{*-}	Sigma*-
3122	Λ^0	Lambda0			
3212	Σ^0	Sigma0	3214	Σ^{*0}	Sigma*0
3222	Σ^+	Sigma+	3224	Σ^{*+}	Sigma*+
3312	Ξ^-	Xi-	3314	Ξ^{*-}	Xi*-
3322	Ξ^0	Xi0	3324	Ξ^{*0}	Xi*0
			3334	Ω^-	Omega-
4112	Σ_c^0	Sigma_c0	4114	Σ_c^{*0}	Sigma*_c0
4122	Λ_c^+	Lambda_c+			
4212	Σ_c^+	Sigma_c+	4214	Σ_c^{*+}	Sigma*_c+
4222	Σ_c^{++}	Sigma_c++	4224	Σ_c^{*++}	Sigma*_c++
4132	Ξ_c^0	Xi_c0			
4312	$\Xi_c'^0$	Xi'_c0	4314	Ξ_c^{*0}	Xi*_c0
4232	Ξ_c^+	Xi_c+			
4322	$\Xi_c'^+$	Xi'_c+	4324	Ξ_c^{*+}	Xi*_c+
4332	Ω_c^0	Omega_c0	4334	Ω_c^{*0}	Omega*_c0
5112	Σ_b^-	Sigma_b-	5114	Σ_b^{*-}	Sigma*_b-
5122	Λ_b^0	Lambda_b0			
5212	Σ_b^0	Sigma_b0	5214	Σ_b^{*0}	Sigma*_b0
5222	Σ_b^+	Sigma_b+	5224	Σ_b^{*+}	Sigma*_b+

KF	Name	Printed
1000001	$\tilde{d}_L$	~d_L
1000002	$\tilde{u}_L$	~u_L
1000003	$\tilde{s}_L$	~s_L

$$\mathrm{KF} = 1000i + 100j + 2s + 1$$

KF	Name	Printed	KF	Name	Printed
			1103	dd_1	dd_1
2101	ud_0	ud_0	2103	ud_1	ud_1
			2203	uu_1	uu_1
3101	sd_0	sd_0	3103	sd_1	sd_1
3201	su_0	su_0	3203	su_1	su_1
			3303	ss_1	ss_1

$$\mathrm{KF} = \{100\max(i,j) + 10\min(i,j) + 2s + 1\}\ \mathrm{sign}(i-j)\,(-1)^{\max(i,j)}$$

KF	Name	Printed	KF	Name	Printed
211	π^+	pi+	213	ρ^+	rho+
311	K^0	K0	313	K^{*0}	K*0
321	K^+	K+	323	K^{*+}	K*+
411	D^+	D+	413	D^{*+}	D*+
421	D^0	D0	423	D^{*0}	D*0
431	D_s^+	D_s+	433	D_s^{*+}	D*_s+
511	B^0	B0	513	B^{*0}	B*0
521	B^+	B+	523	B^{*+}	B*+
531	B_s^0	B_s0	533	B_s^{*0}	B*_s0
541	B_c^+	B_c+	543	B_c^{*+}	B*_c+
111	π^0	pi0	113	ρ^0	rho0
221	η	eta	223	ω	omega
331	η'	eta'	333	ϕ	phi
441	η_c	eta_c	443	J/ψ	J/psi
551	η_b	eta_b	553	Υ	Upsilon
130	K_L^0	K_L0			
310	K_S^0	K_S0			

Output of the codes

```
C...Programme PyList.f
C...EXTERNAL statement links PYDATA on most machines.
      EXTERNAL PYDATA

C-------------------------------------------------------------

      CALL PYLIST(11)    !  PYLIST(12) – full table particle data
      END
```

gives KF

gives also IDC-codes of decay modes

Installation and running:

wget www.*/pythia64**.f** **or through psft**

g77 –c pythia64.o pythia64**.f**

pico PyList.f

g77 PyList.f pythia64.o**

./a.out > pylist12

Processes in Pythia (ISUB)

No.	Subprocess
	Hard QCD processes:
11	$f_i f_j \to f_i f_j$
12	$f_i \bar{f}_i \to f_k \bar{f}_k$
13	$f_i \bar{f}_i \to gg$
28	$f_i g \to f_i g$
53	$gg \to f_k \bar{f}_k$
68	$gg \to gg$
	Soft QCD processes:
91	elastic scattering
92	single diffraction (XB)
93	single diffraction (AX)
94	double diffraction
95	low-$p_\perp$ production
	Open heavy flavour:
	(also fourth generation)
81	$f_i \bar{f}_i \to Q_k \bar{Q}_k$
82	$gg \to Q_k \bar{Q}_k$
83	$q_i f_j \to Q_k f_l$

No.	Subprocess
84	$g\gamma \to Q_k \bar{Q}_k$
85	$\gamma\gamma \to F_k \bar{F}_k$
	Closed heavy flavour:
86	$gg \to J/\psi g$
87	$gg \to \chi_{0c} g$
88	$gg \to \chi_{1c} g$
89	$gg \to \chi_{2c} g$
104	$gg \to \chi_{0c}$
105	$gg \to \chi_{2c}$
106	$gg \to J/\psi \gamma$
107	$g\gamma \to J/\psi g$
108	$\gamma\gamma \to J/\psi \gamma$
	W/Z production:
1	$f_i \bar{f}_i \to \gamma^*/Z^0$
2	$f_i \bar{f}_j \to W^\pm$
22	$f_i \bar{f}_i \to Z^0 Z^0$
23	$f_i \bar{f}_j \to Z^0 W^\pm$
25	$f_i \bar{f}_i \to W^+ W^-$
15	$f_i \bar{f}_i \to g Z^0$

No.	Subprocess
16	$f_i \bar{f}_j \to g W^\pm$
30	$f_i g \to f_i Z^0$
31	$f_i g \to f_k W^\pm$
19	$f_i \bar{f}_i \to \gamma Z^0$
20	$f_i \bar{f}_j \to \gamma W^\pm$
35	$f_i \gamma \to f_i Z^0$
36	$f_i \gamma \to f_k W^\pm$
69	$\gamma\gamma \to W^+ W^-$
70	$\gamma W^\pm \to Z^0 W^\pm$
	Prompt photons:
14	$f_i \bar{f}_i \to g\gamma$
18	$f_i \bar{f}_i \to \gamma\gamma$
29	$f_i g \to f_i \gamma$
114	$gg \to \gamma\gamma$
115	$gg \to g\gamma$
	Deeply Inel. Scatt.:
10	$f_i f_j \to f_k f_l$
99	$\gamma^* q \to q$

Processes in Pythia (ISUB)

No.	Subprocess
Photon-induced:	
33	$f_i\gamma \to f_i g$
34	$f_i\gamma \to f_i\gamma$
54	$g\gamma \to f_k\bar{f}_k$
58	$\gamma\gamma \to f_k\bar{f}_k$
131	$f_i\gamma^*_T \to f_i g$
132	$f_i\gamma^*_L \to f_i g$
133	$f_i\gamma^*_T \to f_i\gamma$
134	$f_i\gamma^*_L \to f_i\gamma$
135	$g\gamma^*_T \to f_i\bar{f}_i$
136	$g\gamma^*_L \to f_i\bar{f}_i$
137	$\gamma^*_T\gamma^*_T \to f_i\bar{f}_i$
138	$\gamma^*_T\gamma^*_L \to f_i\bar{f}_i$
139	$\gamma^*_L\gamma^*_T \to f_i\bar{f}_i$
140	$\gamma^*_L\gamma^*_L \to f_i\bar{f}_i$
80	$q_i\gamma \to q_k\pi^\pm$
Light SM Higgs:	
3	$f_i\bar{f}_i \to h^0$
24	$f_i\bar{f}_i \to Z^0h^0$
26	$f_i\bar{f}_j \to W^\pm h^0$

No.	Subprocess
32	$f_i g \to f_i h^0$
102	$gg \to h^0$
103	$\gamma\gamma \to h^0$
110	$f_i\bar{f}_i \to \gamma h^0$
111	$f_i\bar{f}_i \to g h^0$
112	$f_i g \to f_i h^0$
113	$gg \to g h^0$
121	$gg \to Q_k\overline{Q}_k h^0$
122	$q_i\bar{q}_i \to Q_k\overline{Q}_k h^0$
123	$f_i f_j \to f_i f_j h^0$
124	$f_i f_j \to f_k f_l h^0$
Heavy SM Higgs:	
5	$Z^0Z^0 \to h^0$
8	$W^+W^- \to h^0$
71	$Z^0_L Z^0_L \to Z^0_L Z^0_L$
72	$Z^0_L Z^0_L \to W^+_L W^-_L$
73	$Z^0_L W^\pm_L \to Z^0_L W^\pm_L$
76	$W^+_L W^-_L \to Z^0_L Z^0_L$
77	$W^\pm_L W^\pm_L \to W^\pm_L W^\pm_L$

No.	Subprocess
BSM Neutral Higgs:	
151	$f_i\bar{f}_i \to H^0$
152	$gg \to H^0$
153	$\gamma\gamma \to H^0$
171	$f_i\bar{f}_i \to Z^0H^0$
172	$f_i\bar{f}_j \to W^\pm H^0$
173	$f_i f_j \to f_i f_j H^0$
174	$f_i f_j \to f_k f_l H^0$
181	$gg \to Q_k\overline{Q}_k H^0$
182	$q_i\bar{q}_i \to Q_k\overline{Q}_k H^0$
183	$f_i\bar{f}_i \to g H^0$
184	$f_i g \to f_i H^0$
185	$gg \to g H^0$
156	$f_i\bar{f}_i \to A^0$
157	$gg \to A^0$
158	$\gamma\gamma \to A^0$
176	$f_i\bar{f}_i \to Z^0A^0$
177	$f_i\bar{f}_j \to W^\pm A^0$
178	$f_i f_j \to f_i f_j A^0$
179	$f_i f_j \to f_k f_l A^0$

Processes in Pythia (ISUB)

No.	Subprocess
186	$gg \to Q_k\overline{Q}_k A^0$
187	$q_i\overline{q}_i \to Q_k\overline{Q}_k A^0$
188	$f_i\overline{f}_i \to gA^0$
189	$f_i g \to f_i A^0$
190	$gg \to gA^0$
Charged Higgs:	
143	$f_i\overline{f}_j \to H^+$
161	$f_i g \to f_k H^+$
Higgs pairs:	
297	$f_i\overline{f}_j \to H^\pm h^0$
298	$f_i\overline{f}_j \to H^\pm H^0$
299	$f_i\overline{f}_i \to A^0 h^0$
300	$f_i\overline{f}_i \to A^0 H^0$
301	$f_i\overline{f}_i \to H^+ H^-$
New gauge bosons:	
141	$f_i\overline{f}_i \to \gamma/Z^0/Z'^0$
142	$f_i\overline{f}_j \to W'^+$
144	$f_i\overline{f}_j \to R$

No.	Subprocess
Technicolor:	
149	$gg \to \eta_{tc}$
191	$f_i\overline{f}_i \to \rho^0_{tc}$
192	$f_i\overline{f}_j \to \rho^+_{tc}$
193	$f_i\overline{f}_i \to \omega^0_{tc}$
194	$f_i\overline{f}_i \to f_k\overline{f}_k$
195	$f_i\overline{f}_j \to f_k\overline{f}_l$
361	$f_i\overline{f}_i \to W^+_L W^-_L$
362	$f_i\overline{f}_i \to W^\pm_L \pi^\mp_{tc}$
363	$f_i\overline{f}_i \to \pi^+_{tc}\pi^-_{tc}$
364	$f_i\overline{f}_i \to \gamma\pi^0_{tc}$
365	$f_i\overline{f}_i \to \gamma\pi'^0_{tc}$
366	$f_i\overline{f}_i \to Z^0\pi^0_{tc}$
367	$f_i\overline{f}_i \to Z^0\pi'^0_{tc}$
368	$f_i\overline{f}_i \to W^\pm\pi^\mp_{tc}$
370	$f_i\overline{f}_j \to W^\pm_L Z^0_L$
371	$f_i\overline{f}_j \to W^\pm_L \pi^0_{tc}$
372	$f_i\overline{f}_j \to \pi^\pm_{tc} Z^0_L$
373	$f_i\overline{f}_j \to \pi^\pm_{tc}\pi^0_{tc}$
374	$f_i\overline{f}_j \to \gamma\pi^\pm_{tc}$

No.	Subprocess
375	$f_i\overline{f}_j \to Z^0\pi^\pm_{tc}$
376	$f_i\overline{f}_j \to W^\pm\pi^0_{tc}$
377	$f_i\overline{f}_j \to W^\pm\pi'^0_{tc}$
381	$q_i q_j \to q_i q_j$
382	$q_i\overline{q}_i \to q_k\overline{q}_k$
383	$q_i\overline{q}_i \to gg$
384	$f_i g \to f_i g$
385	$gg \to q_k\overline{q}_k$
386	$gg \to gg$
387	$f_i\overline{f}_i \to Q_k\overline{Q}_k$
388	$gg \to Q_k\overline{Q}_k$
Compositeness:	
146	$e\gamma \to e^*$
147	$dg \to d^*$
148	$ug \to u^*$
167	$q_i q_j \to d^* q_k$
168	$q_i q_j \to u^* q_k$
169	$q_i\overline{q}_i \to e^\pm e^{*\mp}$
165	$f_i\overline{f}_i(\to \gamma^*/Z^0) \to f_k\overline{f}_k$
166	$f_i\overline{f}_j(\to W^\pm) \to f_k\overline{f}_l$

Processes in Pythia (ISUB)

No.	Subprocess
Leptoquarks:	
145	$q_i\ell_j \to L_Q$
162	$qg \to \ell L_Q$
163	$gg \to L_Q\overline{L}_Q$
164	$q_i\overline{q}_i \to L_Q\overline{L}_Q$
Left–right symmetry:	
341	$\ell_i\ell_j \to H_L^{\pm\pm}$
342	$\ell_i\ell_j \to H_R^{\pm\pm}$
343	$\ell_i^\pm\gamma \to H_L^{\pm\pm}e^\mp$
344	$\ell_i^\pm\gamma \to H_R^{\pm\pm}e^\mp$
345	$\ell_i^\pm\gamma \to H_L^{\pm\pm}\mu^\mp$
346	$\ell_i^\pm\gamma \to H_R^{\pm\pm}\mu^\mp$
347	$\ell_i^\pm\gamma \to H_L^{\pm\pm}\tau^\mp$
348	$\ell_i^\pm\gamma \to H_R^{\pm\pm}\tau^\mp$
349	$f_i\overline{f}_i \to H_L^{++}H_L^{--}$
350	$f_i\overline{f}_i \to H_R^{++}H_R^{--}$
351	$f_if_j \to f_kf_lH_L^{\pm\pm}$
352	$f_if_j \to f_kf_lH_R^{\pm\pm}$
353	$f_i\overline{f}_i \to Z_R^0$
354	$f_i\overline{f}_j \to W_R^\pm$

No.	Subprocess
Extra Dimensions:	
391	$f\overline{f} \to G^*$
392	$gg \to G^*$
393	$q\overline{q} \to gG^*$
394	$qg \to qG^*$
395	$gg \to gG^*$
SUSY:	
201	$f_i\overline{f}_i \to \tilde{e}_L\tilde{e}_L^*$
202	$f_i\overline{f}_i \to \tilde{e}_R\tilde{e}_R^*$
203	$f_i\overline{f}_i \to \tilde{e}_L\tilde{e}_R^*+$
204	$f_i\overline{f}_i \to \tilde{\mu}_L\tilde{\mu}_L^*$
205	$f_i\overline{f}_i \to \tilde{\mu}_R\tilde{\mu}_R^*$
206	$f_i\overline{f}_i \to \tilde{\mu}_L\tilde{\mu}_R^*+$
207	$f_i\overline{f}_i \to \tilde{\tau}_1\tilde{\tau}_1^*$
208	$f_i\overline{f}_i \to \tilde{\tau}_2\tilde{\tau}_2^*$
209	$f_i\overline{f}_i \to \tilde{\tau}_1\tilde{\tau}_2^*+$
210	$f_i\overline{f}_j \to \tilde{\ell}_L\tilde{\nu}_\ell^*+$
211	$f_i\overline{f}_j \to \tilde{\tau}_1\tilde{\nu}_\tau^*+$
212	$f_i\overline{f}_j \to \tilde{\tau}_2\tilde{\nu}_\tau^*+$
213	$f_i\overline{f}_i \to \tilde{\nu}_\ell\tilde{\nu}_\ell^*$

No.	Subprocess
214	$f_i\overline{f}_i \to \tilde{\nu}_\tau\tilde{\nu}_\tau^*$
216	$f_i\overline{f}_i \to \tilde{\chi}_1\tilde{\chi}_1$
217	$f_i\overline{f}_i \to \tilde{\chi}_2\tilde{\chi}_2$
218	$f_i\overline{f}_i \to \tilde{\chi}_3\tilde{\chi}_3$
219	$f_i\overline{f}_i \to \tilde{\chi}_4\tilde{\chi}_4$
220	$f_i\overline{f}_i \to \tilde{\chi}_1\tilde{\chi}_2$
221	$f_i\overline{f}_i \to \tilde{\chi}_1\tilde{\chi}_3$
222	$f_i\overline{f}_i \to \tilde{\chi}_1\tilde{\chi}_4$
223	$f_i\overline{f}_i \to \tilde{\chi}_2\tilde{\chi}_3$
224	$f_i\overline{f}_i \to \tilde{\chi}_2\tilde{\chi}_4$
225	$f_i\overline{f}_i \to \tilde{\chi}_3\tilde{\chi}_4$
226	$f_i\overline{f}_i \to \tilde{\chi}_1^\pm\tilde{\chi}_1^\mp$
227	$f_i\overline{f}_i \to \tilde{\chi}_2^\pm\tilde{\chi}_2^\mp$
228	$f_i\overline{f}_i \to \tilde{\chi}_1^\pm\tilde{\chi}_2^\mp$
229	$f_i\overline{f}_j \to \tilde{\chi}_1\tilde{\chi}_1^\pm$
230	$f_i\overline{f}_j \to \tilde{\chi}_2\tilde{\chi}_1^\pm$
231	$f_i\overline{f}_j \to \tilde{\chi}_3\tilde{\chi}_1^\pm$
232	$f_i\overline{f}_j \to \tilde{\chi}_4\tilde{\chi}_1^\pm$
233	$f_i\overline{f}_j \to \tilde{\chi}_1\tilde{\chi}_2^\pm$
234	$f_i\overline{f}_j \to \tilde{\chi}_2\tilde{\chi}_2^\pm$

Processes in Pythia (ISUB)

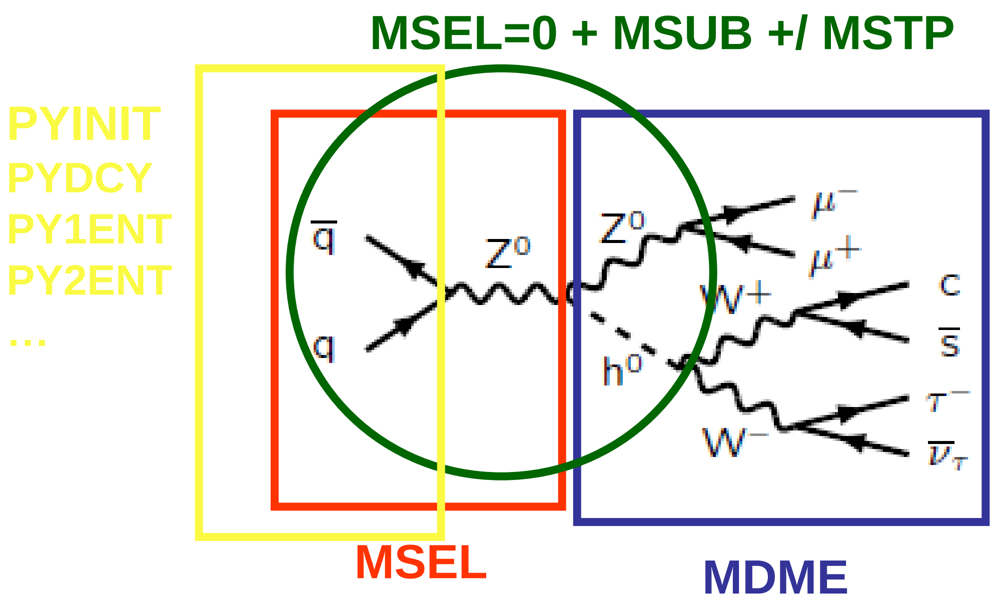
No.	Subprocess
235	$f_i\bar{f}_j \to \tilde{\chi}_3\tilde{\chi}_2^\pm$
236	$f_i\bar{f}_j \to \tilde{\chi}_4\tilde{\chi}_2^\pm$
237	$f_i\bar{f}_i \to \tilde{g}\tilde{\chi}_1$
238	$f_i\bar{f}_i \to \tilde{g}\tilde{\chi}_2$
239	$f_i\bar{f}_i \to \tilde{g}\tilde{\chi}_3$
240	$f_i\bar{f}_i \to \tilde{g}\tilde{\chi}_4$
241	$f_i\bar{f}_j \to \tilde{g}\tilde{\chi}_1^\pm$
242	$f_i\bar{f}_j \to \tilde{g}\tilde{\chi}_2^\pm$
243	$f_i\bar{f}_i \to \tilde{g}\tilde{g}$
244	$gg \to \tilde{g}\tilde{g}$
246	$f_i g \to \tilde{q}_{iL}\tilde{\chi}_1$
247	$f_i g \to \tilde{q}_{iR}\tilde{\chi}_1$
248	$f_i g \to \tilde{q}_{iL}\tilde{\chi}_2$
249	$f_i g \to \tilde{q}_{iR}\tilde{\chi}_2$
250	$f_i g \to \tilde{q}_{iL}\tilde{\chi}_3$
251	$f_i g \to \tilde{q}_{iR}\tilde{\chi}_3$
252	$f_i g \to \tilde{q}_{iL}\tilde{\chi}_4$
253	$f_i g \to \tilde{q}_{iR}\tilde{\chi}_4$

No.	Subprocess
254	$f_i g \to \tilde{q}_{jL}\tilde{\chi}_1^\pm$
256	$f_i g \to \tilde{q}_{jL}\tilde{\chi}_2^\pm$
258	$f_i g \to \tilde{q}_{iL}\tilde{g}$
259	$f_i g \to \tilde{q}_{iR}\tilde{g}$
261	$f_i\bar{f}_i \to \tilde{t}_1\tilde{t}_1^*$
262	$f_i\bar{f}_i \to \tilde{t}_2\tilde{t}_2^*$
263	$f_i\bar{f}_i \to \tilde{t}_1\tilde{t}_2^* +$
264	$gg \to \tilde{t}_1\tilde{t}_1^*$
265	$gg \to \tilde{t}_2\tilde{t}_2^*$
271	$f_i f_j \to \tilde{q}_{iL}\tilde{q}_{jL}$
272	$f_i f_j \to \tilde{q}_{iR}\tilde{q}_{jR}$
273	$f_i f_j \to \tilde{q}_{iL}\tilde{q}_{jR} +$
274	$f_i\bar{f}_j \to \tilde{q}_{iL}\tilde{q}^*_{jL}$
275	$f_i\bar{f}_j \to \tilde{q}_{iR}\tilde{q}^*_{jR}$
276	$f_i\bar{f}_j \to \tilde{q}_{iL}\tilde{q}^*_{jR} +$
277	$f_i\bar{f}_i \to \tilde{q}_{jL}\tilde{q}^*_{jL}$
278	$f_i\bar{f}_i \to \tilde{q}_{jR}\tilde{q}^*_{jR}$
279	$gg \to \tilde{q}_{iL}\tilde{q}^*_{iL}$

No.	Subprocess
280	$gg \to \tilde{q}_{iR}\tilde{q}^*_{iR}$
281	$bq_i \to \tilde{b}_1\tilde{q}_{iL}$
282	$bq_i \to \tilde{b}_2\tilde{q}_{iR}$
283	$bq_i \to \tilde{b}_1\tilde{q}_{iR} + \tilde{b}_2\tilde{q}_{iL}$
284	$b\bar{q}_i \to \tilde{b}_1\tilde{q}^*_{iL}$
285	$b\bar{q}_i \to \tilde{b}_2\tilde{q}^*_{iR}$
286	$b\bar{q}_i \to \tilde{b}_1\tilde{q}^*_{iR} + \tilde{b}_2\tilde{q}^*_{iL}$
287	$f_i\bar{f}_i \to \tilde{b}_1\tilde{b}_1^*$
288	$f_i\bar{f}_i \to \tilde{b}_2\tilde{b}_2^*$
289	$gg \to \tilde{b}_1\tilde{b}_1^*$
290	$gg \to \tilde{b}_2\tilde{b}_2^*$
291	$bb \to \tilde{b}_1\tilde{b}_1$
292	$bb \to \tilde{b}_2\tilde{b}_2$
293	$bb \to \tilde{b}_1\tilde{b}_2$
294	$bg \to \tilde{b}_1\tilde{g}$
295	$bg \to \tilde{b}_2\tilde{g}$
296	$b\bar{b} \to \tilde{b}_1\tilde{b}_2^* +$

Process setup

PYINIT + MSEL +/ MSUB +/ MSTP +/ MDME +/ …



MSEL lets (=0) to make choosing process by "hands" (MSUB +/ MSTP)

Some other parameters

MSTP – see manual

MSTP(1)=4 – 4-th generation install

MSEL=4 – switching on of 4-th gen.

MSUB(81)=1

MSUB(82)=1 for quarks of 4-th gen.

MSTP(51) – PDF (D=7 – CTEQ5L)

PMAS(KC,1) – mass of КC-particle

MDCY(KC,1)=0,1 – switching off/on decay (for modes (IDC) switched on with MDME at branching BRAT(IDC)), **excluding resonances**

PMAS(KC,2)/ PMAS(KC,2) – width/lifetime (GeV/mm) – does not work for resonances

MWID(KC)=0,1 – switching off/on resonance decay

CKIN(1/2) – min/max sqrt(s)

CKIN(3/4) – min/max pT (for 2→2)

CKIN(13/14) – min/max η(D=+-40)

Common parameters of generated event

COMMON/PYJETS/N,NPAD,K(4000,5),P(4000,5),V(4000,5)

N – the number of products in one event

K(1-4000,1-5)

K(I,1) – status of i-th (intermediate) product (see manual)

K(I,2) – KF-code of i-th product (at i-th line)

K(I,3) – number of line of the parent (if any)

K(I,4/5) – number of 1st/last daughter (if any)

P(1-4000,1-5)

P(I,1-5) – px,py,pz,E,m of i-th product

V(1-4000,1-5)

V(I,1-5) – x,y,z,t,lifetime of i-th product

Some functions of Pythia

CALL PYEVNT – generation of the given event => all the data arrays are filled out: N, K(I,1-5), P(I,1-5), …

PYEDIT(1) – leave only stable (long living) products – i.e. to decay everything what decays quickly

PYLIST(2) – printing out of event

PYSTAT (1) – printing out of cross section

PYHIST – drawing of histogram/ CALL PYFILL(1,P(I,4),1D0) – filling/ PYBOOK(1,"Spectrum",100,0D0,100D0) – booking

CALL PYEXEC – fulfill the chosen options

Output of Pythia (1)

http://home.fnal.gov/~mrenna/041207_pythia_tutorial/03_041207_pythia_tutorial_PS_2_sample.out.txt

Event listing (standard)

I	particle/jet		K(I,1)	K(I,2)	K(I,3)	K(I,4)	K(I,5)	P(I,1)	P(I,2)	P(I,3)	P(I,4)	P(I,5)
1	!p+!		21	2212	0	0	0	0.00000	0.00000	3499.99987	3500.00000	0.93827
2	!p+!		21	2212	0	0	0	0.00000	0.00000	-3499.99987	3500.00000	0.93827
3	!g!		21	21	1	0	0	-1.52733	1.37843	184.46649	184.47796	0.00000
4	!u!		21	2	2	0	0	0.38909	-0.48905	-224.85248	224.85335	0.00000
5	!ubar!		21	-2	3	0	0	-0.84004	-2.69293	-100.17106	100.21077	0.00000
6	!u!		21	2	4	0	0	0.20126	-0.25296	-116.30618	116.30663	0.00000
7	!Z0!		21	23	0	0	0	-0.63878	-2.94589	-216.47723	216.51739	2.88154
8	!mu-!		21	13	7	0	0	-0.55408	-2.42260	-79.10463	79.14373	0.10566
9	!mu+!		21	-13	7	0	0	-0.08470	-0.52330	-137.37260	137.37366	0.10566
10	(Z0)		11	23	7	11	12	-0.63878	-2.94589	-216.47723	216.51739	2.88154
11	mu-		1	13	8	0	0	-0.55408	-2.42260	-79.10463	79.14373	0.10566
12	mu+		1	-13	9	0	0	-0.08470	-0.52330	-137.37260	137.37366	0.10566
13	(u)	A	12	2	3	28	28	0.76118	2.14258	-4.98506	5.48906	0.33000
14	(g)	I	12	21	3	28	28	-1.97802	2.50209	183.83048	183.85815	0.00000
15	(uu_1)	V	11	2203	1	28	28	1.02216	-0.90376	2672.07524	2672.0757	0.77133

…………………………

Структура корневого процесса

Output of Pythia (2)

MSTP(125)=0 (D=1) **see p.86 of manual**

Event listing (standard)

I	particle/jet		K(I,1)	K(I,2)	K(I,3)	K(I,4)	K(I,5)	P(I,1)	P(I,2)	P(I,3)	P(I,4)	P(I,5)
1	(Z0)		11	23	0	2	3	-0.63878	-2.94589	-216.47723	216.51739	2.88154
2	mu-		1	13	0	0	0	-0.55408	-2.42260	-79.10463	79.14373	0.10566
3	mu+		1	-13	0	0	0	-0.08470	-0.52330	-137.37260	137.37366	0.10566
4	(u)	A	12	2	0	19	19	0.76118	2.14258	-4.98506	5.48906	0.33000
5	(g)	I	12	21	0	19	19	-1.97802	2.50209	183.83048	183.85815	0.00000
6	(uu_1)	V	11	2203	0	19	19	1.02216	-0.90376	2672.07524	2672.07570	0.77133
7	(d)	A	12	1	0	38	38	0.50517	-0.47468	6.86331	6.90611	0.33000
8	(g)	I	12	21	0	38	38	0.10823	-0.62221	-2.75806	2.82944	0.00000
9	(g)	I	12	21	0	38	38	0.60915	-0.18717	0.00388	0.63727	0.00000
10	(g)	I	12	21	0	38	38	-9.03034	-3.15734	-24.98398	26.75286	0.00000
11	(g)	I	12	21	0	38	38	-0.22724	0.10068	1.80241	1.81946	0.00000
12	(g)	I	12	21	0	38	38	-4.29475	2.36977	59.45086	59.65287	0.00000
13	(g)	I	12	21	0	38	38	-3.10020	-1.79857	526.70190	526.71409	0.00000
14	(g)	I	12	21	0	38	38	0.22724	-0.10068	15.51683	15.51882	0.00000
15	(g)	I	12	21	0	38	38	3.10020	1.79857	27.66076	27.89200	0.00000
16	(g)	I	12	21	0	38	38	4.29475	-2.36977	2.90457	5.70064	0.00000
17	(g)	I	12	21	0	38	38	9.03034	3.15734	-1525.02585	1525.05586	0.00000
18	(ud_0)	V	11	2101	0	38	38	-0.38909	0.48905	-1722.58005	1722.58027	0.57933

==

19	(string)	11	92	4	20	37	-0.19468	3.74091	2850.92066	2861.42291	244.90488
20	(rho0)	11	113	19	96	97	0.58193	1.70730	-3.37863	3.92555	0.86099
21	p+	1	2212	19	0	0	-0.02072	-0.13075	0.04995	0.94888	0.93827
22	nbar0	1	-2112	19	0	0	-0.17745	0.63069	0.16922	1.15788	0.93957

Common blocks in fortran 77

COMMON-blocks are used instead of global variables:

In the main programme:

```
common /coeff/ alpha, beta
```

In any subroutine, used the same variables**:**

```
common /coeff/ alpha, beta
```

The most full declaration of the Pythia common-blocks is in

http://projects.hepforge.org/pythia6/examples/main60.f

Tasks

1а) $pp \rightarrow \gamma \rightarrow \mu\mu$	σ, dN_{μ}/dE
1б) $pp \rightarrow \gamma \rightarrow \mu\mu(\mu \rightarrow e\nu\nu)$	σ, dN_{e}/dE
1в) $pp \rightarrow \gamma \rightarrow \mu\mu$	σ, $dN_{\mu(\gamma)}/dE$
2а,б,в) $pp \rightarrow \gamma + Z \rightarrow \mu\mu$	
3а,б,в) $pp \rightarrow W \rightarrow \mu\nu$	
4) $pp \rightarrow W \rightarrow e\nu$	σ, $dN_{e(W)}/dE$ vs $dN_{e(tot)}/dE$
5) $pp \rightarrow tt \rightarrow \dots$	σ(PDF), dN_{μ}/dE
6) $pp \rightarrow f_4 f_4$	σ, dN_{f4}/dE
7) $ee \rightarrow \dots$	

Example

pp→Z→μμ (1)

```
              PROGRAM Z-mumu
C*****************************************************************

C...All real arithmetic in double precision.
    IMPLICIT DOUBLE PRECISION(A-H, O-Z)
C...Three Pythia functions return integers, so need declaring.
    INTEGER PYK,PYCHGE,PYCOMP

              DIMENSION Dmu(0:3500)

C...EXTERNAL statement links PYDATA on most machines.
    EXTERNAL PYDATA

C...Commonblocks.
C...The event record.
    COMMON/PYJETS/N,NPAD,K(4000,5),P(4000,5),V(4000,5)
C...Parameters.
C...Particle properties + some flavour parameters.
    COMMON/PYDAT2/KCHG(500,4),PMAS(500,4),PARF(2000),VCKM(4,4)
C...Decay information.
C...Note that dimensions below grew from 4000 to 8000 in Pythia 6.2!
    COMMON/PYDAT3/MDCY(500,3),MDME(8000,2),BRAT(8000),KFDP(8000,5)
C...Selection of hard scattering subprocesses.
    COMMON/PYSUBS/MSEL,MSELPD,MSUB(500),KFIN(2,-40:40),CKIN(200)
C...Parameters.
    COMMON/PYPARS/MSTP(200),PARP(200),MSTI(200),PARI(200)

C*******************************************************
C...Number of events, CM energy.
              NEV=1000
              ECM=7000.D0
C*******************************************************
```

Example

pp→Z→μμ (2)

```
C particle:    e    mu     gamma   Z0      t
C     KF:     11   13       22      23      6
C     KC:     11   13       22      23      6

C  process:   ff->Z/gamma   qq->QQ   gg->QQ   ff->Zg   ff->Wg
C    ISUB:        1              81        82         15       19
C
C   decays:  A->N4N4  A->E4E4  A->U4U4  A->D4D4     A->tt
C IDC A=gamma:  -      173        169       168          167
C  IDC A=Z:      189      188        181       180          179

C...Selection of the processes: pp->Z/gamma->...
      MSEL=0
      MSUB(1)=1   ! =0 - process is switched off, =1 - switched on
      MSTP(43)=3  ! D=3, 1- only gamma, 2- only Z, 3- Z+gamma

C...Switches of needed Z-boson mediated channels
      DO 10 ISUB=174,189
10    MDME(ISUB,1)=0
C....for Z->mumu
      MDME(184,1)=1
C....makes muon decay
c     MDCY(PYCOMP(13),1)=1

C...Setting sqrt(s) range acceptable for processes of interest
c     CKIN(1)=2.*PMAS(13,1)+0d0

      CALL PYINIT('CMS','p','p',ECM)
c     CALL PYINIT('CMS','e-','e+',ECM)

C...Histograms.
      CALL PYBOOK(13,'mu-spectrum',100,0D0,1000D0)

      mu=0   ! <10^31-1
      Emu=0.0
      DATA Dmu/3501*0d0/
```

Example

pp→Z→μμ (3)

```
C...Event generation loop.
          DO 200 IEV=1,NEV
          if(mod(iev,100).eq.0) write(*,*) 'begin event no', iev
c         if(IEV/5000.eq.IEV/5000.) write(*,*)'number of event = ',IEV
          CALL PYEVNT
c         CALL PYEDIT(1)

C...List first event
          IF(IEV.LE.1) CALL PYLIST(2)

C...Counting of particles
          DO 200 I=1,N
           En=P(I,4)
           IF(K(I,2).EQ.13) then
            mu=mu+1
            Emu=Emu+En
            iE=int(En)
            Dmu(iE)=Dmu(iE)+1.D0/NEV
            CALL PYFILL(13,P(I,4),1.D0)
           ENDIF
C...End of event generation loop.

200       CONTINUE

C...Cross section - not relevant in this case. Histograms.
          CALL PYHIST
          CALL PYSTAT(1)
          write(*,*)'multiplicities of mu',real(mu)/NEV

          open(1,file='mu-spectrum.txt',status='new')
          write(1,4)(Dmu(iE),iE=0,1000)
4         format(5(2x,e9.3))
c5        format(122(5(2x,e9.3),/),/,/)

          END
```

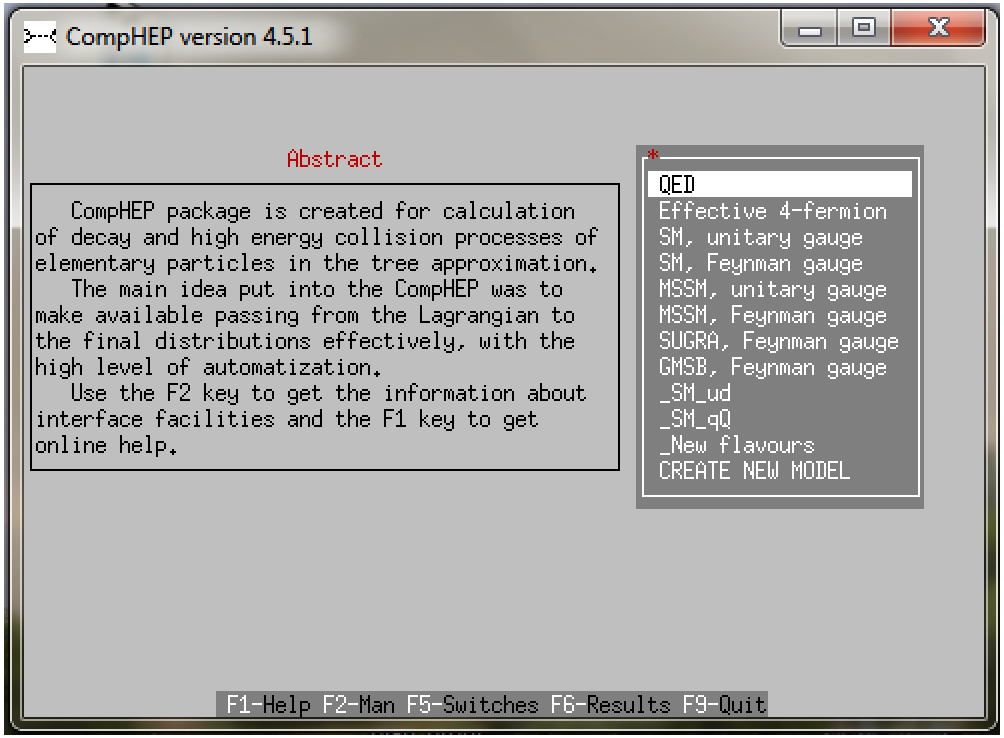
CompHEP

http://comphep.sinp.msu.ru/

Brief description

Package is aimed for calculation of cross sections and generation of hard (basic) processes from matrix element (lagrangian) in the lowest order of perturbation theory. No higher orders, loops, hadronization are possible.

All control is realized through interface (interactively).



Although, distant control is foreseen, but much less suitable.

Installation

After registration at the webcite (it is not required for CalcHep), one dowloads the last version of archive **CompHEP**.

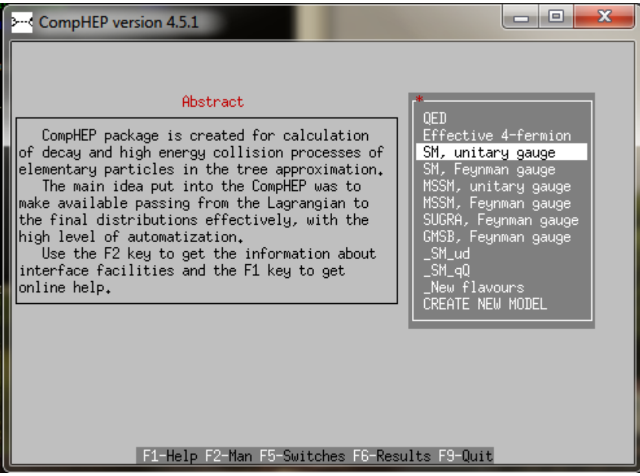
Additionally, one needs to download and launch **Xming** (or similar) to work at remote Linux-server under Windows. In **putty,** tick on **X11**. Under Linux, to enter at server E307 use **ssh –x**.

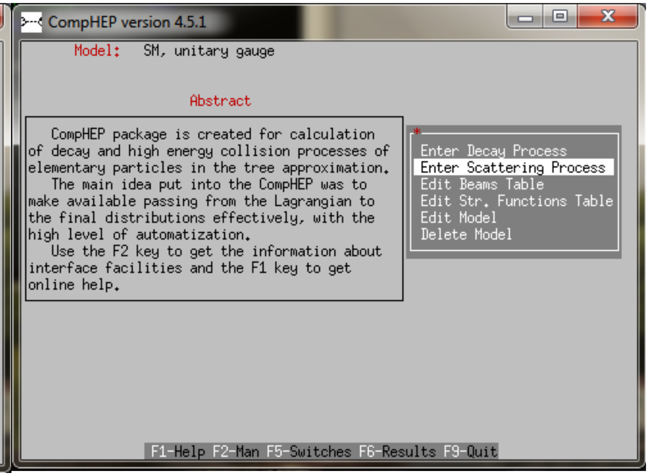
Further, follow instructions at the webcite.

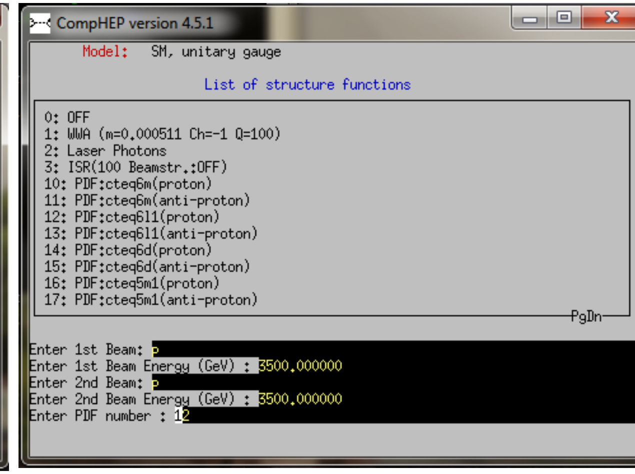
mkdir comphep
cd comphep
tar xzvf comphep-XXXX.tgz
cd comphep-XXXXX
./configure **или** **./configure ---with-gcc4**
make
make setup WDIR=$HOME/comphep/run-XXXX - Creation of working directory
cd run-XXXX
./comphep - before it, launch **Xming** and **rm LOCK**

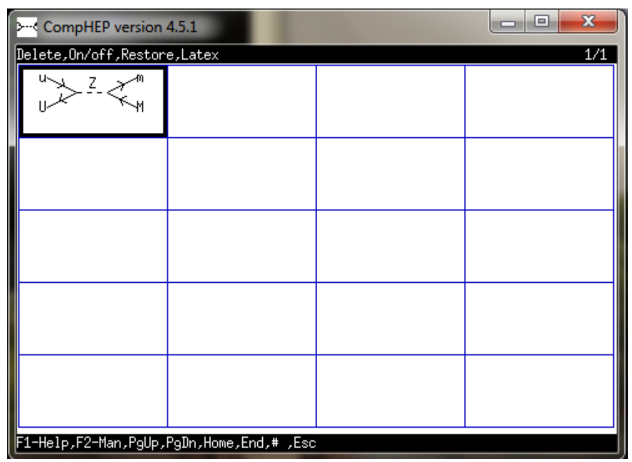
Example 1

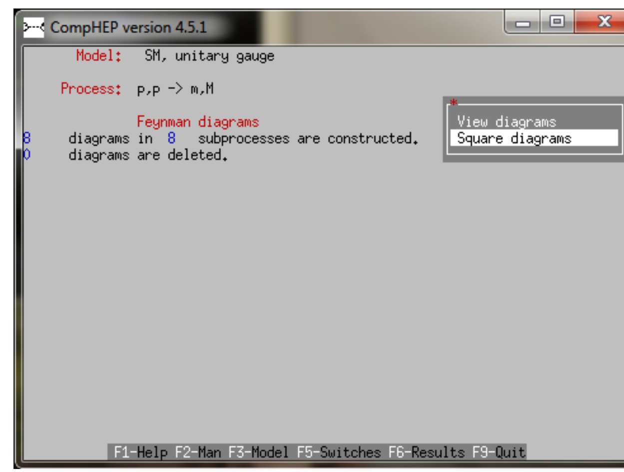
pp→Z→μμ







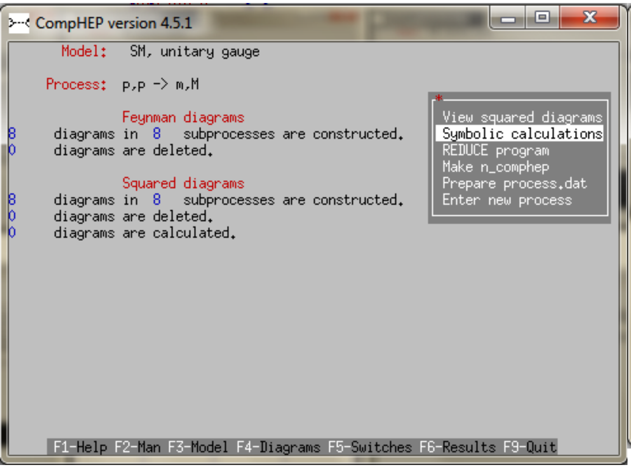


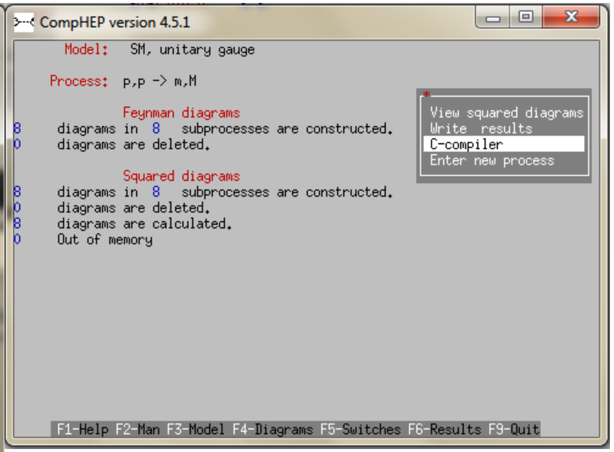


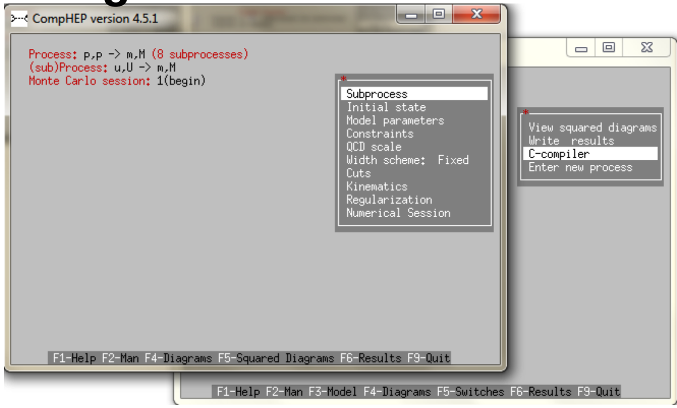
Example 1

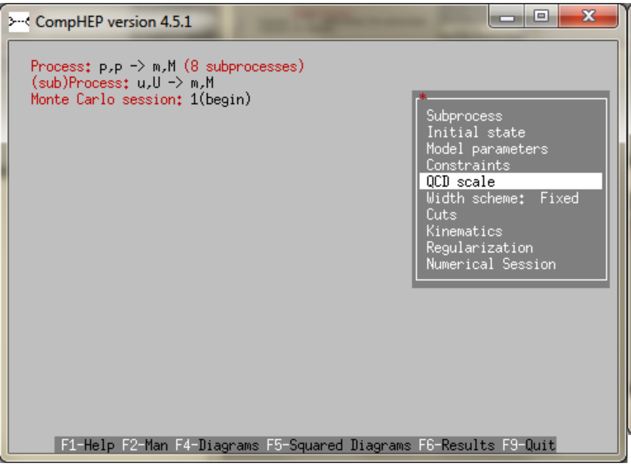
pp→Z→μμ

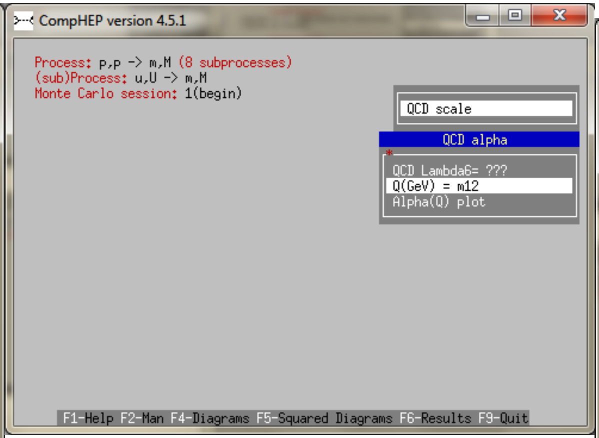
Momentum assignment: 12→34

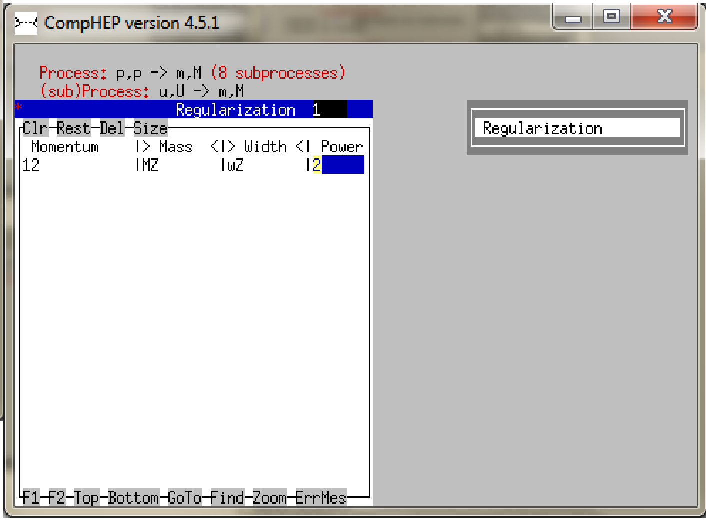






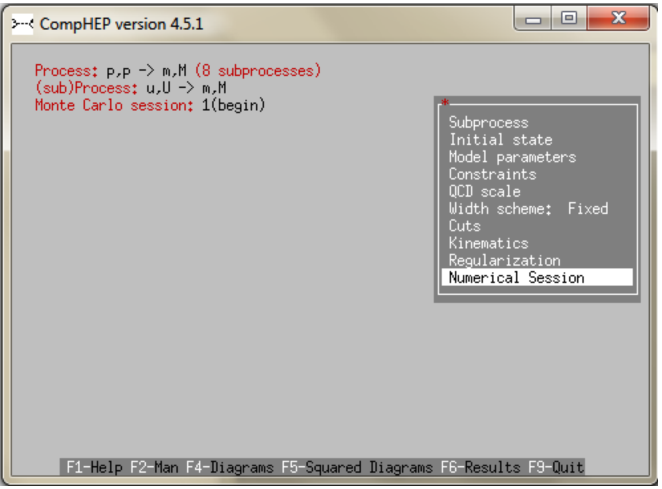


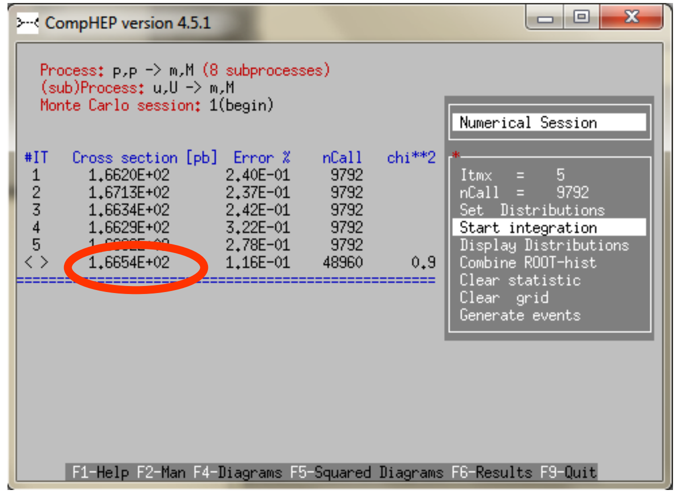




Example 1

pp→Z→μμ





Further, one needs to repeat for the processes with other quarks (subprocess) and add the result

Some other options:

Initial states – one can change init.energy, PDF

Model parameters and Constraints – the used parameters (masses,..)

Kinematics – scheme of momentum (particles) transitions (accessible, if ambiguous)

Set/Display distribution – assignment (before integration)/drawing (after) of distribution (over limited set of variables)

Generate events – event generation (4-momentums of products) →F6

Write results – to write results for $|M|^2$ in one of the formats suggested

Example 1

pp→Z→μμ

Output of CompHEP

(“new format”)

CompHEP version 4.5.1

```
##beam: ID=1, energy=3.50000E+03, KF=2212, name='proton', mass=9.38000E-01,;
##beam: ID=2, energy=3.50000E+03, KF=2212, name='proton', mass=9.38000E-01,;
##strfun: IDbeam=1, name='CTEQ', version='6l1', PDFid=58, PDFgr=4,;
##strfun: IDbeam=2, name='CTEQ', version='6l1', PDFid=58, PDFgr=4,;
##process: ID=1, name='u U -> m M', CrosSec=2.41904E+02, CrosSecErr=1.68650E-0
3, Nparton=4,
           master=3, type='scattering',;
##generator: IDprocess=1, name='CompHEP', name='4.5.1',;
##n_event: IDprocess=1, N=   10000, mult=0.00000E+00, maxW=1.50512E+00, CutN=0
,;
##parton: IDprocess=1, in=1, out=0, KF=2, name='u',
          mass=0.00000E+00,;
##parton: IDprocess=1, in=2, out=0, KF=-2, name='U',
          mass=0.00000E+00,;
##parton: IDprocess=1, in=0, out=3, KF=13, name='m',
          mass=1.05660E-01,;
##parton: IDprocess=1, in=0, out=4, KF=-13, name='M',
          mass=1.05660E-01,;
##QCDinfo: IDprocess=1, NL=1, Nflavour=6, QCDLambda=1.65200E-01,;
##format: IDprocess=1, ProcNumber='i', p1.3='17.10E', p2.3='17.10E', p3.1='17.
10E',
          p3.2='17.10E', p3.3='17.10E', p4.1='17.10E', p4.2='17.10E', p4.3='17
.10E',
          Qsquared='10.3E', color_chain='string',;
##total: Nproc=1, Nevent=   10000, CrosSec=2.41904E+02, CrosSecErr=1.68650E-03
,;
;
 1: 2.8497141097E+02:-1.0083682765E+00:-1.2024559980E+01:-1.1004961890E+01: 1.
8122624651E+02: 1.2024559980E+01: 1.1004961890E+01: 1.0273679618E+02: 3.390E+0
1:(1 2):
 1: 4.9196038114E+02:-6.0232040253E-01:-9.7732280980E+00: 1.2790439482E+01: 1.
```

p_{1z} p_{2z} p_{3x} p_{3y}

p_{3z} p_{4x} p_{4y} p_{4z} q^2

Example of ROOT script to read ComHep results (1)

```
TCanvas *vC1;
TGraph *grin, *grout;

void rd2()
{
// File name
//#include <math.h>
//#include <TH2F.h>
//#include <TH1F.h>
#define Nfiles 5
#define nch 20000
#define nchMU 25000
//   Int_t Nfiles = 5;
 TString vInFile [Nfiles] = {"events_1.txt", "events_2.txt", "events_3.txt", $
    Double_t mass = 2000.0, mass2=mass*mass;
    Double_t mu=0.10566, mu2=mu*mu;

   TString vInFileMU [4] = {"/afs/cern.ch/user/b/belotsky/comphep/run/mu/events$
   "/afs/cern.ch/user/b/belotsky/comphep/run/mu/events_2.txt",
   "/afs/cern.ch/user/b/belotsky/comphep/run/mu/events_3.txt",
   "/afs/cern.ch/user/b/belotsky/comphep/run/mu/events_4.txt"}

// File info
//   TString str [30];
    TString str1,str2,str3,str4,str5,str6;
    Float_t p1,p2;
    Double_t crosec [Nfiles], crosecMU [4];
    Char_t ch="A";
    Int_t nev [Nfiles], nevMU [4];

// histogram parameters
    Double_t weight [Nfiles];
    Float_t X,Y,w;
// read file and add to fit object
    Double_t p1x [Nfiles][nch], p1y [Nfiles][nch], p1z [Nfiles][nch];
    Double_t p2x [Nfiles][nch], p2y [Nfiles][nch], p2z [Nfiles][nch];

    Double_t ps1 [Nfiles][nch], ps2 [Nfiles][nch];

    Double_t Resx [Nfiles][nch], Resy [Nfiles][nch];

    Double_t pm1x [4][nchMU], pm1y [4][nchMU], pm1z [4][nchMU];
    Double_t pm2x [4][nchMU], pm2y [4][nchMU], pm2z [4][nchMU];
    Double_t psm1 [4][nchMU], psm2 [4][nchMU];

    Double_t Resmx [4][nchMU], Resmy [4][nchMU];
```

```
// read file

  cout << endl << "Start of reading data on muon background." << endl;
  for (Int_t Nf=0; Nf<4; Nf++)
 {
    ifstream vInputMU;
 vInputMU.open(vInFileMU[Nf], ios::in);

    Int_t Npos1=1269, Npos2=1347;

    while(ch != '='){
    vInputMU.seekg(Npos1);
    ch = vInputMU.peek();
//    cout << Npos1 << " and ch = " << ch << endl;
    Npos1++;}
    vInputMU.seekg(Npos1);
    vInputMU >> crosecMU[Nf];
//    cout << "muon partial cross section = " << crosecMU[Nf] << endl;

 while(ch != ':'){
    vInputMU.seekg(Npos2);
    ch = vInputMU.peek();
    Npos2++;}

    Int_t L, Nposi=Npos1+10;
    for(Int_t i=0; i<12;){
    vInputMU.seekg(Nposi); ch=vInputMU.peek();
    if(ch == ':'){if(i == 0){L=Nposi;} i++;}
    Nposi++;}
    L=Nposi-L-1;
//    cout << "Lenght of line is " << L << endl;

 Int_t ich=0;
    while(! vInputMU.eof()){
    vInputMU.seekg(Npos2);
    vInputMU >> pm1x[Nf][ich] >> ch >> pm1y[Nf][ich] >> ch >> pm1z[Nf][ich] >> $
    vInputMU >> pm2x[Nf][ich] >> ch >> pm2y[Nf][ich] >> ch >> pm2z[Nf][ich];
    psm1[Nf][ich]=pm1x[Nf][ich]*pm1x[Nf][ich]+pm1y[Nf][ich]*pm1y[Nf][ich]+pm1z[$
    psm2[Nf][ich]=pm2x[Nf][ich]*pm2x[Nf][ich]+pm2y[Nf][ich]*pm2y[Nf][ich]+pm2z[$

    Npos2=Npos2+L;
    ich++;}
    nevMU[Nf]=ich-1;
    cout << "Number of events in " << Nf+1 << " channel is " << nevMU[Nf] << en$

    vInputMU.close();
}
```

Example of ROOT script to read ComHep results (1)

```
Double_t crosectotMU = 0.0;
   for(Int_t Nf=0; Nf<4; Nf++){
   crosectotMU += 2.*crosecMU[Nf];}
   cout << "Total muon cross section = " << crosectotMU << " pb." << endl;
 cout << endl << "Start of reading data on useful signal." << endl;
 for (Int_t Nf=0; Nf<Nfiles; Nf++)
 {
   ifstream vInput;
  vInput.open(vInFile[Nf], ios::in);

   Int_t Npos1=1269, Npos2=1347;

   while(ch != '='){
   vInput.seekg(Npos1);
   ch = vInput.peek();
//    cout << Npos1 << " and ch = " << ch << endl;
 Npos1++;}
   vInput.seekg(Npos1);
   vInput >> crosec[Nf];

   while(ch != ':'){
   vInput.seekg(Npos2);
   ch = vInput.peek();
   Npos2++;}
 Int_t L, Nposi=Npos1+10;
   for(Int_t i=0; i<12;){
   vInput.seekg(Nposi); ch=vInput.peek();
   if(ch == ':'){if(i == 0){L=Nposi;} i++;}
   Nposi++;}
   L=Nposi-L-1;
//    cout << "Lenght of line is " << L << endl;

   Int_t ich=0;
   while(! vInput.eof()){
   vInput.seekg(Npos2);
   vInput >> p1x[Nf][ich] >> ch >> p1y[Nf][ich] >> ch >> p1z[Nf][ich] >> ch;
//    cout << p1x[Nf][ich] << " " << p1y[Nf][ich] << " " << p1z[Nf][ich] << end$
   vInput >> p2x[Nf][ich] >> ch >> p2y[Nf][ich] >> ch >> p2z[Nf][ich];
//    cout << p2x[Nf][ich] << " " << p2y[Nf][ich] << " " << p2z[Nf][ich] << end$
   ps1[Nf][ich]=p1x[Nf][ich]*p1x[Nf][ich]+p1y[Nf][ich]*p1y[Nf][ich]+p1z[Nf][ic$
   ps2[Nf][ich]=p2x[Nf][ich]*p2x[Nf][ich]+p2y[Nf][ich]*p2y[Nf][ich]+p2z[Nf][ic$

   Npos2=Npos2+L;
   ich++;}
   nev[Nf]=ich-1;
   cout << "Number of events in " << Nf+1 << " channel is " << nev[Nf] << endl;
```

```
vInput.close()
 Double_t crosectot = 0.0, fact=2.0;
   for(Int_t Nf=0; Nf<Nfiles; Nf++){
   if(Nf==4){fact=1.0;}
   crosectot += fact*crosec[Nf];}
   cout << "Total useful signal cross section = " << crosectot << " pb." << endl;

// THETA1 VS THETA2 DISTRIBUTION

   tm1 = new TH1F("tm1","MUON theta_1",45,0.,180.);
   ttm2 = new TH2F("ttm2","MUON theta_1 vs theta_2",30,0.,180.,30,0.,180.);
   for (Int_t Nf=0; Nf<4; Nf++) {
    weight[Nf]=crosecMU[Nf]/nevMU[Nf];
    w = float(weight[Nf]);
    for (Int_t iev=0; iev<nevMU[Nf]; iev++){
     Resmx[Nf][iev] = acos(pm1z[Nf][iev]/sqrt(psm1[Nf][iev]))*180./3.14159265;
     Resmy[Nf][iev] = acos(pm2z[Nf][iev]/sqrt(psm2[Nf][iev]))*180./3.14159265;
     X = float(Resmx[Nf][iev]);
     Y = float(Resmy[Nf][iev]);
     ttm2->Fill(X,Y,w);
     tm1->Fill(X,w);
     ttm2->Fill(180.-X,180.-Y,w);
     tm1->Fill(180.-X,w);
    }
  }
...............

  cout << "The following histogrames are created:" << endl;
  cout << "tt2 = theta1 vs theta2\n" << "t1 = theta1\n";
  cout << "e2 = Ekin1 vs Ekin2\n" << "et2 = Ekin1 vs theta1" << endl;
  cout << "m1 = m12\n" << "tr1 = theta12\n" << "mt2 = m12 vs theta12." << endl;
  cout << "And the same for muons named with appending 'm' before 1/2" << endl;
  cout << "Now mt2 is drawing." << endl;

  mt2->Draw("cont1");

/*  multigr = new TMultiGraph();


  for (Int_t Nf=0; Nf<Nfiles; Nf++)
  {
  multigr -> Add(gr[Nf]);
  }
 multigr->Draw("ALP");
}
```

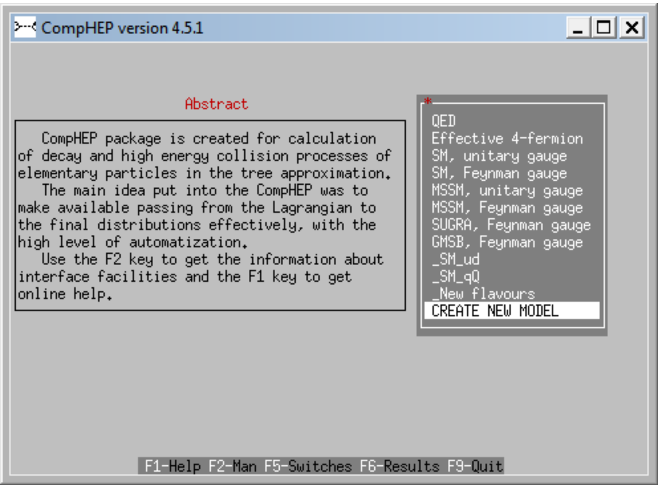
Task 1

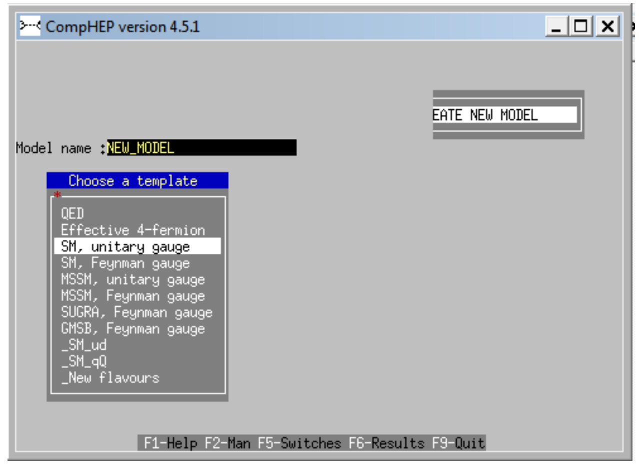
Repeat the task fulfilled in Pythia, point a):

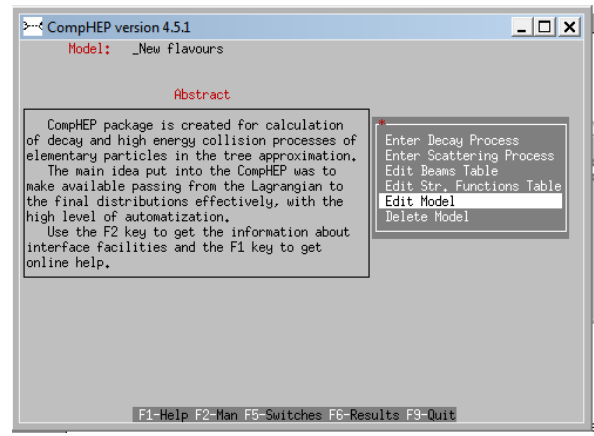
calculate cross section, write simulation data

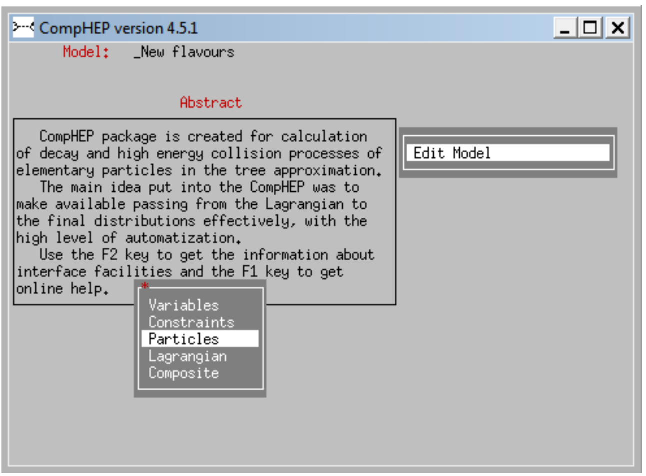
Example 2

$pp \to U_4 U_4$









Particles 21

Clr-Rest-Del-Size

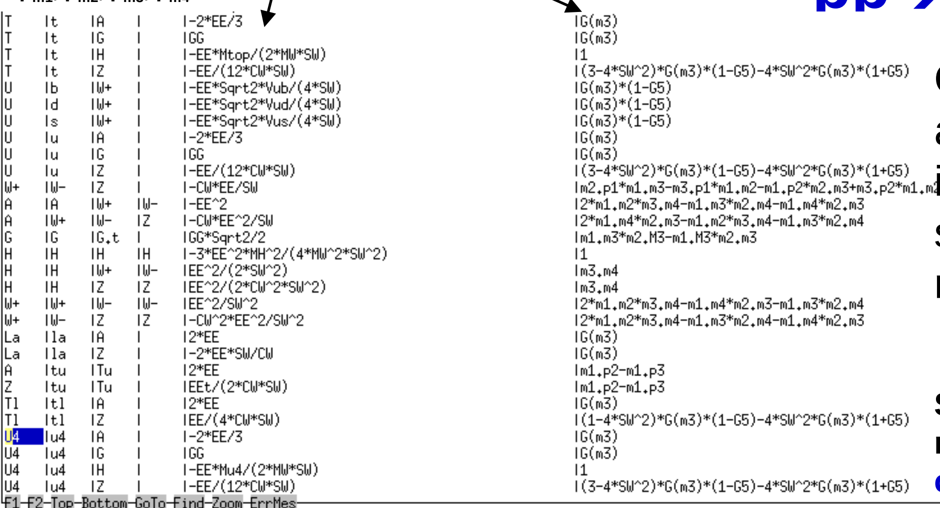
Full name	P	aP	2*spin	mass	width	color	aux	> LaTeX(A)	<|> LaTeX(A+)
gluon	G	G	2	0	0	8	G	G	G
photon	A	A	2	0	0	1	G	A	A
Z boson	Z	Z	2	MZ	wZ	1		Z	Z
W boson	W+	W-	2	MW	wW	1		W^+	W^-
neutrino	ne	Ne	1	0	0	1	L	\nu^e	\bar{\nu}^e
electron	e	E	1	0	0	1		e	\bar{e}
mu-neutrino	nm	Nm	1	0	0	1	L	\nu^\mu	\bar{\nu}^\mu
muon	m	M	1	Mm	0	1		\mu	\bar{\mu}
tau-neutrino	nl	Nl	1	0	0	1	L	\nu^\tau	\bar{\nu}^\tau
tau-lepton	l	L	1	Mtau	0	1		\tau	\bar{\tau}
u-quark	u	U	1	0	0	3		u	\bar{u}
d-quark	d	D	1	0	0	3		d	\bar{d}
c-quark	c	C	1	Mc	0	3		c	\bar{c}
s-quark	s	S	1	Ms	0	3		s	\bar{s}
t-quark	t	T	1	Mtop	wtop	3		t	\bar{t}
b-quark	b	B	1	Mb	0	3		b	\bar{b}
Higgs	H	H	0	MH	wH	1		H	H
AC-lepton	la	La	1	Mla	0	1		L_{ac}^{--}	\bar{L}_{ac}^{++}
tera-lepton	tl	Tl	1	Mtl	0	1		\zeta^{--}	\bar{zeta}^{++}
tera-baryon	tu	Tu	0	Mtu	0	1		UU^{++}	\bar{U}\bar{U}^{--}
U4-quark	u4	U4	1	Mu4	0	3		U_4^{+2/3}	\bar{U}_4^{-2/3}

Example 2

$pp \rightarrow U_4 U_4$

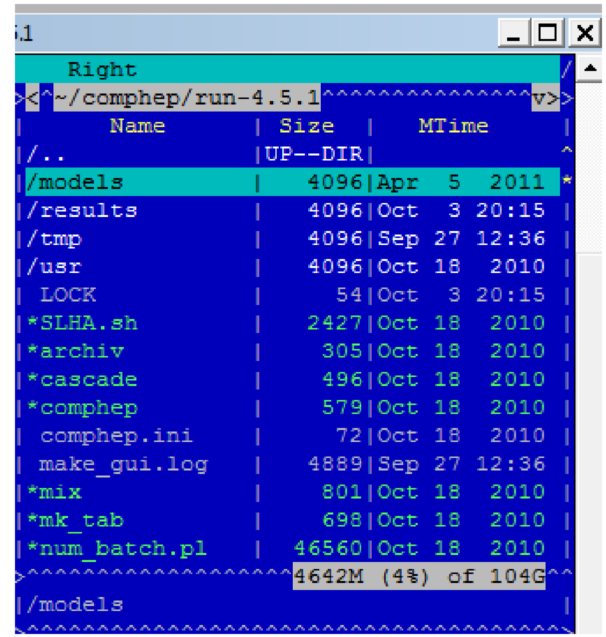
Mathematical operations are very limited

p_{m1}, p_{m2}, p_{m3}, p_{m4}

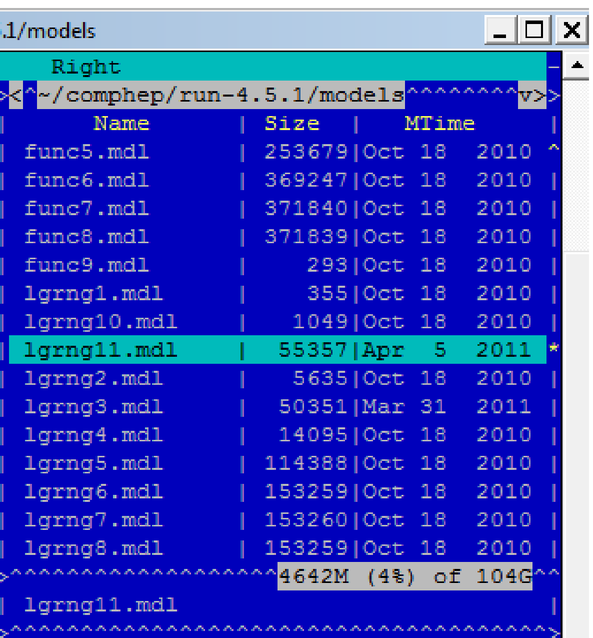


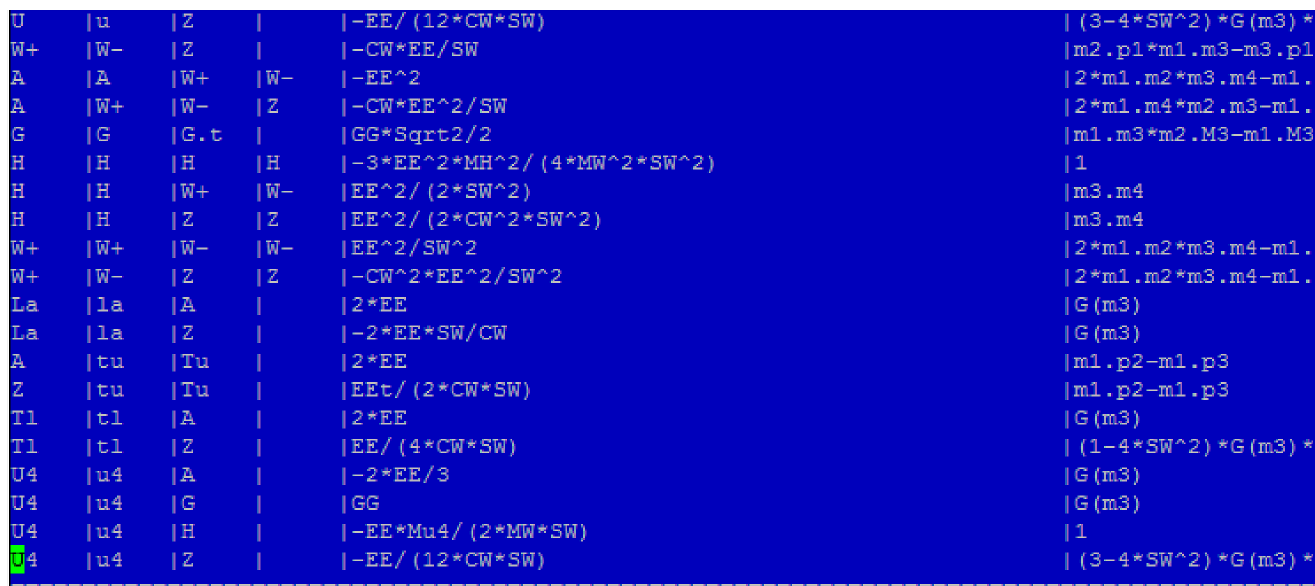
$G(m3)=\gamma_\mu(V_{\mu3})$ $m1.p3=p3_{\mu1}$

Or alternatively, in more suitable manner:



Similarly, for the rest: variables, constraints, particles





Task 2

1) pp→$L_{AC}^{++}L_{AC}^{--}$ (Q=-2, m=500 GeV, T3=0)

$$L_{Zff} \sim \bar{F}_L(\bar{g}(T_3 - Q\cdot\xi)\not{Z})F_L + \bar{f}_R(\bar{g}(-Q\cdot\xi)\not{Z})f_R$$

2) pp→$\zeta^{++}\zeta^{--}$ (Q=-2, m=500 GeV, T3=-1/2)

3) pp→$\varphi^{+}\varphi^{-}$ L= e($\varphi\partial_\mu\varphi^*-\varphi^*\partial_\mu\varphi$)$A_\mu$ (m=500 GeV)

MadGraph v.4

Tree-level event generator

Generation of the code on-line

- Register on the web-site
  http://madgraph.hep.uiuc.edu/
- For code generation use **MadGraph v4**

Code can be generated either by:

I. Fill the form:

Model:	SM	Model descriptions
Input Process:	e+e->mu+mu-	Examples/format
Max QCD Order:	99	
Max QED Order:	99	
p and j definitions:	p=j=d u s c d~ u~ s~ c~ g	
sum over leptons:	l+ = e+, mu+ ; l- = e-, mu- ; vl = ve, vm ; vl~ = ve~, vm~	

Submit

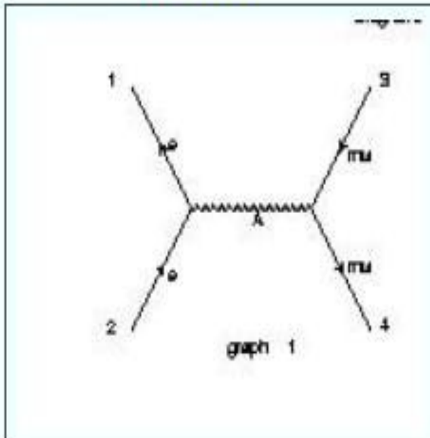
Created: Tue Nov 1 00:58:08 CDT 2011

Process: e+e->mu+mu-

QCD=99

QED=99

Model: sm



Links	Status
Process Information	Generation Complete
Code Download	Available
On-line Event Generation	Available (access restricted)
Results and Event Database	No runs available

Installation of MadGraph

- Download MadGraph V4
- Unpack the archive to the working directory (tar -zxvf MG_ME_V4.5.1.tar)
- Enter the directory MG_ME_V4.5.1, run make

Event generation and calculation of the cross-section of the process

- Download the generated code (Code Download)
- Unpack the code to the newly created directory in the folder MG_ME_V4.5.1

Dir name	Role or Content
lib	libraries
Events	event files, results, logfiles, plots.
bin	executables and scripts
Source	process independant source files
Cards	steering cards
SubProcesses	directories containing subprocesses and process dependant source files
HTML	html files to surf on results

- For event generation and calculation of the cross-section of the process e+e-→mu+mu- execute ./bin/generate_events in your directory
- If you don't need event generation you should execute the following commands one by one instead of the above mentioned command

  ./bin/survey

  ./bin/refine

In this case you get calculation of the cross-section only.

- To see the results open results.html in the SubProcesses directory

Process results

s= 43.728± 0.124(fb)

Graph	Cross Sect(fb)	Error(fb)	Events (K)	Eff	Unwgt	Luminosity
Sum	**43.728**	**0.124**	**62**	**0.7**		
P0_e+e-_mu+mu-	43.728	0.124	62	0.7		463.00

- Generated events can be found in the Events directory
- The format of this file is LHE (run1_unweighted_events.lhe.gz)

The structure of the file with actual events

```
<event>
 4   0  0.4372800E-05  0.9118800E+02  0.7546772E-02  0.1180000E+00
      -11    -1     0     0     0     0  0.00000000000E+00  0.00000000000E+00  0.19915391434E+03  0.19915391434E+03  0.00000000000E+00 0.  -1.
       11    -1     0     0     0     0  0.00000000000E+00  0.00000000000E+00 -0.29258538952E+03  0.29258538952E+03  0.00000000000E+00 0.   1.
      -13     1     1     2     0     0  0.19193037637E+03 -0.80279241429E+02  0.77979187277E+02  0.22217736074E+03  0.00000000000E+00 0.  -1.
       13     1     1     2     0     0 -0.19193037637E+03  0.80279241429E+02 -0.17141066246E+03  0.26956194312E+03  0.00000000000E+00 0.   1.
</event>
```

Working with cards in MadGraph

- Cards are used to change the parameters of the model, to choose the energy of colliding particles and number of generated events, to set the cuts and so on. These cards can be found in Cards directory
- To choose the values of the parameters of the model, masses of gauge bosons, particle widths use param_card.dat
- Energy and type of colliding particles, cuts, PDF-functions can be edited with the help of run_card.dat

param_card.dat

```
Block SMINPUTS      # Standard Model inputs
     1         1.32506980E+02   # alpha_em(MZ)(-1) SM MSbar
     2         1.16639000E-05   # G_Fermi
     3         1.18000000E-01   # alpha_s(MZ) SM MSbar
     4         9.11880000E+01   # Z mass (as input parameter)
Block MGYUKAWA     # Yukawa masses m/v=y/sqrt(2)
#    PDG          YMASS
     5         4.20000000E+00   # mbottom for the Yukawa  y_b
     4         1.42000000E+00   # mcharm  for the Yukawa  y_c
     6         1.64500000E+02   # mtop    for the Yukawa  y_t
    15         1.77700000E+00   # mtau    for the Yukawa  y_ta
Block MGCKM     # CKM elements for MadGraph
     1   1     9.75000000E-01   # Vud for Cabibbo matrix
Block MASS      #  Mass spectrum (kinematic masses)
#       PDG       Mass
         5     4.70000000E+00   # bottom   pole mass
         6     1.74300000E+02   # top      pole mass
        15     1.77700000E+00   # tau      mass
        23     9.11880000E+01   # Z        mass
        24     8.04190000E+01   # W        mass
        25     1.20000000E+02   # H        mass
#            PDG       Width
DECAY         6     1.50833649E+00   # top width
DECAY        23     2.44140351E+00   # Z   width
DECAY        24     2.04759951E+00   # W   width
DECAY        25     5.75308848E-03   # H   width
#        BR             NDA        ID1        ID2
     8.27451012E-02    2           4          -4    # BR( H -> c  cbar  )
     7.17809696E-01    2           5          -5    # BR( H -> b  bbar  )
     0.00000000E+00    2           6          -6    # BR( H -> t  tbar  )
     4.31720144E-02    2          15         -15    # BR( H -> tau- tau+)
     6.90597075E-03    2          23          23    # BR( H -> Z   Z^(*))
     7.45906395E-02    2          24         -24    # BR( H -> W   W^(*))
     3.01765558E-02    2          21          21    # BR( H -> g   g    )
     1.42800773E-03    2          22          22    # BR( H -> A   A    )
```

run_card.dat

```
# Number of events and rnd seed                                        *
#***********************************************************************
  10000       = nevents ! Number of unweighted events requested
      22 = iseed   ! rnd seed (0=assigned automatically=default))
#***********************************************************************
# Collider type and energy                                             *
#***********************************************************************
        1     = lpp1  ! beam 1 type (0=NO PDF)
        1     = lpp2  ! beam 2 type (0=NO PDF)
     7000     = ebeam1  ! beam 1 energy in GeV
     7000     = ebeam2  ! beam 2 energy in GeV
#***********************************************************************
# Beam polarization from -100 (left-handed) to 100 (right-handed)     *
#***********************************************************************
        0     = polbeam1 ! beam polarization for beam 1
        0     = polbeam2 ! beam polarization for beam 2
#***********************************************************************
# PDF CHOICE: this automatically fixes also alpha_s and its evol.     *
#***********************************************************************
 'cteq6l1'    = pdlabel     ! PDF set
#***********************************************************************
# Renormalization and factorization scales                             *
#***********************************************************************
 F        = fixed_ren_scale  ! if .true. use fixed ren scale
 F        = fixed_fac_scale  ! if .true. use fixed fac scale
 91.1880  = scale            ! fixed ren scale
 91.1880  = dsqrt_q2fact1    ! fixed fact scale for pdf1
 91.1880  = dsqrt_q2fact2    ! fixed fact scale for pdf2
 1        = scalefact        ! scale factor for event-by-event scales
```

Type of colliding particles (lpp1 and lpp2 in the card run_card.dat)

- -1: PDF for antiproton
- 0: PDF is switched off
- 1: PDF for proton
- 2: PDF for photon

PDF-functions in MadGraph

name	pdflabel	data file	$\alpha_S(m_Z)$	nloop
MRST2002NLO	`mrs02nl`	mrst2002nlo.dat	0.1197	2
CTEQ6M	`cteq6_m`	cteq6m.tbl	0.118	2
CTEQ6D	`cteq6_d`	cteq6d.tbl	0.118	2
CTEQ6L	`cteq6_l`	cteq6l.tbl	0.118	2
CTEQ6L1	`cteq6l1`	cteq6l1.tbl	0.130	1
CTEQ5M	`cteq5_m`	cteq5m.tbl	0.118	2
CTEQ5L	`cteq5_l`	cteq5l.tbl	0.127	1

Generating events off-line

- Execute the following command in the MG_ME_V4.5.1 directory

  cp –R Template Dir_name

- Folder named Dir_name should appear
- Edit the card proc_card.dat

```
# Begin PROCESS # This is TAG. Do not modify this line

pp>gg  @1             # First Process
QCD=10                # Max QCD couplings
QED=10                # Max QED couplings
end_coup              # End the couplings input

done                  # this tells MG there are no more procs
```

- To generate the code off-line in the Dir_name directory execute

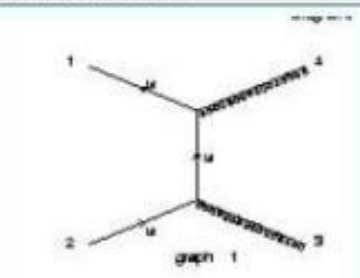
./bin/newprocess

- To see the diagrams open the file index.html

MadEvent Card for pp>gg

Created: Mon Nov 7 14:53:13 CST 2011

Process: **pp>gg**
QCD=10
QED=10
Model: **sm**



Links	Status
Process Information	Generation Complete
Code Download	Available
On-line Event Generation	Available (access restricted)
Results and Event Database	No runs available

- Edit the cards param_card.dat and run_card.dat (e.g. require 200 events)

    - ./bin/generate_events

Hadronization

- Download the Pythia and PGS package from the web-site, install it in the MG_ME_V4.5.1 directory (tar -zxvf …, make)
- For this package to operate you should have cards pythia_card.dat and pgs_card.dat in the Cards directory
- Next step is a usual event generation: the package will operate automatically while the event generation

- After event generation in the Events directory some new files appear such as:

pythia_events.lhe.gz

pythia_events.hep.gz

pgs_events.lhco.gz

Working with the MadAnalysis package

- Download the [MadAnalysis] package from the web-site, install it in the MG_ME_V4.5.1 directory
- Execute ./plots_events, then indicate the pathway to the file with actual events (.lhe or .lhco)
- After the execution of the above command in the MadAnalysis directory you should see the file plot.top (to work with this file you need Topdrawer)

Working with Topdrawer

- Download Topdrawer

(see Downloads/MadAnalysis)

- Version for Windows: in the command line execute the …\td.exe plots.top (file plots.top should be placed to the same folder as the file td.exe)
- Open plots with the help of gsview

Conversion to the ROOT format

- Download the ExRootAnalysis package from the web-site, install it to the MG_ME_V4.5.1 directory
- Enter the ExRootAnalysis directory and execute

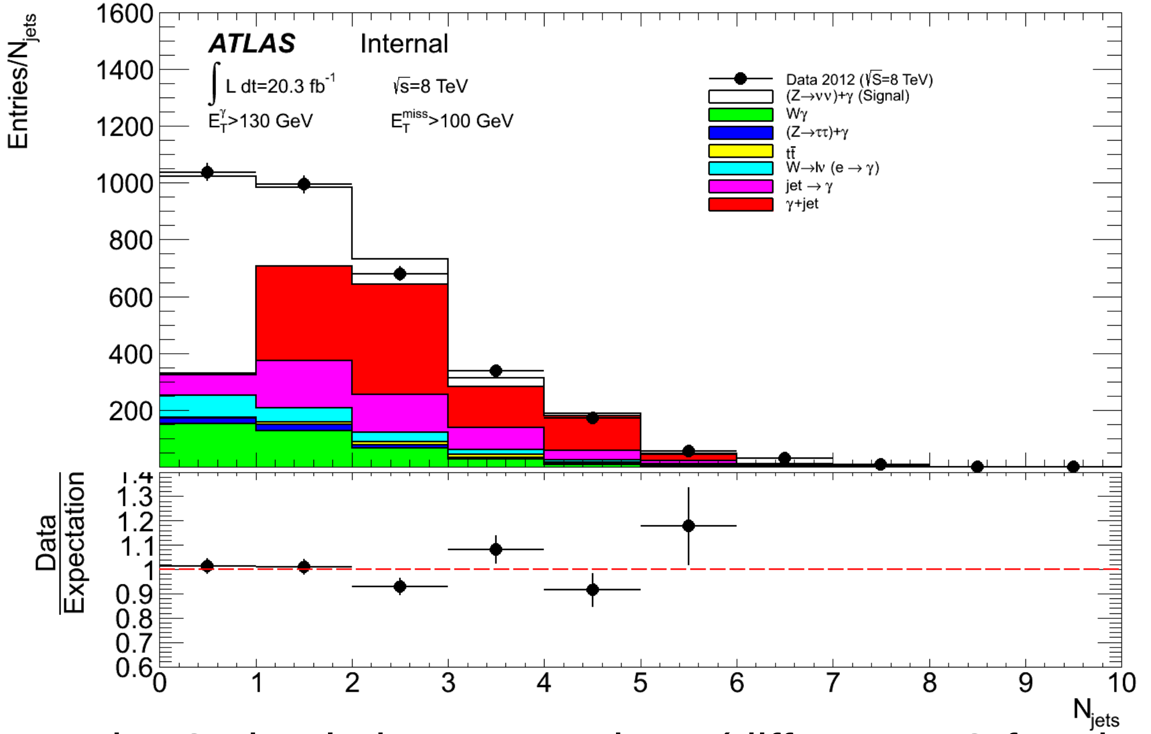
ExRootSTDHEPConverter
"PATH_TO_YOUR_DIRECTORY"/run1_pythia_events.hep
"PATH_TO_YOUR_DIRECTORY"/run1_pythia_events.root

MC generators: experimentalist point of view

Real experiment

We obtain new knowledge from experiment of high energy physics **not in direct way.**

We have to compare real data with MC simulated data treated in the same way.
MC simulation should base on our modern theoretical knowledge.
The target of this course – MC generation - is the start point to obtain MC simulated data (mentioned above).



Real data and MC simulation comparison (different MC for signal and backgrounds)

Structure of the experiment: Where is the place of MC generators?

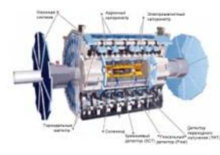
The chain of treatment with data and simulation we can show schematically (below). On this scheme we can show where do we use different program packages and tools to treat (create/change) of the data flows:

Simulation

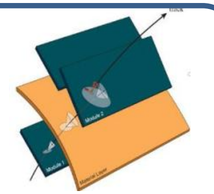
MC Generator

$$\frac{d\sigma_{elas}}{d\Omega} =$$

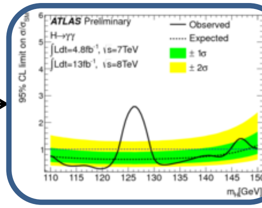
Detector simulator



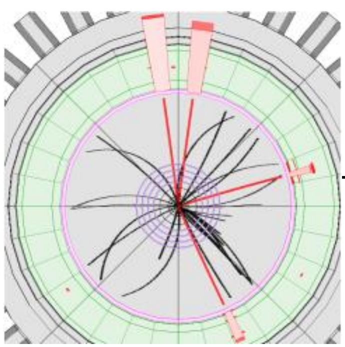
Reconstruction



Analysis

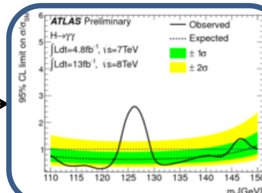


Real experimental data



Reconstruction

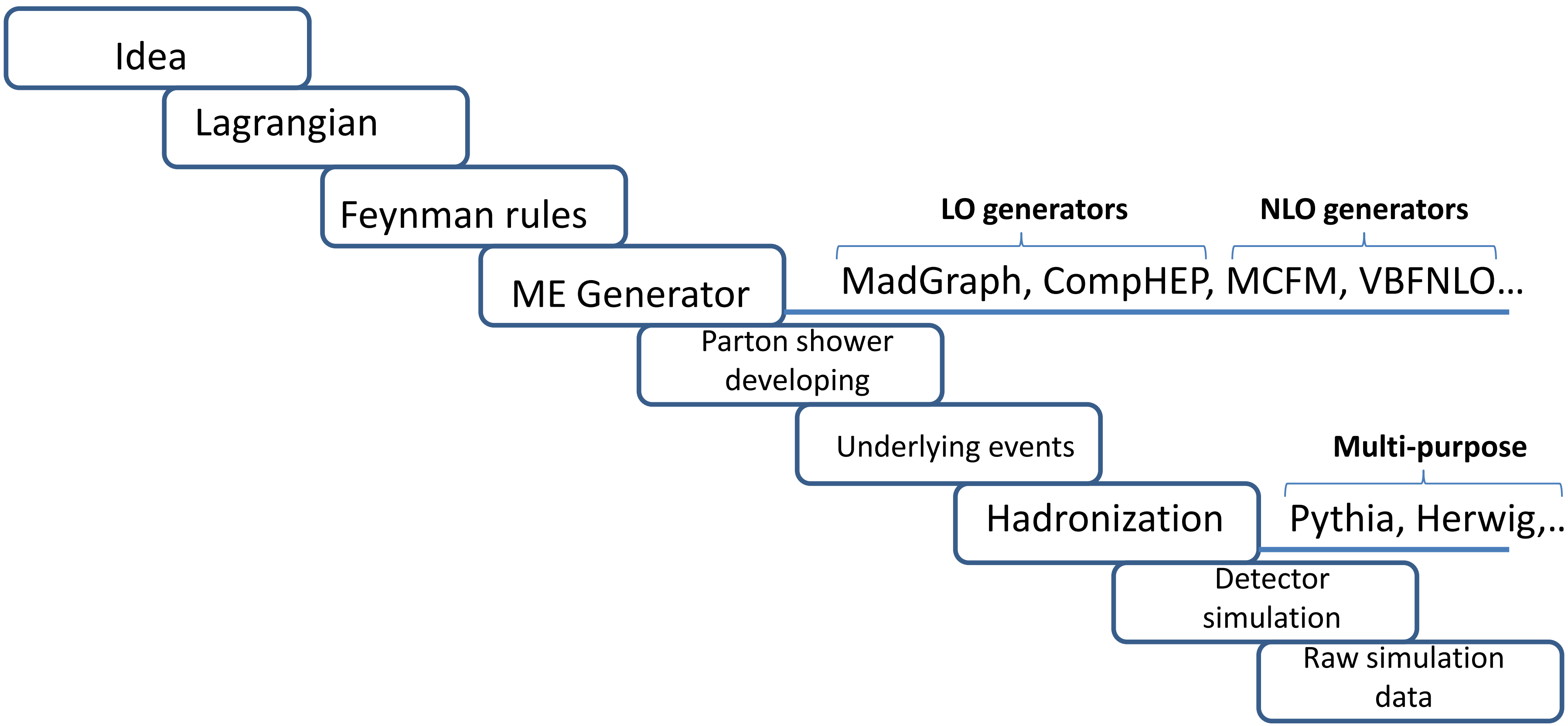
Analysis



Comparison, obtaining of the physical result

MC simulation is more complicated…

MC generator consists on several parts.
There are simple ME generators (calculators) and multi-purpose generators.



Generator MCFM

Generator MCFM

MCFM (*Monte Carlo for FeMtobarn processes*) – parton level ME generator (calculator). This genarator simulates different processes on hadron-hadron colliders with cross-sections on femtobarn level.
For most of the processes matrix elements include more precision cross-section calculation mechanism – NLO (next to leading order) and take into account different spin correlations.

This generator uses Fortran code. However it is under full support even now. It is widely in use for theoretical predictions for the fenmtobarn cross-section level processes.
In particular, it has been used for Tevatron experiments. Currently CERN scientists (from LHC experiments) are using it too.
It gives very good and precise predictions for the cross-sections.

Current version is MCFM-6.6 *(April 2013)*
Site:
http://mcfm.fnal.gov/

Generator MCFM: installation

Download tar archive, unpack.
You can install it with or without additional packages like LHAPDF (additional parton distribution functions), CERNLIB (possibility to make histograms) etc.

(You need Fortran90 compiler)

tar –xvf MCFM-6.6.tar.gz
cd MCFM-6.6
./Install
make

(~5-10 minutes)

```
[esoldato@lxplus425]/afs/cern.ch/work/e/esoldato/workarea/herwig% cd MCFM-6.6
[esoldato@lxplus425]/afs/cern.ch/work/e/esoldato/workarea/herwig/MCFM-6.6% ll
total 48
drwxr-xr-x  3 esoldato zp  2048 Dec  2 00:03 Bin
drwxr-xr-x  2 esoldato zp  2048 Dec  2 00:03 Doc
-rwxr-xr-x  1 esoldato zp  1795 Feb 23  2013 Install
-rw-r--r--  1 esoldato zp 31596 Mar 29  2013 makefile
drwxr-xr-x  5 esoldato zp  2048 Dec  2 00:03 QCDLoop
-rw-r--r--  1 esoldato zp  2652 Mar 29  2013 README
drwxr-xr-x 81 esoldato zp  4096 Dec  2 00:03 src
drwxr-xr-x  6 esoldato zp  2048 Dec  2 00:03 TensorReduction
[esoldato@lxplus425]/afs/cern.ch/work/e/esoldato/workarea/herwig/MCFM-6.6% Install
zsh: command not found: Install
[esoldato@lxplus425]/afs/cern.ch/work/e/esoldato/workarea/herwig/MCFM-6.6% ./Install
Warning: no directory specified for CERNLIB
Warning: no directory specified for LHAPDF
Bin  Doc  Install  makefile  makefile.orig  obj  QCDLoop  README  src  TensorReduction
Compiling QCDLoop library
```

```
make[1]: Leaving directory `/afs/cern.ch/work/e/esoldato/workarea/herwig/MCFM-6.6/TensorReduction/ov'

Installation complete. You may now compile MCFM by running make.
[esoldato@lxplus425]/afs/cern.ch/work/e/esoldato/workarea/herwig/MCFM-6.6%
```

If you need LHAPDF – you have to edit makefile (write the LHAPDF lib path) and execute 'make'

```
o obj/recurrenceB.o obj/recurrenceC.o obj/recurrence.o obj/mcfm.o obj/usercode
mv mcfm Bin/
   ----> MCFM compiled with its own PDFs only <----
   ----> MCFM compiled with histogram output only <----
[esoldato@lxplus425]/afs/cern.ch/work/e/esoldato/workarea/herwig/MCFM-6.6%
```

Generator MCFM: Structure

The directory structure of the installation is as follows:

• *Doc*. The source for this document.

• *Bin*. The directory containing the executable mcfm, and various essential files – notably the options file input.DAT.

• *Bin/Pdfdata*. The directory containing the PDF data-files.

• *obj*. The object files produced by the compiler.

• *src*. The Fortran source files in various subdirectories.

• *QCDLoop*. The source files to version 1.9 of the Fortran library QCDLoop. The location of these libraries is set in the makefile (by QLDIR and FFDIR) and may be changed to reflect existing installations if desired.

```
[esoldato@lxplus425]/afs/cern.ch/work/e/esoldato/workarea/herwig/MCFM-6.6% ll
total 131
drwxr-xr-x  3 esoldato zp  2048 Dec  2 00:14 Bin
drwxr-xr-x  2 esoldato zp  2048 Dec  2 00:06 Doc
-rwxr-xr-x  1 esoldato zp  1795 Feb 23  2013 Install
-rw-r--r--  1 esoldato zp 31660 Dec  2 00:06 makefile
-rw-r--r--  1 esoldato zp 31596 Mar 29  2013 makefile.orig
drwxr-xr-x  2 esoldato zp 53248 Dec  2 00:13 obj
drwxr-xr-x  5 esoldato zp  2048 Dec  2 00:07 QCDLoop
-rw-r--r--  1 esoldato zp  2652 Mar 29  2013 README
drwxr-xr-x 81 esoldato zp  4096 Dec  2 00:03 src
drwxr-xr-x  6 esoldato zp  2048 Dec  2 00:03 TensorReduction
[esoldato@lxplus425]/afs/cern.ch/work/e/esoldato/workarea/herwig/MCFM-6.6%
```

Generator MCFM: Run

To start the mcfm generator you should be in *Bin* folder and execute:
./mcfm [mydir] [myfile.DAT]
By default you'll use *input.DAT* in the same folder as a config file.

input.DAT – file, which contains all main settings

Each parameter in the input file is specified by a line such as
value   [parameter]
and we will give a description of all the parameters below, together with valid and/or sensible inputs for value.

• **file version number**. This should match the version number that is printed when mcfm is executed.

• **nevtrequested**. The default for this parameter is -1 and for the following three parameters it is .false.. This corresponds to the usual mode of operation where the program produces a cross section and a selection of histograms. It is possible to generate n-tuples instead of histograms, as well as unweighted events, for some processes.

Generator MCFM: Settings

• **creategrid**. Flag to control whether or not to write out a grid file suitable for further processing by APPLgrid. Please contact the APPLgrid authors for further information.
• **writetop**. Flag to control whether or not a Topdrawer histogram output file is produced. Please refer to Section 6 for further details.
• **writedat**. Flag to control whether or not the plain histogram output file is produced. Please refer to Section 6 for further details.
• **writegnu**. Flag to control whether or not a gnuplot histogram output file is produced. Please refer to Section 6 for further details.
• **writeroot**. Flag to control whether or not a ROOT script for plotting histograms is produced. Please refer to Section 6 for further details.
• **writepwg**. Flag to control whether a powheg-style analysis file is produced. This option is available only for a limited number of processes. As currently implemented it should be viewed as a development tool, not yet fully supported for the general user.

• **nproc**. The process to be studied is given by choosing a process number, according to Table 9 in Appendix B. f(pi) denotes a generic partonic jet. Processes denoted as "LO" may only be calculated in the Born approximation. For photon processes, "NLO+F" signifies that the calculation may be performed both at NLO and also including the effects of photon fragmentation and experimental isolation.

Generator MCFM: Settings II

• **part**. This parameter has 5 possible values, described below:
– *lord*. The calculation is performed at leading order only.
– *virt*. Virtual (loop) contributions to the next-to-leading order result are calculated (+counterterms to make them finite), including also the lowest order contribution.
12
– *real*. In addition to the loop diagrams calculated by virt, the full next-to-leading order results must include contributions from diagrams involving real gluon emission (-counterterms to make them finite). Note that only the sum of the real and the virt contributions is physical.
– *tota*. For simplicity, the tota option simply runs the virt and real real pieces in series before performing a sum to obtain the full next-to-leading order result. In this case, the number of points specified by ncall1 and ncall2 is automatically increased when performing the real calculation

Generator MCFM: Settings III

• **runstring**. When MCFM is run, it will write output to several files. The label runstring will be appended to the names of these files.
• sqrts. This is the centre-of-mass energy, √s of the colliding particles, measured in GeV.
• **ih1, ih2**. The identities of the incoming hadrons may be set with these parameters, allowing simulations for both p¯p (such as the Tevatron) and pp (such as the LHC). Setting ih1 equal to +1 corresponds to a proton, whilst −1 corresponds to an anti-proton. Values greater than 1000d0 represent a nuclear collision, as described in Section 5.
• **hmass**. For processes involving the Higgs boson, this parameter should be set equal to the putative value of MH.
• **scale**. This parameter may be used to adjust the value of the *renormalization* scale. This is the scale at which αS is evaluated and will
typically be set to a mass scale appropriate to the process (MW, MZ,
Mt for instance). For processes involving vector bosons, setting this
scale to -1d0 chooses a scale equal to the average mass of the bosons
involved.

Generator MCFM: Settings IV

- **itmx1, itmx2**. The program will perform two runs of VEGAS - once for pre-conditioning and then the final run to collect the total crosssection and fill histograms. The number of sweeps for each run is given by itmx1 (pre-conditioning) and itmx2 (final). The default value for both is 10.
- **ncall1, ncall2**. For every sweep of VEGAS, the number of events generated will be ncall1 in the pre-conditioning stage and ncall2 in the final run. The number of events required depends upon a number of factors. The error estimate on a total cross-section will often be reasonable for a fairly small number of events, whereas accurate histograms will require a longer run.
- **pdlabel**. The choice of parton distribution is made by inserting the appropriate 7-character code from Table 3 or 4 here. As mentioned above, this also sets the value of αS(MZ).
- **NGROUP, NSET**. These integers choose the parton distribution functions to be used when using the PDFLIB package.
- **LHAPDF group, LHAPDF set**. These choose the parton distribution functions to be used when using the LHAPDF package – the group is specified by a character string and the set by an integer.

Generator MCFM: Settings V (jets)

• **inclusive**. This logical parameter chooses whether the calculated cross-section should be inclusive in the number of jets found at NLO. An *exclusive cross-section contains the same number of jets at nextto-* leading order as at leading order. An *inclusive cross-section may* instead contain an extra jet at NLO.

• **algorithm**. This specifies the jet-finding algorithm that is used, and can take the values ktal (for the Run II kT -algorithm), ankt (for the "anti-kT " algorithm [6]), cone (for a midpoint cone algorithm), hqrk (for a simplified cone algorithm designed for heavy quark processes) and none (to specify no jet clustering at all). The latter option is only a sensible choice when the leading order cross-section is well-defined without any jet definition: e.g. the single top process, $\bar{q}q' \to t\bar{b}$, which is finite as $p_T(\bar{b}) \to 0$.

• **ptjet min, |etajet| min, |etajet| max**. These specify the values of $p_{T,min}$, $|\eta|_{min}$ and $|\eta|_{max}$ for the jets that are found by the algorithm.

• **Rcut jet**. If the final state of the chosen process contains either quarks or gluons then for each event an attempt will be made to form them into jets. For this it is necessary to define q the jet separation R = η2 + ϕ2 so that after jet combination, all jet pairs are separated by R > Rcut jet.

Generator MCFM: Settings VI (lepton, photon)

- **ptlepton min, |etalepton| max**. These specify the values of pT,min and |η|max for one of the leptons produced in the process.
- **etalepton veto**. This should be specified as a pair of double precision numbers that indicate a rapidity range that should be excluded for the lepton that passes the above cuts.
- **ptmin missing**. Specifies the minimum missing transverse momentum (coming from neutrinos).

- **frag**. This parameter is a logical variable that determines whether the production of photons by a parton fragmentation process is included.
- **ptmin photon**. This specifies the value of pmin
T for the photon with the
largest transverse momentum. Note that this cut, together with all the photon cuts specified in this section of the input file, are applied even if makecuts is set to .false..
- **etamax photon**. This specifies the value of |y|max for any photons produced in the process.
- **R(photon,lept) min**. Using the usual definition of ΔR, this requires that all photon-lepton pairs are separated by ΔR > R(photon,lept) min. This parameter must be non-zero for processes in which photon radiation from leptons is included.

Generator MCFM: processes

nproc	$f(p_1)+f(p_2)\to\dots$	Order
1	$W^+(\to\nu(p_3)+e^+(p_4))$	NLO
6	$W^-(\to e^-(p_3)+\bar{\nu}(p_4))$	NLO
11	$W^+(\to\nu(p_3)+e^+(p_4))+f(p_5)$	NLO
12	$W^+(\to\nu(p_3)+e^+(p_4))+\bar{b}(p_5)$	NLO
13	$W^+(\to\nu(p_3)+e^+(p_4))+\bar{c}(p_5)$	NLO
14	$W^+(\to\nu(p_3)+e^+(p_4))+\bar{c}(p_5)$[massless]	LO
16	$W^-(\to e^-(p_3)+\bar{\nu}(p_4))+f(p_5)$	NLO
17	$W^-(\to e^-(p_3)+\bar{\nu}(p_4))+b(p_5)$	NLO
18	$W^-(\to e^-(p_3)+\bar{\nu}(p_4))+c(p_5)$	NLO
19	$W^-(\to e^-(p_3)+\bar{\nu}(p_4))+c(p_5)$[massless]	LO
20	$W^+(\to\nu(p_3)+e^+(p_4))+b(p_5)+\bar{b}(p_6)$[massive]	NLO
21	$W^+(\to\nu(p_3)+e^+(p_4))+b(p_5)+\bar{b}(p_6)$	NLO
22	$W^+(\to\nu(p_3)+e^+(p_4))+f(p_5)+f(p_6)$	NLO
23	$W^+(\to\nu(p_3)+e^+(p_4))+f(p_5)+f(p_6)+f(p_7)$	LO
24	$W^+(\to\nu(p_3)+e^+(p_4))+b(p_5)+\bar{b}(p_6)+f(p_7)$	LO
25	$W^-(\to e^-(p_3)+\bar{\nu}(p_4))+b(p_5)+\bar{b}(p_6)$[massive]	NLO
26	$W^-(\to e^-(p_3)+\bar{\nu}(p_4))+b(p_5)+\bar{b}(p_6)$	NLO
27	$W^-(\to e^-(p_3)+\bar{\nu}(p_4))+f(p_5)+f(p_6)$	NLO
28	$W^-(\to e^-(p_3)+\bar{\nu}(p_4))+f(p_5)+f(p_6)+f(p_7)$	LO
29	$W^-(\to e^-(p_3)+\bar{\nu}(p_4))+b(p_5)+\bar{b}(p_6)+f(p_7)$	LO
31	$Z^0(\to e^-(p_3)+e^+(p_4))$	NLO
32	$Z^0(\to 3\times(\nu(p_3)+\bar{\nu}(p_4)))$	NLO
33	$Z^0(\to b(p_3)+\bar{b}(p_4))$	NLO
34	$Z^0(\to 3\times(d(p_5)+\bar{d}(p_6)))$	NLO
35	$Z^0(\to 2\times(u(p_5)+\bar{u}(p_6)))$	NLO
36	$Z\to t(\to\nu(p_3)+e^+(p_4)+b(p_5))+\bar{t}(\to\bar{b}(p_6)+e^-(p_7)+\bar{\nu}(p_8))$	LO
41	$Z^0(\to e^-(p_3)+e^+(p_4))+f(p_5)$	NLO
42	$Z_0(\to 3\times(\nu(p_3)+\bar{\nu}(p_4)))+f(p_5)$	NLO
43	$Z^0(\to b(p_3)+\bar{b}(p_4))+f(p_5)$	NLO
44	$Z^0(\to e^-(p_3)+e^+(p_4))+f(p_5)+f(p_6)$	NLO
45	$Z^0(\to e^-(p_3)+e^+(p_4))+f(p_5)+f(p_6)+f(p_7)$	LO
46	$Z^0(\to 3\times(\nu(p_3)+\bar{\nu}(p_4))+f(p_5)+f(p_6)$	NLO
47	$Z^0(\to 3\times(\nu(p_3)+\bar{\nu}(p_4))+f(p_5)+f(p_6)+f(p_7)$	LO
50	$Z^0(\to e^-(p_3)+e^+(p_4))+\bar{b}(p_5)+b(p_6)$[massive]	LO
51	$Z^0(\to e^-(p_3)+e^+(p_4))+b(p_5)+\bar{b}(p_6)$	NLO
52	$Z_0(\to 3\times(\nu(p_3)+\bar{\nu}(p_4)))+b(p_5)+\bar{b}(p_6)$	NLO
53	$Z^0(\to b(p_3)+\bar{b}(p_4))+b(p_5)+\bar{b}(p_6)$	NLO
54	$Z^0(\to e^-(p_3)+e^+(p_4))+b(p_5)+\bar{b}(p_6)+f(p_7)$	LO
56	$Z^0(\to e^-(p_3)+e^+(p_4))+c(p_5)+\bar{c}(p_6)$	NLO
61	$W^+(\to\nu(p_3)+e^+(p_4))+W^-(\to e^-(p_5)+\bar{\nu}(p_6))$	NLO
62	$W^+(\to\nu(p_3)+e^+(p_4))+W^-(\to q(p_5)+\bar{q}(p_6))$	NLO
63	$W^+(\to\nu(p_3)+e^+(p_4))+W^-(\to q(p_5)+\bar{q}(p_6))$[rad.in.dk]	NLO
64	$W^-(\to e^-(p_3)+\bar{\nu}(p_4))W^+(\to q(p_5)+\bar{q}(p_6))$	NLO
65	$W^-(\to e^-(p_3)+\bar{\nu}(p_4))W^+(\to q(p_5)+\bar{q}(p_6))$[rad.in.dk]	NLO
66	$W^+(\to\nu(p_3)+e^+(p_4))+W^-(\to e^-(p_5)+\bar{\nu}(p_6))+f(p_7)$	LO
69	$W^+(\to\nu(p_3)+e^+(p_4))+W^-(\to e^-(p_5)+\bar{\nu}(p_6))$[no pol]	LO
71	$W^+(\to\nu(p_3)+\mu^+(p_4))+Z^0(\to e^-(p_5)+e^+(p_6))$	NLO
72	$W^+(\to\nu(p_3)+\mu^+(p_4))+Z^0(\to 3\times(\nu_e(p_5)+\bar{\nu}_e(p_6)))$	NLO
73	$W^+(\to\nu(p_3)+\mu^+(p_4))+Z^0(\to b(p_5)+\bar{b}(p_6))$	NLO
74	$W^+(\to\nu(p_3)+\mu^+(p_4))+Z^0(\to 3\times(d(p_5)+\bar{d}(p_6)))$	NLO
75	$W^+(\to\nu(p_3)+\mu^+(p_4))+Z^0(\to 2\times(u(p_5)+\bar{u}(p_6)))$	NLO
76	$W^-(\to\mu^-(p_3)+\bar{\nu}(p_4))+Z^0(\to e^-(p_5)+e^+(p_6))$	NLO
77	$W^-(\to e^-(p_3)+\bar{\nu}(p_4))+Z^0(\to 3\times(\nu_e(p_5)+\bar{\nu}_e(p_6)))$	NLO
78	$W^-(\to e^-(p_3)+\bar{\nu}(p_4))+Z^0(\to b(p_5)+\bar{b}(p_6))$	NLO
79	$W^-(\to e^-(p_3)+\bar{\nu}(p_4))+Z^0(\to 3\times(d(p_5)+\bar{d}(p_6)))$	NLO
80	$W^-(\to e^-(p_3)+\bar{\nu}(p_4))+Z^0(\to 2\times(u(p_5)+\bar{u}(p_6)))$	NLO
81	$Z^0(\to\mu^-(p_3)+\mu^+(p_4))+Z^0(\to e^-(p_5)+e^+(p_6))$	NLO
82	$Z^0(\to e^-(p_3)+e^+(p_4))+Z^0(\to 3\times(\nu(p_5)+\bar{\nu}(p_6)))$	NLO
83	$Z^0(\to e^-(p_3)+e^+(p_4))+Z^0(\to b(p_5)+\bar{b}(p_6))$	NLO
84	$Z^0(\to b(p_3)+\bar{b}(p_4))+Z^0(\to 3\times(\nu(p_5)+\bar{\nu}(p_6)))$	NLO
85	$Z^0(\to e^-(p_3)+e^+(p_4))+Z^0(\to 3\times(\nu(p_5)+\bar{\nu}(p_6)))+f(p_7)$	LO
86	$Z^0(\to\mu^-(p_3)+\mu^+(p_4))+Z^0(\to e^-(p_5)+e^+(p_6))$[no gamma*]	NLO
87	$Z^0(\to e^-(p_3)+e^+(p_4))+Z^0(\to 3\times(\nu(p_5)+\bar{\nu}(p_6)))$[no gamma*]	NLO
88	$Z^0(\to e^-(p_3)+e^+(p_4))+Z^0(\to b(p_5)+\bar{b}(p_6))$[no gamma*]	NLO
89	$Z^0(\to b(p_3)+\bar{b}(p_4))+Z^0(\to 3\times(\nu(p_5)+\bar{\nu}(p_6)))$[no gamma*]	NLO
90	$Z^0(\to e^-(p_3)+e^+(p_4))+Z^0(\to e^-(p_5)+e^+(p_6))$	NLO
91	$W^+(\to\nu(p_3)+e^+(p_4))+H(\to b(p_5)+\bar{b}(p_6))$	NLO
92	$W^+(\to\nu(p_3)+e^+(p_4))+H(\to W^+(\nu(p_5),e^+(p_6))W^-(e^-(p_7),\bar{\nu}(p_8)))$	NLO
93	$W^+(\to\nu(p_3)+e^+(p_4))+H(\to Z(e^-(p_5),e^+(p_6))+Z(\mu^-(p_7),\mu(p_8)))$	NLO
94	$W^+(\to\nu(p_3)+e^+(p_4))+H(\to\gamma(p_5)+\gamma(p_6)$	NLO
96	$W^-(\to e^-(p_3)+\bar{\nu}(p_4))+H(\to b(p_5)+\bar{b}(p_6))$	NLO
97	$W^-(\to e^-(p_3)+\bar{\nu}(p_4))+H(\to W^+(\nu(p_5),e^+(p_6))W^-(e^-(p_7),\bar{\nu}(p_8)))$	NLO
98	$W^-(\to e^-(p_3)+\bar{\nu}(p_4))+H(\to Z(e^-(p_5),e^+(p_6))+Z(\mu^-(p_7),\mu^+(p_8)))$	NLO
99	$W^-(\to e^-(p_3)+\bar{\nu}(p_4))+H(\to\gamma(p_5)+\gamma(p_6))$	NLO

Generator MCFM: processes II

101	$Z^0(\to e^-(p_3)+e^+(p_4))+H(\to b(p_5)+\bar{b}(p_6))$	NLO
102	$Z^0(\to 3\times(\nu(p_3)+\bar{\nu}(p_4)))+H(\to b(p_5)+\bar{b}(p_6))$	NLO
103	$Z^0(\to b(p_3)+\bar{b}(p_4))+H(\to b(p_5)+\bar{b}(p_6))$	NLO
104	$Z^0(\to e^-(p_3)+e^+(p_4))+H(\to \gamma(p_5)+\gamma(p_6))$	NLO
105	$Z^0(\to\to 3\times(\nu(p_3)+\bar{\nu}(p_4)))+H(\to \gamma(p_5)+\gamma(p_6))$	NLO
106	$Z^0(\to e^-(p_3)+e^+(p_4))+H(\to W^+(\nu(p_5),e^+(p_6))W^-(e^-(p_7),\bar{\nu}(p_8)))$	NLO
107	$Z^0(\to 3\times(\nu(p_3)+\bar{\nu}(p_4)))+H(\to W^+(\nu(p_5),e^+(p_6))W^-(e^-(p_7),\bar{\nu}(p_8)))$	NLO
108	$Z^0(\to b(p_3)+\bar{b}(p_4))+H(\to W^+(\nu(p_5),e^+(p_6))W^-(e^-(p_7),\bar{\nu}(p_8)))$	NLO
109	$Z^0(\to e^-(p_3)+e^+(p_4))+H(\to Z(e^-(p_5),e^+(p_6))+Z(\mu^-(p_7),\mu^+(p_8)))$	NLO
111	$H(\to b(p_3)+\bar{b}(p_4))$	NLO
112	$H(\to \tau^-(p_3)+\tau^+(p_4))$	NLO
113	$H(\to W^+(\nu(p_3)+e^+(p_4))+W^-(e^-(p_5)+\bar{\nu}(p_6)))$	NLO
114	$H(\to W^+(\nu(p_3)+e^+(p_4))+W^-(q(p_5)+\bar{q}(p_6)))$	NLO
115	$H(\to W^+(\nu(p_3)+e^+(p_4))+W^-(q(p_5)+\bar{q}(p_6)))[rad.in.dk]$	NLO
116	$H(\to Z^0(\mu^-(p_3)+\mu^+(p_4))+Z^0(e^-(p_5)+e^+(p_6))$	NLO
117	$H(\to Z^0(3\times(\nu(p_3)+\bar{\nu}(p_4)))+Z^0(e^-(p_5)+e^+(p_6))$	NLO
118	$H(\to Z^0(\mu^-(p_3)+\mu^+(p_4))+Z^0(b(p_5)+\bar{b}(p_6))$	NLO
119	$H(\to \gamma(p_3)+\gamma(p_4))$	NLO
120	$H(\to Z^0(\mu^-(p_3)+\mu^+(p_4))+\gamma(p_5))$	NLO
121	$H(\to Z^0(3\times(\nu(p_3)+\bar{\nu}(p_4))))+\gamma(p_5))$	NLO
126	$H(\to W^+(\nu(p_3)+e^+(p_4))+W^-(e^-(p_5)+\bar{\nu}(p_6)))$[top, bottom loops, exact]	LO
127	$H(\to W^+(\nu(p_3)+e^+(p_4))+W^-(e^-(p_5)+\bar{\nu}(p_6)))$[+ interf. with $gg\to WW$]	LO
131	$H(\to b(p_3)+\bar{b}(p_4))+b(p_5)(+g(p_6))$	NLO
132	$H(\to b(p_3)+\bar{b}(p_4))+\bar{b}(p_5)(+b(p_6))$	(REAL)
133	$H(\to b(p_3)+\bar{b}(p_4))+b(p_5)+\bar{b}(p_6)$[both observed]	(REAL)
141	$t(\to \nu(p_3)+e^+(p_4)+b(p_5))+\bar{t}(\to b\ (p_6)+e^-(p_7)+\bar{\nu}(p_8))$	NLO
142	$t(\to \nu(p_3)+e^+(p_4)+b(p_5))+\bar{t}(\to b\ (p_6)+e^-(p_7)+\bar{\nu}(p_8))$ [rad.in.dk]	NLO
143	$t(\to \nu(p_3)+e^+(p_4)+b(p_5))+\bar{t}(\to b\ (p_6)+e^-(p_7)+\bar{\nu}(p_8))+f(p_9)$	LO
144	$t(\to \nu(p_3)+e^+(p_4)+b(p_5))+\bar{t}(\to b\ (p_6)+e^-(p_7)+\bar{\nu}(p_8))$ (uncorr)	NLO
145	$t(\to \nu(p_3)+e^+(p_4)+b(p_5))+\bar{t}(\to b\ (p_6)+e^-(p_7)+\bar{\nu}(p_8))$ [rad.in.dk],uncorr	NLO
146	$t(\to \nu(p_3)+e^+(p_4)+b(p_5))+\bar{t}(\to b\ (p_6)+q(p_7)+\bar{q}(p_8))$	NLO
147	$t(\to \nu(p_3)+e^+(p_4)+b(p_5))+\bar{t}(\to b\ (p_6)+q(p_7)+\bar{q}(p_8))$ [rad.in.top.dk]	NLO
148	$t(\to \nu(p_3)+e^+(p_4)+b(p_5))+\bar{t}(\to b\ (p_6)+q(p_7)+\bar{q}(p_8))$ [rad.in.W.dk]	NLO
149	$t(\to q(p_3)+\bar{q}(p_4)+b(p_5))+\bar{t}(\to b\ (p_6)+e^-(p_7)+\bar{\nu}(p_8))$	NLO
150	$t(\to q(p_3)+\bar{q}(p_4)+b(p_5))+\bar{t}(\to b\ (p_6)+e^-(p_7)+\bar{\nu}(p_8))$ [rad.in.top.dk]	NLO
151	$t(\to q(p_3)+\bar{q}(p_4)+b(p_5))+\bar{t}(\to b\ (p_6)+e^-(p_7)+\bar{\nu}(p_8))$ [rad.in.W.dk]	NLO

157	$t\bar{t}$[for total Xsect]	NLO
158	$b\bar{b}$[for total Xsect]	NLO
159	$c\bar{c}$[for total Xsect]	NLO
160	$t\bar{t}+g$[for total Xsect]	LO
161	$t(\to \nu(p_3)+e^+(p_4)+b(p_5))+q(p_6)$[t-channel]	NLO
162	$t(\to \nu(p_3)+e^+(p_4)+b(p_5))+q(p_6)$[decay]	NLO
163	$t(\to \nu(p_3)+e^+(p_4)+b(p_5))+q(p_6)$[t-channel]$mb>0$	NLO
166	$\bar{t}(\to e^-(p_3)+\bar{\nu}(p_4)+\bar{b}(p_5))+q(p_6)$[t-channel]	NLO
167	$\bar{t}(\to e^-(p_3)+\bar{\nu}(p_4)+\bar{b}(p_5))+q(p_6)$[rad.in.dk]	NLO
168	$\bar{t}(\to e^-(p_3)+\bar{\nu}(p_4)+\bar{b}(p_5))+q(p_6)$[t-channel]$mb>0$	NLO
171	$t(\to \nu(p_3)+e^+(p_4)+b(p_5))+\bar{b}(p_6))$[s-channel]	NLO
172	$t(\to \nu(p_3)+e^+(p_4)+b(p_5))+\bar{b}(p_6))$[decay]	NLO
176	$\bar{t}(\to e^-(p_3)+\bar{\nu}(p_4)+\bar{b}(p_5))+b(p_6))$[s-channel]	NLO
177	$\bar{t}(\to e^-(p_3)+\bar{\nu}(p_4)+\bar{b}(p_5))+b(p_6))$[rad.in.dk]	NLO
180	$W^-(\to e^-(p_3)+\bar{\nu}(p_4))+t(p_5)$	NLO
181	$W^-(\to e^-(p_3)+\bar{\nu}(p_4))+t(\nu(p_5)+e^+(p_6)+b(p_7))$	NLO
182	$W^-(\to e^-(p_3)+\bar{\nu}(p_4))+t(\nu(p_5)+e^+(p_6)+b(p_7))$[rad.in.dk]	NLO
183	$W^-(\to e^-(p_3)+\bar{\nu}(p_4))+t(\nu(p_5)+e^+(p_6)+b(p_7))+b(p_8)$	LO
184	$W^-(\to e^-(p_3)+\bar{\nu}(p_4))+t(p_5)+b(p_6)$[massive b]	LO
185	$W^+(\to \nu(p_3)+e^+(p_4))+\bar{t}(p_5)$	NLO
186	$W^+(\to \nu(p_3)+e^+(p_4))+\bar{t}(e^-(p_5)+\bar{\nu}(p_6)+\bar{b}(p_7)$	NLO
187	$W^+(\to \nu(p_3)+e^+(p_4))+\bar{t}(e^-(p_5)+\bar{\nu}(p_6)+\bar{b}(p_7)$[rad.in.dk]	NLO

Generator MCFM: processes III

201	$H(\to b(p_3)+\bar{b}(p_4))+f(p_5)$[full mt dep.]	LO
202	$H(\to \tau^-(p_3)+\tau^+(p_4))+f(p_5)$[full mt dep.]	LO
203	$H(\to b(p_3)+\bar{b}(p_4))+f(p_5)$	NLO
204	$H(\to \tau^-(p_3)+\tau^+(p_4))+f(p_5)$	NLO
206	$A(\to b(p_3)+\bar{b}(p_4))+f(p_5)$[full mt dep.]	LO
207	$A(\to \tau^-(p_3)+\tau^+(p_4))+f(p_5)$[full mt dep.]	LO
208	$H(\to W^+(\nu(p_3),e^+(p_4))W^-(e^-(p_5),\bar{\nu}(p_6)))+f(p_7)$	NLO
209	$H(\to Z^+(e^-(p_3),e^+(p_4))Z(\mu^-(p_5),\mu^+(p_6)))+f(p_7)$	NLO
210	$H(\to \gamma(p_3)+\gamma(p_4))+f(p_5)$	NLO
211	$H(\to b(p_3)+\bar{b}(p_4))+f(p_5)+f(p_6)$[WBF]	NLO
212	$H(\to \tau^-(p_3)+\tau^+(p_4))+f(p_5)+f(p_6)$[WBF]	NLO
213	$H(\to W^+(\nu(p_3),e^+(p_4))W^-(e^-(p_5),\bar{\nu}(p_6)))+f(p_7)+f(p_8)$[WBF]	NLO
214	$H(\to Z(e^-(p_3),e^+(p_4))+Z(\mu^-(p_5),\mu^+(p_6)))+f(p_7)+f(p_8)$[WBF]	NLO
215	$H(\to \gamma(p_3)+\gamma(p_4))+f(p_5)+f(p_6)$[WBF]	NLO
216	$H(\to b(p_3)+\bar{b}(p_4))+f(p_5)+f(p_6)+f(p_7)$[WBF+jet]	LO
217	$H(\to \tau^-(p_3)+\tau^+(p_4))+f(p_5)+f(p_6)+f(p_7)$[WBF+jet]	LO
221	$\tau^-(\to e^-(p_3)+\bar{\nu}_e(p_4)+\nu_\tau(p_5))+\tau^+(\to \bar{\nu}_\tau(p_6)+\nu_e(p_7)+e^+(p_8))$	LO
231	$t(p_3)+\bar{b}(p_4)+q(p_5)$ [t-channel]	NLO
232	$t(p_3)+\bar{b}(p_4)+q(p_5)+q(p_6)$ [t-channel]	LO
233	$t(\to \nu(p_3)+e^+(p_4)+b(p_5))+\bar{b}(p_6)+q(p_7)$ [t-channel]	NLO
234	$t(\to \nu(p_3)+e^+(p_4)+b(p_5))+\bar{b}(p_6)+q(p_7)$ [t-channel, rad.in.dk]	NLO
235	$t(\to \nu(p_3)+e^+(p_4)+b(p_5))+\bar{b}(p_6)+q(p_7)+f(p_8)$ [t-channel]	LO
236	$\bar{t}(p_3)+b(p_4)+q(p_5)$ [t-channel]	NLO
237	$\bar{t}(p_3)+b(p_4)+q(p_5)+q(p_6)$ [t-channel]	LO
238	$\bar{t}(\to e^-(p_3)+\bar{\nu}(p_4)+\bar{b}(p_5))+b(p_6)+q(p_7)$ [t-channel]	NLO
239	$\bar{t}(\to e^-(p_3)+\bar{\nu}(p_4)+\bar{b}(p_5))+b(p_6)+q(p_7)$ [t-channel, rad.in.dk]	NLO
240	$\bar{t}(\to e^-(p_3)+\bar{\nu}(p_4)+\bar{b}(p_5))+b(p_6)+q(p_7)+f(p_8)$ [t-channel]	L0
251	$W^+(\to \nu(p_3)+e^+(p_4))+W^+(\to \nu(p_5)+e^+(p_6))+f(p_7)+f(p_8)$	LO
252	$W^+(\to \nu(p_3)+e^+(p_4))+W^+(\to \nu(p_5)+e^+(p_6))+f(p_7)+f(p_8)+f(p_9)$	LO
253	$W^+(\to \nu(p_3)+e^+(p_4))+Z(-->e^-(p_5)+e^+(p_6))+f(p_7)+f(p_8)$	LO
254	$W^-(-->e^-(p_3)+\bar{\nu}(p_4))+Z(-->e^-(p_5)+e^+(p_6))+f(p_7)+f(p_8)$	LO
255	$W^+(-->\nu(p_3)+e^+(p_4))+Z(-->e^-(p_5)+e^+(p_6))+b(p_7)+f(p_8)$	LO
256	$W^-(-->e^-(p_3)+\bar{\nu}(p_4))+Z(-->e^-(p_5)+e^+(p_6))+b(p_7)+f(p_8)$	LO
259	$W^+(-->\nu(p_3)+e^+(p_4))+Z(-->e^-(p_5)+e^+(p_6))+b(p_7)+b\ (p_8)$	LO
260	$W^-(-->e^-(p_3)+\bar{\nu}(p_4))+Z(-->e^-(p_5)+e^+(p_6))+b(p_7)+b\ (p_8)$	LO
261	$Z^0(\to e^-(p_3)+e^+(p_4))+b(p_5)$	NLO
262	$Z^0(\to e^-(p_3)+e^+(p_4))+c(p_5)$	NLO
263	$Z^0(\to e^-(p_3)+e^+(p_4))+\bar{b}(p_5)+b(p_6)$[1 b-tag]	LO
264	$Z^0(\to e^-(p_3)+e^+(p_4))+\bar{c}(p_5)+c(p_6)$[1 c-tag]	LO
266	$Z^0(\to e^-(p_3)+e^+(p_4))+b(p_5)(+\bar{b}(p_6))$	NLO
267	$Z^0(\to e^-(p_3)+e^+(p_4))+c(p_5)(+\bar{c}(p_6))$	NLO

270	$H(\gamma(p_3)+\gamma(p_4))+f(p_5)+f(p_6)$[in heavy top limit]	NLO
271	$H(b(p_3)+\bar{b}(p_4))+f(p_5)+f(p_6)$[in heavy top limit]	NLO
272	$H(\tau^-(p_3)+\tau^+(p_4))+f(p_5)+f(p_6)$[in heavy top limit]	NLO
273	$H(\to W^+(\nu(p_3),e^+(p_4))W^-(e^-(p_5),\bar{\nu}(p_6)))+f(p_7)+f(p_8)$	NLO
274	$H(\to Z(e^-(p_3),e^+(p_4))Z(\mu^-(p_5),\mu^+(p_6)))+f(p_7)+f(p_8)$	NLO
275	$H(b(p_3)+\bar{b}(p_4))+f(p_5)+f(p_6)+f(p_7)$[in heavy top limit]	LO
276	$H(\tau^-(p_3)+\tau^+(p_4))+f(p_5)+f(p_6)+f(p_7)$[in heavy top limit]	LO
278	$H(\to W^+(\nu(p_3),e^+(p_4))W^-(e^-(p_5),\bar{\nu}(p_6)))+f(p_7)+f(p_8)+f(p_9)$	LO
279	$H(\to Z(e^-(p_3),e^+(p_4))Z(\mu^-(p_5),\mu^+(p_6)))+f(p_7)+f(p_8)+f(p_9)$	LO
280	$\gamma(p_3)+f(p_4)$	NLO+F
282	$f(p_1)+f(p_2)\to \gamma(p_3)+f(p_4)+f(p_5)$	LO
283	$f(p_1)+f(p_2)\to \gamma(p_3)+b(p_4)$	LO
284	$f(p_1)+f(p_2)\to \gamma(p_3)+c(p_4)$	LO
285	$f(p_1)+f(p_2)\to \gamma(p_3)+\gamma(p_4)$	NLO+F
286	$f(p_1)+f(p_2)\to \gamma(p_3)+\gamma(p_4)+f(p_5)$	LO
290	$W^+(\to \nu(p_3)+e^+(p_4))+\gamma(p_5)$	NLO+F
292	$W^+(\to \nu(p_3)+e^+(p_4))+\gamma(p_5)+f(p_6)$	LO
295	$W^-(\to e^-(p_3)+\bar{\nu}(p_4))+\gamma(p_5)$	NLO+F
297	$W^-(\to e^-(p_3)+\bar{\nu}(p_4))+\gamma(p_5)+f(p_6)$	LO
300	$Z^0(\to e^-(p_3)+e^+(p_4))+\gamma(p_5)$	NLO+F
301	$Z^0(\to e^-(p_3)+e^+(p_4))+\gamma(p_5)+\gamma(p_6)$	NLO +F
302	$Z^0(\to e^-(p_3)+e^+(p_4))+\gamma(p_5)+f(p_6)$	NLO + F
303	$Z^0(\to e^-(p_3)+e^+(p_4))+\gamma(p_5)+\gamma(p_6)+f(p_7)$	LO
304	$Z^0(\to e^-(p_3)+e^+(p_4))+\gamma(p_5)+f(p_6)+f(p_7)$	LO
305	$Z^0(\to 3(\nu(p_3)+\bar{\nu}(p_4)))+\gamma(p_5)$	NLO + F
306	$Z^0(\to 3(\nu(p_3)+\bar{\nu}(p_4)))+\gamma(p_5)+\gamma(p_6)$	NLO + F
307	$Z^0(\to 3(\nu(p_3)+\bar{\nu}(p_4)))+\gamma(p_5)+f(p_6)$	NLO + F
308	$Z^0(\to 3(\nu(p_3)+\bar{\nu}(p_4)))+\gamma(p_5)+\gamma(p_6)+f(p_7)$	LO
309	$Z^0(\to 3(\nu(p_3)+\bar{\nu}(p_4)))+\gamma(p_5)+f(p_6)+f(p_7)$	LO
311	$f(p_1)+b(p_2)\to W^+(\to \nu(p_3)+e^+(p_4))+b(p_5)+f(p_6)$	LO
316	$f(p_1)+b(p_2)\to W^-(\to e^-(p_3)+\bar{\nu}(p_4))+b(p_5)+f(p_6)$	LO
321	$f(p_1)+c(p_2)\to W^+(\to \nu(p_3)+e^+(p_4))+c(p_5)+f(p_6)$	LO
326	$f(p_1)+c(p_2)\to W^-(\to e^-(p_3)+\bar{\nu}(p_4))+c(p_5)+f(p_6)$	LO
331	$W^+(\to \nu(p_3)+e^+(p_4))+c(p_5)+f(p_6)$[c-s interaction]	LO
336	$W^-(\to e^-(p_3)+\bar{\nu}(p_4))+c(p_5)+f(p_6)$[c-s interaction]	LO
341	$f(p_1)+b(p_2)\to Z^0(\to e^-(p_3)+e^+(p_4))+b(p_5)+f(p_6)[+f(p_7)]$	NLO
342	$f(p_1)+b(p_2)\to Z^0(\to e^-(p_3)+e^+(p_4))+b(p_5)+f(p_6)[+\bar{b}(p_7)]$	(REAL)
346	$f(p_1)+b(p_2)\to Z^0(\to e^-(p_3)+e^+(p_4))+b(p_5)+f(p_6)+f(p_7)$	LO
347	$f(p_1)+b(p_2)\to Z^0(\to e^-(p_3)+e^+(p_4))+b(p_5)+f(p_6)+\bar{b}(p_7)$	LO

Generator MCFM: processes IV

351	$f(p_1) + c(p_2) \to Z^0(\to e^-(p_3) + e^+(p_4)) + c(p_5) + f(p_6)[+f(p_7)]$
352	$f(p_1) + c(p_2) \to Z^0(\to e^-(p_3) + e^+(p_4)) + c(p_5) + f(p_6)[+\bar{c}(p_7)]$
356	$f(p_1) + c(p_2) \to Z^0(\to e^-(p_3) + e^+(p_4)) + c(p_5) + f(p_6) + f(p_7)$
357	$f(p_1) + c(p_2) \to Z^0(\to e^-(p_3) + e^+(p_4)) + c(p_5) + f(p_6) + \bar{c}(p_7)$
361	$c(p_1) + \bar{s}(p_2) \to W^+(\to \nu(p_3) + e^+(p_4))$[mc=0 in NLO]
362	$c(p_1) + \bar{s}(p_2) \to W^+(\to \nu(p_3) + e^+(p_4))$[massless corrections only]
363	$c(p_1) + \bar{s}(p_2) \to W^+(\to \nu(p_3) + e^+(p_4))$[massive charm in real]
401	$W^+(\to \nu(p_3) + e^+(p_4)) + b(p_5)$ [1,2 or 3 jets, 4FNS]
402	$W^+(\to \nu(p_3) + e^+(p_4)) + (b + \bar{b})(p_5)$ [1 or 2 jets, 4FNS]
403	$W^+(\to \nu(p_3) + e^+(p_4)) + b(p_5) + \bar{b}(p_6)$ [2 or 3 jets, 4FNS]
406	$W^-(\to e^-(p_3) + \bar{\nu}(p_4)) + b(p_5)$ [1,2 or 3 jets, 4FNS]
407	$W^-(\to e^-(p_3) + \bar{\nu}(p_4)) + (b + \bar{b})(p_5)$ [1 or 2 jets, 4FNS]
408	$W^-(\to e^-(p_3) + \bar{\nu}(p_4)) + b(p_5) + \bar{b}(p_6)$ [2 or 3 jets, 4FNS]
411	$f(p_1) + b(p_2) \to W^+(\to \nu(p_3) + e^+(p_4)) + b(p_5) + f(p_6)$ [5FNS]
416	$f(p_1) + b(p_2) \to W^-(\to e^-(p_3) + \bar{\nu}(p_4)) + b(p_5) + f(p_6)$ [5FNS]
421	$W^+(\to \nu(p_3) + e^+(p_4)) + b(p_5)$ [1,2 or 3 jets, 4FNS+5FNS]
426	$W^-(\to e^-(p_3) + \bar{\nu}(p_4)) + b(p_5)$ [1,2 or 3 jets, 4FNS+5FNS]
431	$W^+(\to \nu(p_3) + e^+(p_4)) + b(p_5) + \bar{b}(p_6) + f(p_7)$ [massive]
436	$W^-(\to e^-(p_3) + \bar{\nu}(p_4)) + b(p_5) + \bar{b}(p_6) + f(p_7)$ [massive]
500	$W^+(\to \nu(p_3) + e^+(p_4)) + t(p_5) + \bar{t}(p_6)$[massive]
501	$t(\to \nu(p_3) + e^+(p_4) + b(p_5)) + \bar{t}(\to b\ (p_6) + e^-(p_7) + \bar{\nu}(p_8)) + W^+(\nu(p_9), \mu^+(p_{10}))$
502	$t(\to \nu(p_3) + e^+(p_4) + b(p_5)) + \bar{t}(\to b\ (p_6) + e^-(p_7) + \bar{\nu}(p_8)) + W^+(\nu(p_9), \mu^+(p_{10}))$[rad.in.dk]
503	$t(\to \nu(p_3) + e^+(p_4) + b(p_5)) + \bar{t}(\to b\ (p_6) + q(p_7) + q\ (p_8)) + W^+(\nu(p_9), \mu^+(p_{10}))$
506	$t(\to q(p_3) + q\ (p_4) + b(p_5)) + \bar{t}(\to b\ (p_6) + e^-(p_7) + \bar{\nu}(p_8)) + W^+(\nu(p_9), \mu^+(p_{10}))$
510	$W^-(\to e^-(p_3) + \bar{\nu}(p_4)) + t(p_5) + \bar{t}(p_6)$[massive]
511	$t(\to \nu(p_3) + e^+(p_4) + b(p_5)) + \bar{t}(\to b\ (p_6) + e^-(p_7) + \bar{\nu}(p_8)) + W^-(\mu^-(p_9), \bar{\nu}(p_{10}))$
512	$t(\to \nu(p_3) + e^+(p_4) + b(p_5)) + \bar{t}(\to b\ (p_6) + e^-(p_7) + \bar{\nu}(p_8)) + W^-(\mu^-(p_9), \bar{\nu}(p_{10}))$[rad.in.dk]
513	$t(\to \nu(p_3) + e^+(p_4) + b(p_5)) + \bar{t}(\to b\ (p_6) + q(p_7) + q\ (p_8)) + W^-(\mu^-(p_9), \bar{\nu}(p_{10}))$
516	$t(\to q(p_3) + q\ (p_4) + b(p_5)) + \bar{t}(\to b\ (p_6) + e^-(p_7) + \bar{\nu}(p_8)) + W^-(\mu^-(p_9), \bar{\nu}(p_{10}))$
529	$Z^0(\to e^-(p_3) + e^+(p_4)) + t(p5) + \bar{t}(p_6)$
530	$t(\to \nu(p_3) + e^+(p_4) + b(p_5)) + \bar{t}(\to e^-(p_7) + \bar{\nu}(p_8) + b\ (p_6)) + Z(e^-(p_9), e^+(p_{10}))$
531	$t(\to \nu(p_3) + e^+(p_4) + b(p_5)) + \bar{t}(\to e^-(p_7) + \bar{\nu}(p_8) + b\ (p_6)) + Z(b(p_9), b\ (p_{10}))$
532	$t(\to \nu(p_3) + e^+(p_4) + b(p_5)) + \bar{t}(\to q(p_7) + \bar{q}(p_8) + b\ (p_6)) + Z(e^-(p_9), e^+(p_{10}))$
533	$t(\to q(p_3) + \bar{q}(p_4) + b(p_5)) + \bar{t}(\to e^-(p_7) + \bar{\nu}(p_8) + b\ (p_6)) + Z(e^-(p_9), e^+(p_{10}))$

540	$H(b(p_3) + \bar{b}(p_4)) + t(p_5) + q(p_6)$	N
541	$H(b(p_3) + \bar{b}(p_4)) + \bar{t}(p_5) + q(p_6)$	N
544	$H(b(p_3) + \bar{b}(p_4)) + t(\nu(p_5) + e^+(p_6) + b(p_7)) + q(p_9)$	N
547	$H(b(p_3) + \bar{b}(p_4)) + \bar{t}(e^-(p_5) + \bar{\nu}(p_6) + b(p_7)) + q(p_9)$	N
550	$H(\gamma(p_3) + \gamma(p_4)) + t(p_5) + q(p_6)$	N
551	$H(\gamma(p_3) + \gamma(p_4)) + \bar{t}(p_5) + q(p_6)$	N
554	$H(\gamma(p_3) + \gamma(p_4)) + t(\nu(p_5) + e^+(p_6) + b(p_7)) + q(p_9)$	N
557	$H(\gamma(p_3) + \gamma(p_4)) + \bar{t}(e^-(p_5) + \bar{\nu}(p_6) + b(p_7)) + q(p_9)$	N
560	$Z(e-(p_3) + e+(p_4)) + t(p_5) + q(p_6)$	N
561	$Z(e-(p_3) + e+(p_4)) + \bar{t}(p_5) + q(p_6)$	N
562	$Z(e-(p_3) + e+(p_4)) + t(p_5) + q(p_6) + f(p_7)$	L
563	$Z(e-(p_3) + e+(p_4)) + \bar{t}(p_5) + q(p_6) + f(p_7)$	L
564	$Z(e-(p_3) + e+(p_4)) + t(--> \nu(p_5) + e^+(p_6) + b(p_7)) + q(p_8)$	N
566	$Z(e-(p_3) + e+(p_4)) + t(--> \nu(p_5) + e^+(p_6) + b(p_7)) + q(p_8) + f(p_9)$	L
567	$Z(e-(p_3) + e+(p_4)) + \bar{t}(--> e^-(p_5) + \bar{\nu}(p_6) + \bar{b}(p_7)) + q(p_8)$	N
569	$Z(e-(p_3) + e+(p_4)) + \bar{t}(--> e^-(p_5) + \bar{\nu}(p_6) + \bar{b}(p_7)) + q(p_8) + f(p_9)$	L
640	$t(p_3) + \bar{t}(p_4) + H(p_5)$	L
641	$t(\to \nu(p_3) + e^+(p_4) + b(p_5)) + \bar{t}(\to \bar{\nu}(p_7) + e^-(p_8) + \bar{b}(p_6)) + H(b(p_9) + \bar{b}(p_{10}))$	L
644	$t(\to \nu(p_3) + e^+(p_4) + b(p_5)) + \bar{t}(\to \bar{q}(p_7) + q(p_8) + \bar{b}(p_6)) + H(b(p_9) + \bar{b}(p_{10}))$	L
647	$t(\to q(p_3) + \bar{q}(p_4) + b(p_5)) + \bar{t}(\to \bar{\nu}(p_7) + e^-(p_8) + \bar{b}(p_6)) + H(b(p_9) + \bar{b}(p_{10}))$	L
651	$t(\to \nu(p_3) + e^+(p_4) + b(p_5)) + \bar{t}(\to \bar{\nu}(p_7) + e^-(p_8) + \bar{b}(p_6)) + H(\gamma(p_9) + \gamma(p_{10}))$	L
654	$t(\to \nu(p_3) + e^+(p_4) + b(p_5)) + \bar{t}(\to \bar{q}(p_7) + q(p_8) + \bar{b}(p_6)) + H(\gamma(p_9) + \gamma(p_{10}))$	L
657	$t(\to q(p_3) + \bar{q}(p_4) + b(p_5)) + \bar{t}(\to \bar{\nu}(p_7) + e^-(p_8) + \bar{b}(p_6)) + H(\gamma(p_9) + \gamma(p_{10}))$	L
661	$t(\to \nu(p_3) + e^+(p_4) + b(p_5)) + \bar{t}(\to \bar{\nu}(p_7) + e^-(p_8) + \bar{b}(p_6)) + H(W^+(p_9, p_{10})W^-(p_{11}, p_{12})$	L
664	$t(\to \nu(p_3) + e^+(p_4) + b(p_5)) + \bar{t}(\to \bar{q}(p_7) + q(p_8) + \bar{b}(p_6)) + H(W^+(p_9, p_{10})W^-(p_{11}, p_{12})$	L
667	$t(\to q(p_3) + \bar{q}(p_4) + b(p_5)) + \bar{t} \to (\bar{\nu}(p_7) + e^-(p_8) + \bar{b}(p_6)) + H(W^+(p_9, p_{10})W^-(p_{11}, p_{12})$	L
800	$V \to (\chi(p_3) + \bar{\chi}(p_4)) + f(p_5)$ [Vector Mediator]	N
801	$A \to (\chi(p_3) + \bar{\chi}(p_4)) + f(p_5)$ [Axial Vector Mediator]	N
802	$S \to (\chi(p_3) + \bar{\chi}(p_4)) + f(p_5)$ [Scalar Mediator]	N
803	$PS \to (\chi(p_3) + \bar{\chi}(p_4)) + f(p_5)$ [Pseudo Scalar Mediator]	N
804	$GG \to (\chi(p_3) + \bar{\chi}(p_4)) + f(p_5)$ [Gluonic DM operator]	N
805	$S --(\chi(p_3) + \bar{\chi}(p_4)) + f(p_5)$ [Scalar Mediator, mt loops]	N
820	$V \to (\chi(p_3) + \bar{\chi}(p_4)) + \gamma(p_5)$ [Vector Mediator]	N
821	$A \to (\chi(p_3) + \bar{\chi}(p_4)) + \gamma(p_5)$[Axial Vector Mediator]	N
822	$S \to (\chi(p_3) + \bar{\chi}(p_4)) + \gamma(p_5)$[Scalar Mediator]	N
823	$PS \to (\chi(p_3) + \bar{\chi}(p_4)) + \gamma(p_5)$[Pseudo Scalar Mediator]	N
840	$V \to (\chi(p_3) + \bar{\chi}(p_4)) + f(p_5) + f(p_6)$ [Vector Mediator]	L
841	$A \to (\chi(p_3) + \bar{\chi}(p_4)) + f(p_5) + f(p_6)$ [Axial Vector Mediator]	L
842	$S \to (\chi(p_3) + \bar{\chi}(p_4)) + f(p_5) + f(p_6)$ [Scalar Mediator]	L
843	$PS \to (\chi(p_3) + \bar{\chi}(p_4)) + f(p_5)7\mp f(p_6)$ [Pseudo Scalar Mediator]	L
844	$GG \to (\chi(p_3) + \bar{\chi}(p_4)) + f(p_5) + f(p_6)$ [Gluonic DM operator]	L
845	$V \to (\chi(p_3) + \bar{\chi}(p_4)) + \gamma(p_5) + f(p_6)$ [Vector Mediator]	L
846	$A \to (\chi(p_3) + \bar{\chi}(p_4)) + \gamma(p_5) + f(p_6)$ [Axial Vector Mediator]	L
847	$S \to (\chi(p_3) + \bar{\chi}(p_4)) + \gamma(p_5) + f(p_6)$ [Scalar Mediator]	L
848	$PS \to (\chi(p_3) + \bar{\chi}(p_4)) + \gamma(p_5) + f(p_6)$ [Pseudo Scalar Mediator]	L
902	Check of Volume of 2 particle phase space	

Generator MCFM: Exercise

<u>Process 300:</u>

pp->Zγ->eeγ

Obtain:

σ -?

σ(PDF= cteq6 l, mstw8nl)-?

σ(photonptmin=20GeV,30GeV,40GeV,50GeV,60GeV)-?

LHAPDF package

Package gives a simple possibility to use additional (modern) PDF-sets (parton distribution functions) by generators.
Site: https://lhapdf.hepforge.org/

Use version 5 for installation! 6-th needs more packages to be installed: the Boost C++ utility library (http://www.boost.org) and the yaml-cpp (http://code.google.com/p/yaml-cpp/) parser.

Installation:
*tar -xvzf lhapdf-**v.r.p**.tar.gz*
*cd lhapdf-**v.r.p***
./configure --prefix=/path/to/directory
make
make install

However, folder in prefix should not be the same as folder where program has been unpacked (lhapdf-***v.r.p***)

(~5 minutes)

Generator Herwig/Herwig++

Multipurpose generator Herwig

Hadron Emission Reactions With Interfering Gluons
Herwig++ is a general-purpose event generator for the simulation of high-energy lepton-lepton, lepton-hadron and hadron-hadron collisions with special emphasis on the accurate simulation of QCD radiation.

History of this generator starts from full-functional Fortran version.
Fortran Herwig is still under support (last released version is *6.521 –March 2013*), however it is not being developed (only bug fixes etc).
From site: *“Version 6.5 has always been foreseen as the final Fortran version of HERWIG. The recent sub-version releases are tidying up the last few loose ends.”*
http://www.hep.phy.cam.ac.uk/theory/webber/Herwig/

For serious event generation, HERWIG has been replaced by Herwig++, version 2.4 of which is already available, providing a complete simulation of most types of collider events. This new program is on C++ and it has all features and advanages of old HERWIG.
Currently version 2.7.0 is available on site:
https://herwig.hepforge.org/

Let’s start from Herwig++!

Generator Herwig++

Herwig++ already includes several features more advanced than the last FORTRAN version. Herwig++ provides a full simulation of high energy collisions with the following special features:

• Initial- and final-state QCD jet evolution taking account of soft gluon interference via angular ordering;
• A detailed treatment of the suppression of QCD radiation from massive particles, the *deadcone* effect;
• The simulation of BSM physics including correlations between the production and decay of the BSM particles together with the ability to add new models by simply encoding the Feynman rules;
• An eikonal model for multiple partonic scatterings to describe the underlying event;
• A cluster model of the hadronization of jets based on non-perturbative gluon splitting;
• A sophisticated model of hadron and tau decays using matrix elements to give the momenta of the decay products for many modes and including a detailed treatment of off-shell effects and spin correlations.

Generator Herwig++: installation

Herwig++ installation includes installation Herwig++ itself and special tool ThePEG (**Toolkit for High Energy Physics Event Generation**). Herwig++ is based on this tool.

All additional packages can be installed before ThePEG and Herwig configuration. If you need install them afterwards – you should re-configure ThePEG and Herwig++.

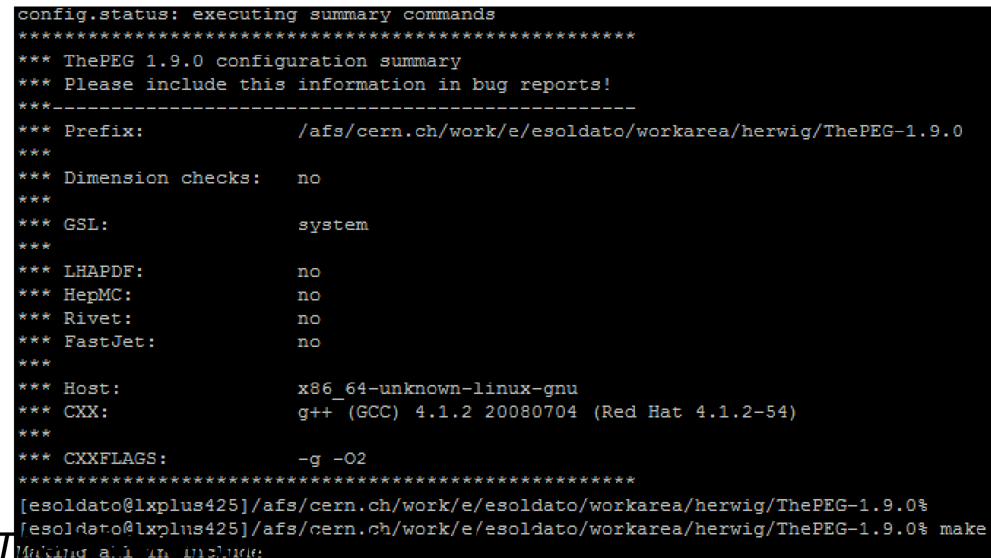
Installation:

ThePEG

Download ThePEG, then

tar xjvf ThePEG-.tar.bz2*
*cd ThePEG**
./configure --prefix=/path/where/T
make
make check
make install

(~20 minutes)



Generator Herwig++: installation II

Herwig++
Download Herwig++, then

tar xjvf Herwig++-.tar.bz2*
*cd Herwig++**
./configure --prefix=/path/where/Herwig++/should/be/installed --with-thepeg=/path/where/ThePEG/is/installed
make
make check
make install

(~1 h 20 m)

```
config.status: executing libtool commands
config.status: executing summary commands
*****************************************************
*** Herwig++ 2.7.0 configuration summary
*** Please include this information in bug reports!
***--------------------------------------------------
*** Prefix:             /afs/cern.ch/work/e/esoldato/workarea/herwig/Herwig++-2.7.0
***
*** BSM models:         yes
*** Dipole shower:      yes  * warning: LHAPDF disabled *
***
*** Herwig debug mode:  no
***
*** ThePEG:             /afs/cern.ch/work/e/esoldato/workarea/herwig/ThePEG-1.9.0
*** ThePEG headers:     /afs/cern.ch/work/e/esoldato/workarea/herwig/ThePEG-1.9.0/include
***
*** GSL:                system
*** boost:              system
*** Fastjet:            no
***
*** Host:               x86_64-unknown-linux-gnu
*** CC:                 gcc (GCC) 4.1.2 20080704 (Red Hat 4.1.2-54)
*** CXX:                g++ (GCC) 4.1.2 20080704 (Red Hat 4.1.2-54)
*** FC:                 GNU Fortran (GCC) 4.1.2 20080704 (Red Hat 4.1.2-54)
***
*** CXXFLAGS:           -O3
*****************************************************
[esoldato@lxplus425]/afs/cern.ch/work/e/esoldato/workarea/herwig/Herwig++-2.7.0% make
Making all in include
```

Generator Herwig++: first run

Create a temporary directory to try out Herwig++, and copy all files ending in .in from the Herwig installation:

mkdir herwigtest

cd herwigtest

cp $HERWIGPATH/share/Herwig++/.in ./*

Run the following command to set up a generator file. This command will be explained further on.

Herwig++ read LEP.in

This should have generated a file LEP.run. It contains a full example setup of a LEP event generator. To run it, try

Herwig++ run LEP.run -N50 -d1

The flag -N specifies the number of events to generate. The flag -d1 enables a more detailed output in LEP.log.

After the run, there should be two new files: LEP.out, which contains an overall cross section, and LEP.log, containing the detailed record of the requested 50 events. The log format is described in detail in EventRecordFormat.

Re-running the above command will give the *same* 50 events again unless the seed of the random number generator is changed by specifying the flag -seed NNNNN.

If you want to modify generator parameters, you'll need to modify LEP.in and regenerate LEP.run.

Generator Herwig++: Matrix elements

In Herwig++ the library of matrix elements for QCD and electroweak processes is relatively small, certainly with respect to the large range of processes available in its FORTRAN predecessor. Indeed, the library of Standard Model processes is largely intended to provide a core of important processes with which to test the program.

For e+e− colliders only four hard processes are included:

• Quark-antiquark production, via interfering photon and Z0 bosons, is implemented in the **MEee2gZ2qq** class. No approximation is made regarding the masses of the particles. This process is essential for us to validate the program using QCD analyses of LEP data.
• Dilepton pair production, via interfering photon and Z0 bosons, is implemented in the **MEee2gZ2ll** class. No approximation is made regarding the masses of the particles4. This
process is used to check the implementation of spin correlations in τ decays.
• The Bjorken process, Z0h0 production, which is implemented in the **MEee2ZH** class. This process is included as it is very similar to the production of Z0h0 and W±h0 in hadronhadron collisions and uses the same base class for most of the calculation.
• The vector-boson fusion (VBF) processes, e+e− → e+e−h0 and e+e− → νe¯νeh0, are implemented in the **MEee2HiggsVBF** class.

Generator Herwig++: Matrix elements II

A much wider range of matrix elements is included in the standalone code for the simulation of events in hadron colliders:

• Difermion production via s-channel electroweak gauge bosons. The matrix elements for the production of fermion-antifermion pairs through W± bosons, or interfering photons and Z0 bosons, are implemented in the **MEqq2W2ff** and **MEqq2gZ2ff** classes respectively. Only s-channel electroweak gauge boson diagrams are included for the hadronic modes.

• The production of a Z0 or W± boson in association with a hard jet is simulated using the **MEPP2ZJet** or **MEPP2WJet** class respectively. The decay products of the bosons are included in the 2 → 3 matrix element and the option of including the photon for Z0 production is supported.

• The 2 → 2 QCD scattering processes are implemented in the **MEQCD2to2** class. Currently all the particles are treated as massless in these processes.

• The matrix element for the production of a heavy quark-antiquark pair (top or bottom quark pairs), is coded in the **MEPP2QQ** class. No approximations are made regarding the masses of the outgoing q¯q pair.

• The **MEPP2GammaGamma** class implements the matrix element for the production of prompt photon pairs. In addition to the tree-level q¯q → γγ process the loop-mediated gg → γγ process is included.

• Direct photon production in association with a jet is simulated using the **MEPP2GammaJet** class. As with the QCD 2 → 2 process all of the particles are treated as massless in these processes.

• The production of an s-channel Higgs boson via both gg → h0 and q¯q → h0 is simulated using the **MEPP2Higgs** class.

Etc…

Also some NLO ME are available.

Generator Herwig++: Exercise

Use *LHC.in*, **MEPP2QQ** matrix element and generate top-antitop pairs.

pp->$t\bar{t}$

σ-?

σ(PDF=MRST,CTEQ6L)-?

Make root output and plot θ(top,antitop)