LUND UNIVERSITY

MC4LHC workshop
CERN
17 July 2006

Status and Plans for PYTHIA 8

Torbjörn Sjöstrand

CERN/PH and

Department of Theoretical Physics, Lund University

PYTHIA 6 status

PYTHIA 6 still being actively developed and maintained:

- multiple interactions and underlying event, with
- transverse-momentum-ordered showers
- SUSY interfaces (SLHA) and simulation
- regular bug fixes and minor improvements
- moving to CEDAR HepForge (code management, bugtracking)

Currently PYTHIA 6.403:

- 73,600 lines of code (including comments/blanks)
- 580 page PYTHIA 6.4 Physics and Manual
  T. Sjöstrand, S. Mrenna and P. Skands,
  JHEP**05** (2006) 026 [hep-ph/0603175]
- available on `http://www.thep.lu.se/∼torbjorn/Pythia.html`
- together with sample main programs, old code, etc.

. . . but

- only add, never subtract

⇒ has become bloated and unmanageable

- is in Fortran 77, so not understood by young people

PYTHIA 8: A fresh start

Problem: PYTHIA 7 stalled, no other manpower

Solution?: take a sabbatical and work “full-time”!

(⇒ baseline model, S. Mrenna & P. Skands join later ?)

Tentative schedule:

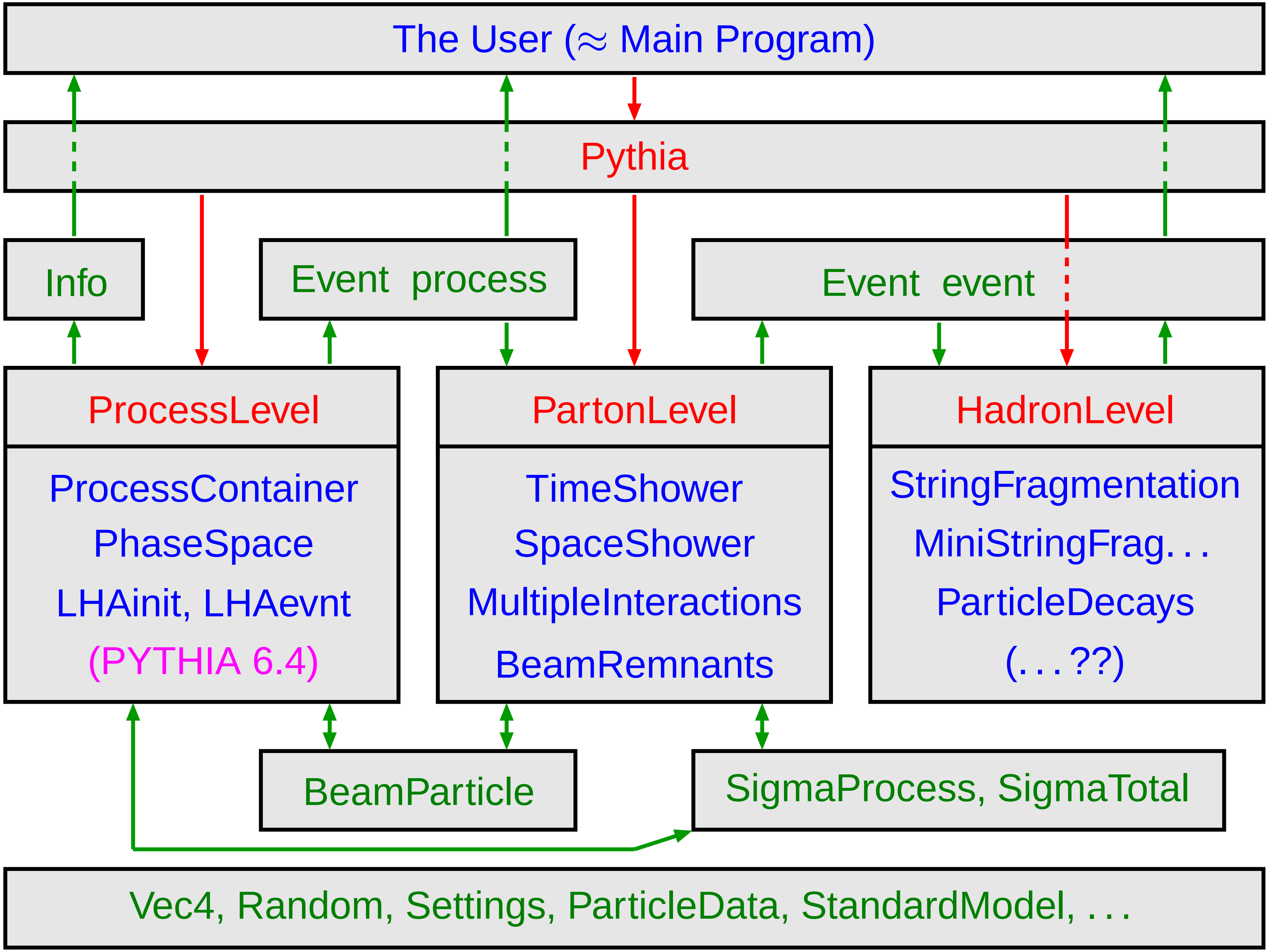
time	date	processes	final states
0 =	1 Sept. 2004	—	—
1 =	1 Sept. 2005	LHA-style input	incomplete draft
2 =	1 Sept. 2006	a few processes	complete, buggy(?)
3 =	1 Sept. 2007	more processes	stable, debugged

. . . but don’t forget Murphy’s law

Objectives:

- clean up, keep the most recent models
- core program completely standalone, but
- Les Houches Accord style input central
- interfaces to other libraries foreseen

Current PYTHIA 8 structure



The User (≈ Main Program)

Pythia

Info

Event process

Event event

ProcessLevel

ProcessContainer

PhaseSpace

LHAinit, LHAevnt

(PYTHIA 6.4)

PartonLevel

TimeShower

SpaceShower

MultipleInteractions

BeamRemnants

HadronLevel

StringFragmentation

MiniStringFrag. . .

ParticleDecays

(. . . ??)

BeamParticle

SigmaProcess, SigmaTotal

Vec4, Random, Settings, ParticleData, StandardModel, . . .

Event generation structure

1) Initialization step

- select process(es) to study
- modify physics parameters
- set kinematics constraints
- modify generator settings
- initialize generator
- book histograms

```
#include "Pythia.h"
using namespace Pythia8;
Pythia pythia;
pythia.readString("command");
pythia.readFile("filename");
pythia.init(idBeamA,idBeamB,eCM);
```

2) Generation loop

- generate one event at a time
- analyze it (or store for later)
- add results to histograms
- print a few events

```
pythia.next();
pythia.process.list();
pythia.event.list();
int id = pythia.event[i].id();
```

3) Finishing step

- print deduced cross-sections
- print/save histograms etc.

```
pythia.statistics();
pythia.settings.listChanged();
```

Settings and Particle Data

Can read in settings and particle data changes by

- `pythia.readString("command")`
- `pythia.readFile("filename")` with one `command` per line in file

Settings come in three kinds

- Flags: on/off switches, `bool`
- Modes: enumerated options, `int`
- Parameters: continuum of values, `double`

and `command` is of form `location:name = value`, e.g.

`PartonLevel:ISR = off` no initial-state radiation

`InFlux:nQuark = 4` do not consider b pdf's

`TimeShower:pTmin = 1.0` cut off final-state radiation at 1 GeV

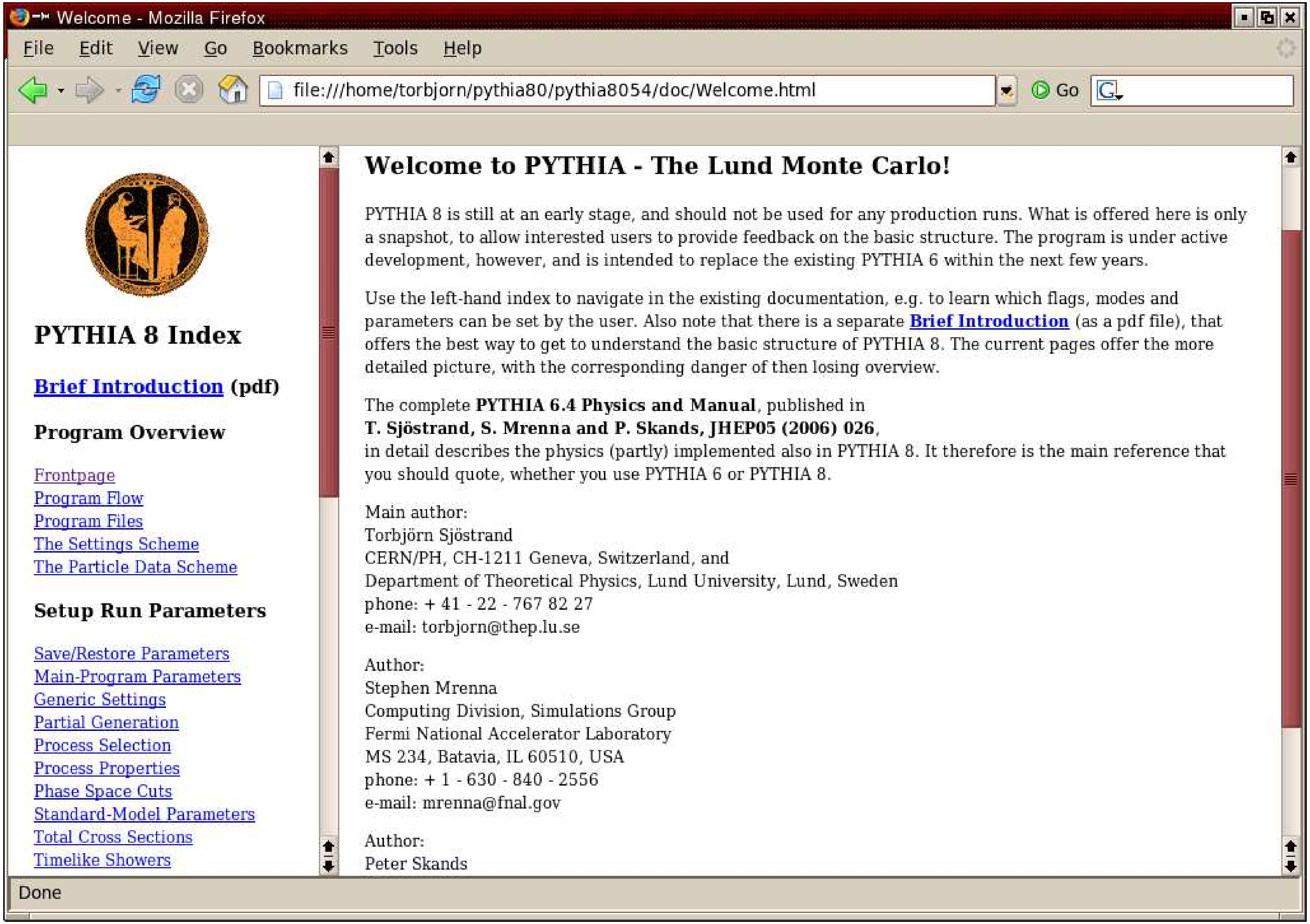
To access **particle data**, instead `command` should be of form

`id:property = value` or `id:channel:property = value`, e.g.

`3122:mayDecay = no` do not allow Λ^0 to decay (`no = off = false = 0`)

`215:3:products = 211 111 111` to let $a_2^+ \to \pi^+\pi^0\pi^0$

Online manual ⟹ GUI??



Welcome - Mozilla Firefox
File Edit View Go Bookmarks Tools Help
file:///home/torbjorn/pythia80/pythia8054/doc/Welcome.html
Go
Done

PYTHIA 8 Index

Brief Introduction (pdf)

Program Overview

Frontpage
Program Flow
Program Files
The Settings Scheme
The Particle Data Scheme

Setup Run Parameters

Save/Restore Parameters
Main-Program Parameters
Generic Settings
Partial Generation
Process Selection
Process Properties
Phase Space Cuts
Standard-Model Parameters
Total Cross Sections
Timelike Showers

Welcome to PYTHIA - The Lund Monte Carlo!

PYTHIA 8 is still at an early stage, and should not be used for any production runs. What is offered here is only a snapshot, to allow interested users to provide feedback on the basic structure. The program is under active development, however, and is intended to replace the existing PYTHIA 6 within the next few years.

Use the left-hand index to navigate in the existing documentation, e.g. to learn which flags, modes and parameters can be set by the user. Also note that there is a separate **Brief Introduction** (as a pdf file), that offers the best way to get to understand the basic structure of PYTHIA 8. The current pages offer the more detailed picture, with the corresponding danger of then losing overview.

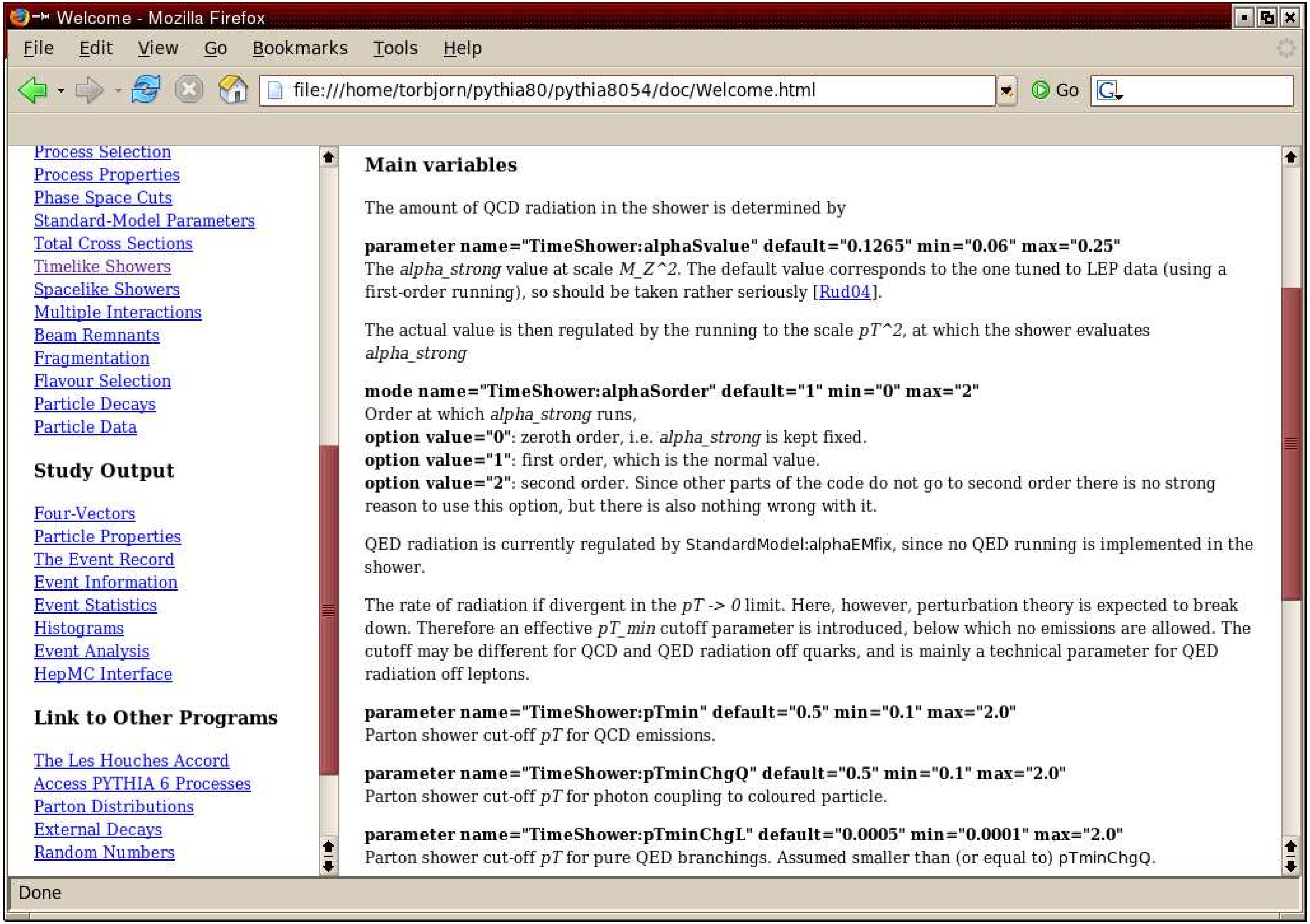
The complete **PYTHIA 6.4 Physics and Manual**, published in
T. Sjöstrand, S. Mrenna and P. Skands, JHEP05 (2006) 026,
in detail describes the physics (partly) implemented also in PYTHIA 8. It therefore is the main reference that you should quote, whether you use PYTHIA 6 or PYTHIA 8.

Main author:
Torbjörn Sjöstrand
CERN/PH, CH-1211 Geneva, Switzerland, and
Department of Theoretical Physics, Lund University, Lund, Sweden
phone: + 41 - 22 - 767 82 27
e-mail: torbjorn@thep.lu.se

Author:
Stephen Mrenna
Computing Division, Simulations Group
Fermi National Accelerator Laboratory
MS 234, Batavia, IL 60510, USA
phone: + 1 - 630 - 840 - 2556
e-mail: mrenna@fnal.gov

Author:
Peter Skands

Example: timelike parton showers



Welcome - Mozilla Firefox
File Edit View Go Bookmarks Tools Help
file:///home/torbjorn/pythia80/pythia8054/doc/Welcome.html
Go
Done

Process Selection
Process Properties
Phase Space Cuts
Standard-Model Parameters
Total Cross Sections
Timelike Showers
Spacelike Showers
Multiple Interactions
Beam Remnants
Fragmentation
Flavour Selection
Particle Decays
Particle Data

Study Output

Four-Vectors
Particle Properties
The Event Record
Event Information
Event Statistics
Histograms
Event Analysis
HepMC Interface

Link to Other Programs

The Les Houches Accord
Access PYTHIA 6 Processes
Parton Distributions
External Decays
Random Numbers

Main variables

The amount of QCD radiation in the shower is determined by

parameter name="TimeShower:alphaSvalue" default="0.1265" min="0.06" max="0.25"
The *alpha_strong* value at scale *M_Z^2*. The default value corresponds to the one tuned to LEP data (using a first-order running), so should be taken rather seriously [Rud04].

The actual value is then regulated by the running to the scale *pT^2*, at which the shower evaluates *alpha_strong*

mode name="TimeShower:alphaSorder" default="1" min="0" max="2"
Order at which *alpha_strong* runs,
option value="0": zeroth order, i.e. *alpha_strong* is kept fixed.
option value="1": first order, which is the normal value.
option value="2": second order. Since other parts of the code do not go to second order there is no strong reason to use this option, but there is also nothing wrong with it.

QED radiation is currently regulated by StandardModel:alphaEMfix, since no QED running is implemented in the shower.

The rate of radiation if divergent in the *pT -> 0* limit. Here, however, perturbation theory is expected to break down. Therefore an effective *pT_min* cutoff parameter is introduced, below which no emissions are allowed. The cutoff may be different for QCD and QED radiation off quarks, and is mainly a technical parameter for QED radiation off leptons.

parameter name="TimeShower:pTmin" default="0.5" min="0.1" max="2.0"
Parton shower cut-off *pT* for QCD emissions.

parameter name="TimeShower:pTminChgQ" default="0.5" min="0.1" max="2.0"
Parton shower cut-off *pT* for photon coupling to coloured particle.

parameter name="TimeShower:pTminChgL" default="0.0005" min="0.0001" max="2.0"
Parton shower cut-off *pT* for pure QED branchings. Assumed smaller than (or equal to) pTminChgQ.

Hard-process generation

Currently limited selection implemented internally,
but can use Fortran PYTHIA 6 library transparently via LHA interface.
Provide settings as before, with `command` of form
`Pythia6:variable = value`, e.g. `Pythia6:msel = 6`.
`variable` can be anything recognized by `PYGIVE`,
but only ones relevant for hard process are actually used.

Can also use Les Houches Accord for any other process,
input via runtime Fortran interface or via files.

Overview processes that can be generated:

Process	internal	PYTHIA 6	Les Houches
minbias/elastic/diffractive	+	–	–
QCD jets (incl. $c\overline{c}$, $b\overline{b}$)	+	+	+
prompt photons	+	+	+
other PYTHIA 6 processes	–	+	+
anything else (within limits)	–	–	+

Trying It Out

- Download `pythia8053.tgz` from `http://www.thep.lu.se/∼torbjorn/Pythia.html`, link “`Future`”
- Unzip and expand with `tar xvfz pythia8053.tgz`
- Move to the thus created `pythia8053` directory
- Follow the `README` instructions (edit links to PYTHIA 6, HepMC)
- `make` will compile in $\sim$ 4 minutes (half for PYTHIA 6)
- The `pythia8051.pdf` file contains an introduction to the program
- Open `doc/Welcome.html` in a web browser for the full manual (in the future: GUI with xml + Javascript ?)
- The `examples` subdirectory contains 14 sample main programs (`make mainNN` and then `mainNN.exe > outfile`)

Example of a main program

```
// Test program main06: study pTZ spectrum at the Tevatron.
#include "Pythia.h"
using namespace Pythia8;
int main() {
  // Generator. Process selection. Tevatron initialization. Histogram.
  Pythia pythia;
  pythia.readString("Pythia6:msel = 11");
  pythia.readString("Pythia6:ckin(1) = 80.");
  pythia.readString("PartonLevel:MI = off");
  pythia.readString("Beams:primordialKTwidth = 2.");
  pythia.init( 2212, -2212, 1960.);
  Hist pTZ("dN/dpTZ",100,0.,100.);
  // Begin event loop. Generate event. Skip if error. List first few.
  for (int iEvent = 0; iEvent < 10000; ++iEvent) {
    if (!pythia.next()) continue;
    if (iEvent < 2) pythia.event.list();
    // Loop over particles in event. Find last Z0 copy. Fill its pT.
    int iZ = 0;
    for (int i = 0; i < pythia.event.size(); ++i)
      if (pythia.event[i].id() == 23) iZ = i;
    pTZ.fill( pythia.event[iZ].pT() );
  // End of event loop. Statistics. Histogram. Done.
  }
  pythia.statistics();
  cout << pTZ;
  return 0;
}
```

Missing topics

- Many, many processes
- More on phase space selection (resonances, $2 \rightarrow 3, \ldots$)
- Resonance widths and decays
- More external-input facilities, with improved ME/PS matching
- MI/ISR/FSR interleaving (with colour flow models)
- Hadronization: popcorn baryons, updated decay tables, Bose–Einstein
- Event analysis routines, other facilities
- ...

Notable simplifications

- Only transverse-momentum-ordered showers
- Only most recent multiple interactions scenario(s)
- Only string fragmentation

+ Complete second pass to sort out minor issues

Outlook

● C++ PYTHIA 8 is coming along ●

⋆ Roughly according to three-year plan the first year! ⋆

⋆ On hold during autumn 2005, now progressing again ⋆

⋆ $\sim$ 1 sub-subversion per working week, currently 8.054 ⋆

⋆ Put recent versions on web at irregular intervals ⋆

⋆ First (almost) production-quality release, 8.100, early 2007 (?) ⋆

⋆ Debugged and tuned by LHC startup 2008 (??) ⋆

⋆ Overtaking Fortran version usage by 2009 (???) ⋆

● Feedback is most welcome ●

⋆ Basic structure is settling down . . . ⋆

⋆ . . . but still lot of room for changes ⋆

⋆ e.g. interfacing to other programs ⋆