Integration of LCG MCDB with CMSSW

L. Dudko, SINP MSU, Moscow
on behalf of LCG MCDB group

http://mcdb.cern.ch

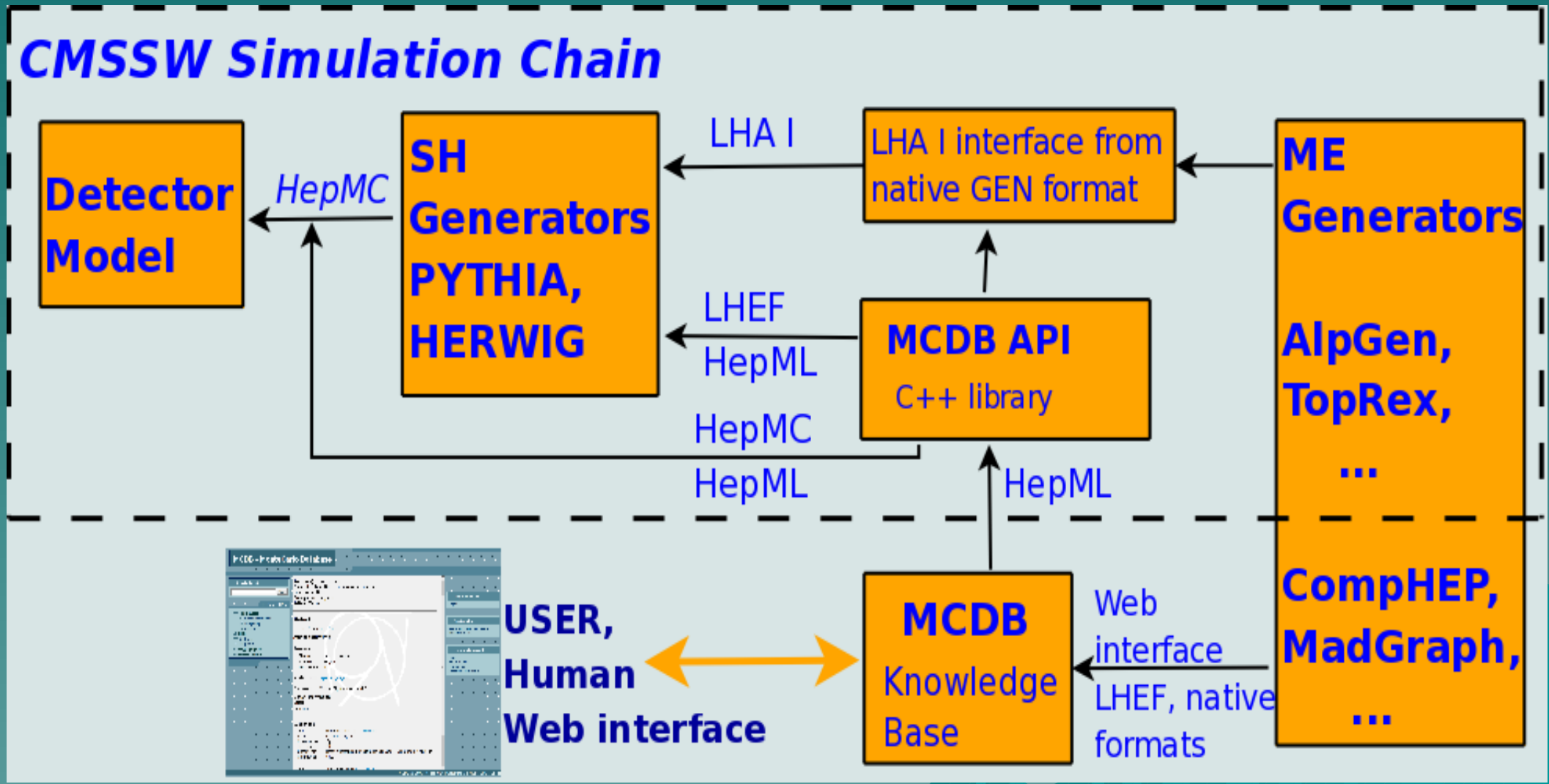
OUTLINE:

- Knowledge Base and API
- Dedicated Software
- Realization and Plans

LCG MCDB group:

S. Belov, JINR
L. Dudko, SINP MSU
W.Pokorski, CERN
A. Sherstnev, Univ. of Cambridge

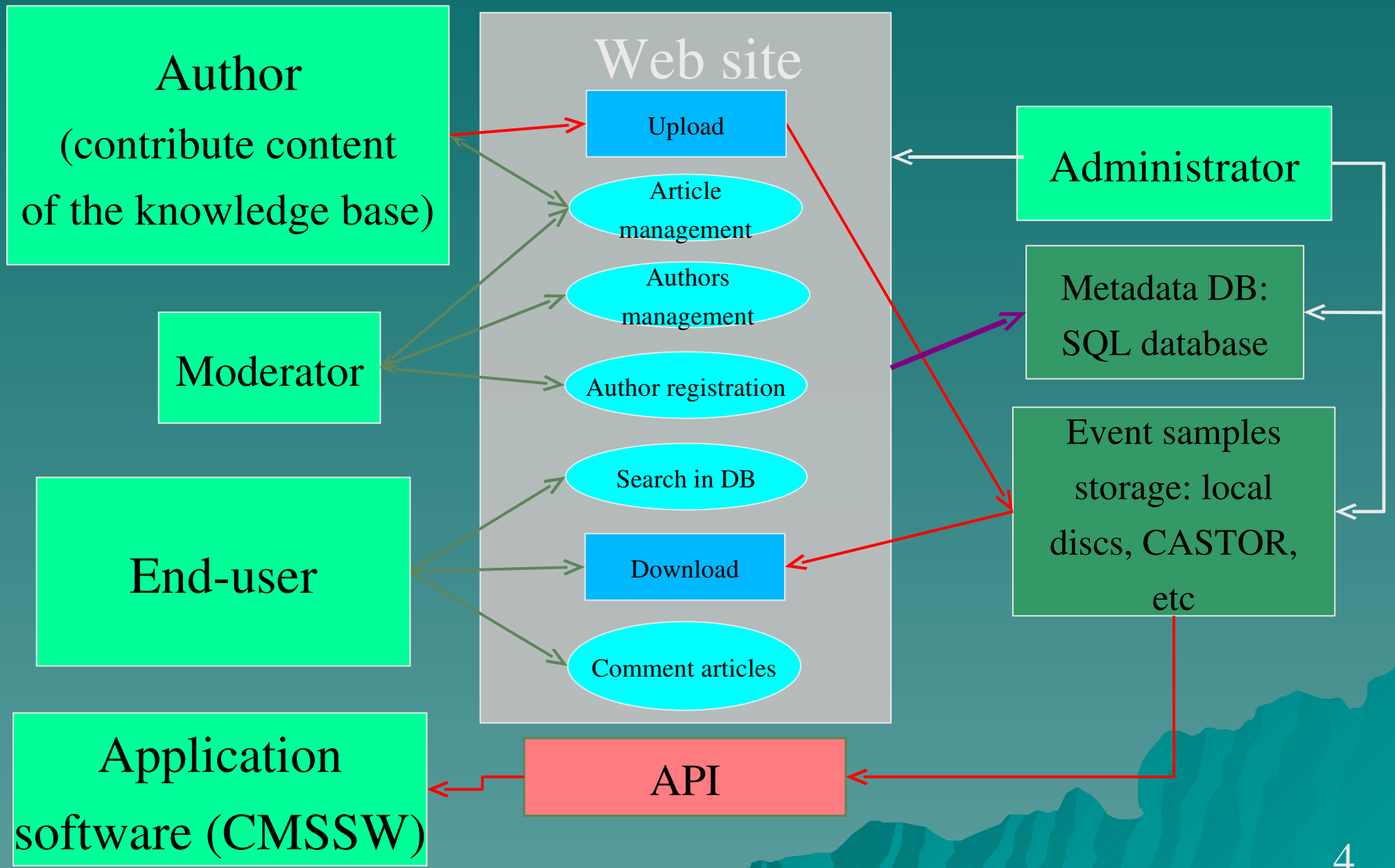
What is MCDB in the Simulation Chain



Overview of the Knowledge Base

- **The Major Tasks of LCG MCDB**
  - **Share sophisticated MC generated samples between different groups**
    - **Samples prepared by experts in MC event generation**
    - **Resource-intensive samples (Human or/and CPU resources)**
  - **Provide infrastructure to keep MC samples and sample documentation**
  - **Facilitate communication between MC experts and users in LHC collaborations**

LCG MCDB Scheme



The Major Features of LCG MCDB (I)

- Powerful WEB interface with Content Management System for the authors of event samples and end-users
- Tree graph of physics categories with articles published in MCDB to browse the database content
- Power search engine based on SQL/XML to search the content of the knowledge base
- Flexible and reliable authorization system based on CERN AFS/Kerberos logins or LCG GRID certificates
- SQL structure of event sample documentation
- BackUp of samples and SQL information

The Major Features of LCG MCDB (II)

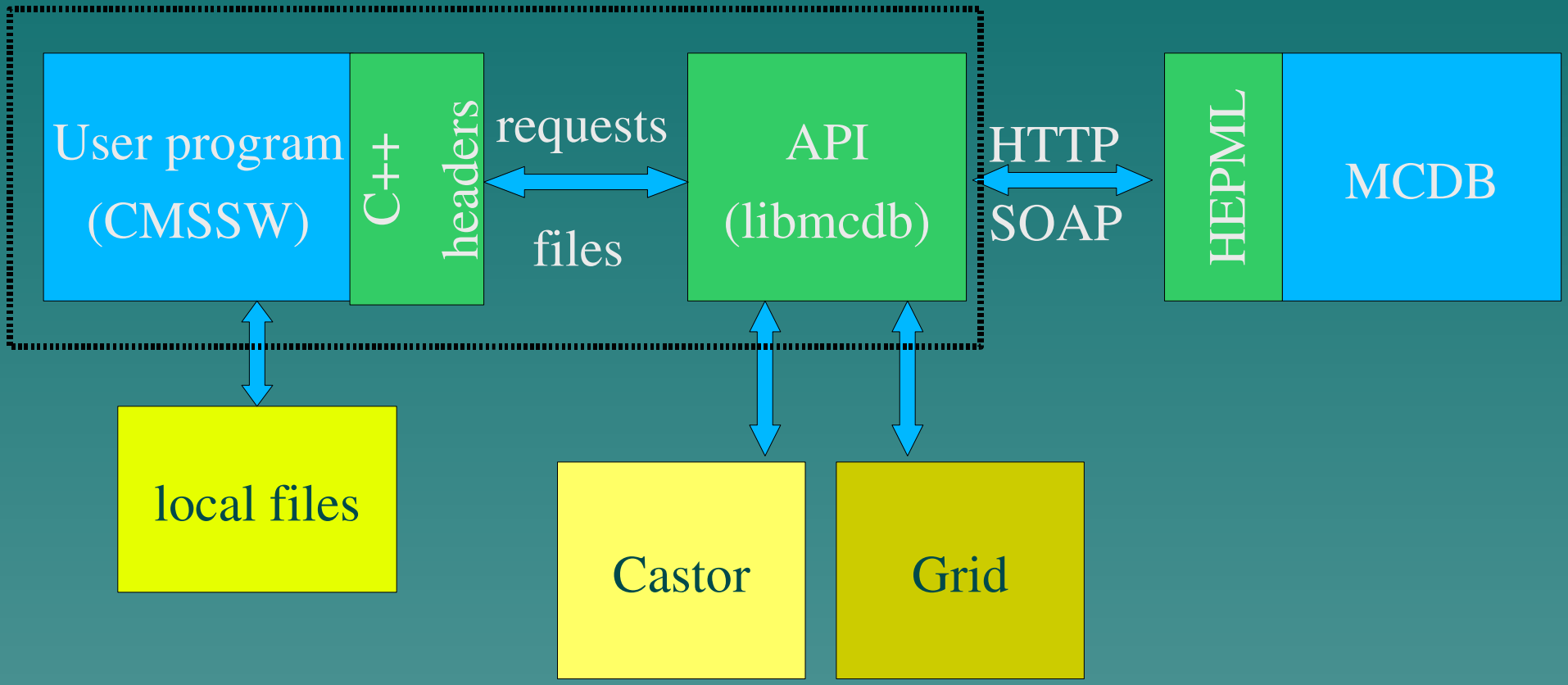
- CASTOR is the native storage for the event samples
- Direct uploading of multiple files from AFS/CASTOR/GRID (wild-card characters are possible) to LCG MCDB
- Direct downloading of files from LCG MCDB with HTTP/CASTOR/GRID (URI)
- Application Programming Interface (API) for the LHC collaborations environment software
- LHEF/HepML unification of event file format

PARAMETERS OF EVENT SAMPLE DESCRIPTION

MCDB XML Scheme inside HepML specifications

- General information
  - Title
  - Abstract
  - Authors
  - Experiment and/or Group
- Physics process
  - Initial state
  - Final state
  - QCD scale
  - Process PDF
- Event files
  - Physics process/subprocesses
  - File name
  - Events number
  - cross section and uncertainty
- Used generator
  - Name and version
  - Description
  - Home page address
- Theoretical model
  - Name
  - Description
  - Set of parameters and their values with author's descriptions
- Applied cuts

MCDB API structure



STEP I: MCDB => API

- HTTP request with ArticleID from API to MCDB server

  (e.g. http://mcdb.cern.ch/cgi-bin/xmlquery.cgi?article=116)

- HepML description and paths to samples, as an answer from MCDB to API
- API parses the HepML (XML) block and fill C++ classes with the description
- API make local copy of the event sample, download the remote sample via SEAL (RFIO, GridFTP, HTTP, ...)

STEP II: API => CMSSW

- API provides:
  - local path to file with input events
  - Sample Description in the form of C++ objects
- CMSSW process the local event file to the next level of simulation
- CMSSW passes the C++ objects with event description to the output files

MCDB API C++ Classes

http://mcdb.cern.ch/doc/API/public/mcdb.hpp

```
namespace mcdb
{
class Article;
class File;
class Author;
class Cut;
class Generator;
class Model;
class Process;
class Subprocess;
}
```

```
class Generator{
 public:
   Generator();
   ~Generator();
   string& name();
   string& name(const string&);
   string& version();
   string& version(const string&);
   string& homepage();
   string& homepage(const string&);
 private:
   string name_;
   string version_;
   string homepage_;
};

class Process{
 public:
   Process();
   ~Process();
   string& initialState();
   string& initialState(const string&);
   string& finalState();
   string& finalState(const string&);
   string& factScale();
   string& factScale(const string&);
   string& renormScale();
   string& renormScale(const string&);
   string& pdf();
   string& pdf(const string&);
 private:
```

```
class Model{
 public:
    Model();
    ~Model();
    class ModelParameter;
    string& name();
    string& name(const string&);
    string& description();
    string& description(const string&);
    vector<ModelParameter>&
parameters();
    vector<ModelParameter>&
parameters(const
vector<ModelParameter>&);
    class ModelParameter
    {
     public:
        ModelParameter();
        ~ModelParameter();
        string& name();
        string& name(const string&);
        string& value();
        string& value(const string&);
     private:
        string name_;
        string value_;
    };
 private:
    string name_;
    string description_;
    vector<ModelParameter> parameters_;
```

How it works in reality

Stable versions of CMSSW

Subroutine in MadGraph and CompHEP interfaces:

CMSSW_1_6_5/src/GeneratorInterface/MadGraphInterface/src/MCDBInterface.cc

testMCDBInterface.cfg:

```
untracked bool   getInputFromMCDB = true
int32            MCDBArticleID = 113
string MadGraphInputFile = "rfio://castor/cern.ch/user/d/doudko/run_1_unweighted_events.lhe"
```

Only download the sample locally, does not parse sample description

Development version of CMSSW

Library with MCDB API code in:

CMSSW/src/GeneratorInterface/MCDBInterface/

part of CMSSW config file is almost the same

Inside any other interface add:

```
mcdb::MCDB mcdb;
mcdb::Article a = mcdb.getArticle(MCDBArticleID);
```

Automatically process and document in CMSSW the sample from MCDB

Fully Automatic Way to Document the sample.

New Developments from HepML.

- ME Generator link libHepML
- ME Generator call libHepML functions to fill the description of the events
- libHepML forms HepML header with sample description inside LHEF header
  (works in C++, C, Fortran)
- libHepML provides methods to parse HepML description and create C++ objects (MCDB API) or describe the sample in MCDB (SOAP)
- The first realisation of LHEF/HepML is available in CompHEP 4.5

HepML in LHEF

- J. Alwall et al., A standard format for Les Houches Event Files (2006) [hep-ph/0609017]
- structure:

```
<LesHouchesEvents version="1.0">
  <header>
    <hepml>
        <!-- HepML sample description here -->
    </hepml>
  </header>
  <init> ... </init>
  <event> ... </event>
  <event> ... </event>
  ........
</LesHouchesEvents>
```

C Functions from libHepML

```
void * init_lhaef_document (void);
int set_general_description(void * doc, lhaef_general_description * sample);
int set_file_description(void * doc, lhaef_file * file);
int set_generator(void * doc, lhaef_generator * gen);
int set_model(void * doc, lhaef_model * model);
int add_model_parameter(void * doc, lhaef_parameter * par);
int set_cutset(void * doc, int cutsetnumber);
int add_cut(void * doc, lhaef_cut * cut);
int set_author_record(void * doc);
int add_author(void * doc, lhaef_author * author);
int create_process(void * doc, lhaef_process * proc);
int add_process_beam(void * doc, int i, lhaef_beam * beam);
int add_process_beam_pdf(void * doc, int i, lhaef_pdf * pdf);
int add_process_beam_pdf_alphas(void * doc, lhaef_alphas * alphas);
int add_process_alphas(void * doc, lhaef_alphas * alphas);
int add_subprocess(void * doc, lhaef_subprocess * subproc, int cutsetnumber);
char * form_hepml_document(void * doc);
```

Plans of LCG MCDB Team

- Implement MCDB API in CMSSW (automatic processing and documentation of the samples from MCDB)
- If it will be requested, implement Fully automatic chain of sample documentation
  (from ME Generator to CMSSW, based on LHEF/HepML)
- Implement items from MCDB-TODO
  (long list of necessary changes described in MCDB TWiki)
- Finish documentation on MCDB-CMSSW
- Publish MCDB software as an Open Source project in HepFourge

MCDB Software

- Stable versions of MCDB server and MCDB API are available in the download section: http://mcdb.cern.ch/distribution
- Development versions are in MCDB CVS:
  simu.cvs.cern.ch:/cvs/simu/GENSER/MCDB
  http://simu.cvs.cern.ch/cgi-bin/simu.cgi/simu/GENSER/MCDB/

Conclusion

- LCG MCDB Knowledge Base is stable
  - MC experts are kindly inviting to share the knowledge by means of LCG MCDB
- LCG MCDB API provides automatic processing of MCDB samples and documentation in the Collaborations
- Complete implementation of MCDB API in CMSSW is in progress
- LHEF/HepML libraries/utilities/interfaces could be developed in collaboration with other groups

BACKUP SLIDES

History: CMS MCDB

- http://cmsdoc.cern.ch/cms/generators/mcdb/
- Operates in CMS during the last four years, widely uses by the Higgs group
- Only parton level files; AFS storage; Only phonetic search; No SQL

Documentation

- **Main Web Page** http://mcdb.cern.ch
  - Description of the project
  - Users and Authors HOW-TOs
  - Developers documentation
- **Wiki** https://twiki.cern.ch/twiki/bin/view/LCG/LCGMCDB
- [hep-ph/0404241] LCG MCDB proposal
- [hep-ph/0604120] LCG MCDB report (p.200-204)
- [hep-ph/0703287] LCG MCDB description
- Core software supported by LCG Software Project Infrastructure (MySQL; CASTOR; CGI; Perl; Apache)
- Mailing lists - USERS: lcg-mcdb-users@cern.ch
  Developers: project-lcg-mcdb@cern.ch

LCG MCDB Software

- LCG MCDB team has developed new Content Management System (CMS):
  - It allows to Store and document large files
  - It has Flexible authorization and authors management system (Kerberos, GRID)
  - Powerful search engine
  - Template system to enter documentation
  - Operates with MySQL
  - Different storage systems are possible
  - It is Open Source CMS (available from CVS)

External Interfaces

- API for the collaboration software to access samples and their description automatically (see S.Belov talk)
  - HepML output from MCDB
  - C++ classes and libraries to parse HepML and create corresponding C++ objects
- API to upload HepML/LHEF event sample automatically (in our plans)
  - Parse HepML header of LHEF sample
  - Create corresponding documentation in LCG MCDB automatically

LHEF/HepML Library

- Library to parse native event file format of most popular ME generators and convert it to LHEF/HepML format
  - Provides possibility to convert old events and their description to unified format by means of standalone utilities based on this library
  - Provides standard API for the authors of MC generators to keep events and their documentation in LHEF/HepML form during event generation (call of external subroutines inside the code)

LCG MCDB HepML Summary

- Concentrate efforts on passing the information from ME to SH generators as a part of LHEF HepML header
- XML schemas as a part of HepML specifications
- API C++ library and parser of LHEF/HepML
- Library and utilities to convert native generator format to LHEF/HepML or provide API to write LHEF/HepML directly from the generators
  - Can expect external support (GENSER, CMS, ...)

Terminology

- **Article** – a document describing a set of event samples. This document is the main unit of the content of MCDB
- **Category** – a branch of articles and physics models concerned a particular type of physical processes.
- **Author** – an expert in Monte-Carlo generators. (S)he can upload new event samples to MCDB and describe them in corresponding articles
- **User** – anybody who interests in new MC samples and agree with MCDB users license

WEB Interface

- Users Area:
  - Browse physics categories and articles with complete documentation on the available samples
  - Search the MCDB with complex query
  - Download available MC samples
  - Post comments to the articles and check the previous discussions on the particular article/sample

WEB Interface

- Authors Area:
  - Upload new event sample(s)
  - Document the event sample(s) in new article with help of templates system (pre-entered information)
  - Publish new article in the Users Area
  - Edit his/her previous articles or do the article publicly inaccessible for a while
  - Answer users comments/questions to his/her articles

LCG MCDB Subsystems

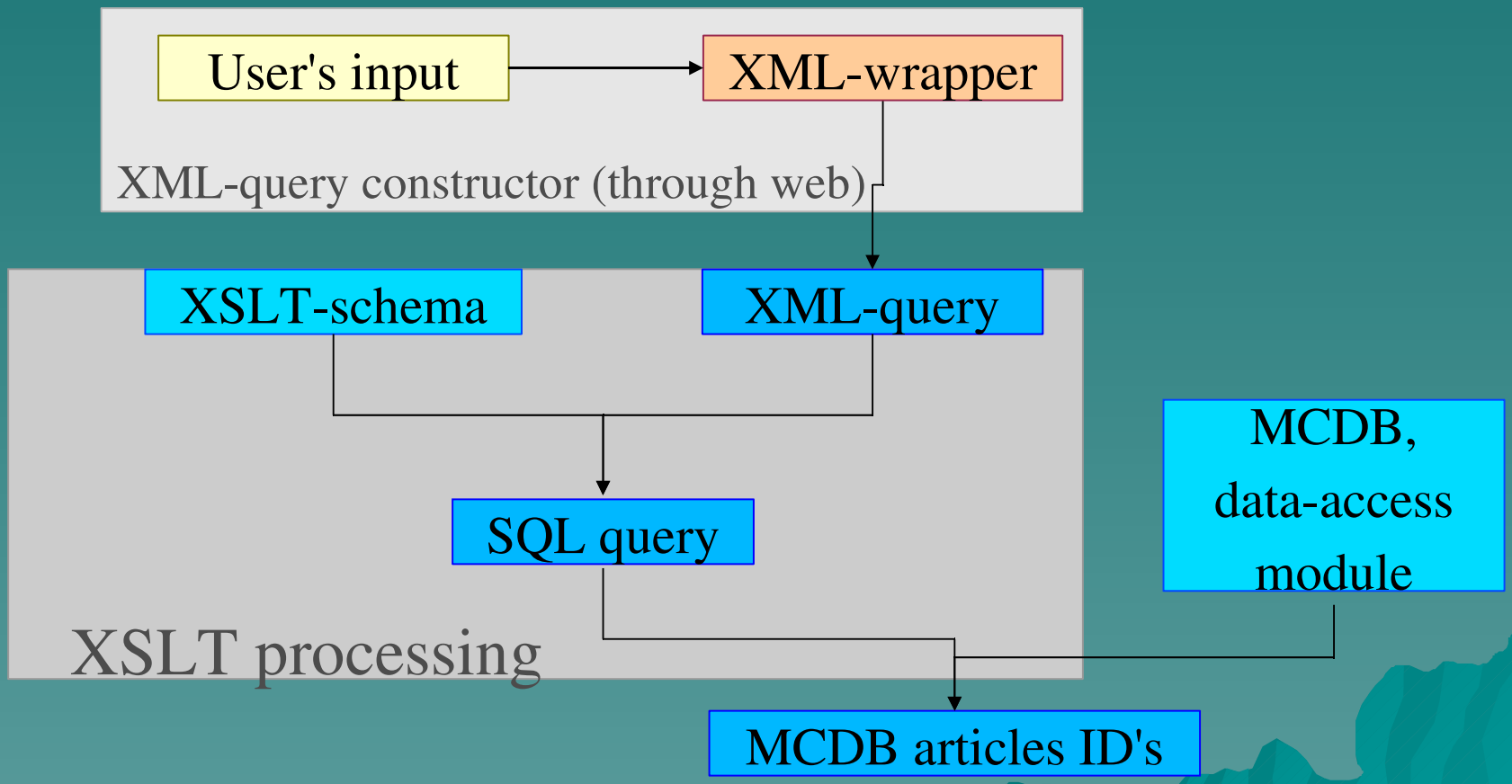
- SQL DB (MySQL)
  - Provides the possibility to keep information in a very structured way
  - Facilitate authors to keep complete set of information for the new MC sample
  - Provides the possibility for the external interfaces to search information in MCDB
  - Direct connection between sample and its description (documentation)

LCG MCDB Subsystems

- CASTOR as a native storage for the samples
  - Native storage for the LHC Collaborations
  - Direct access to files is possible (not via WEB)
  - GridFTP access
- Authors authentication system
  - CERN AFS/Kerberos authorization (SSL secure)
  - LCG GRID certificates

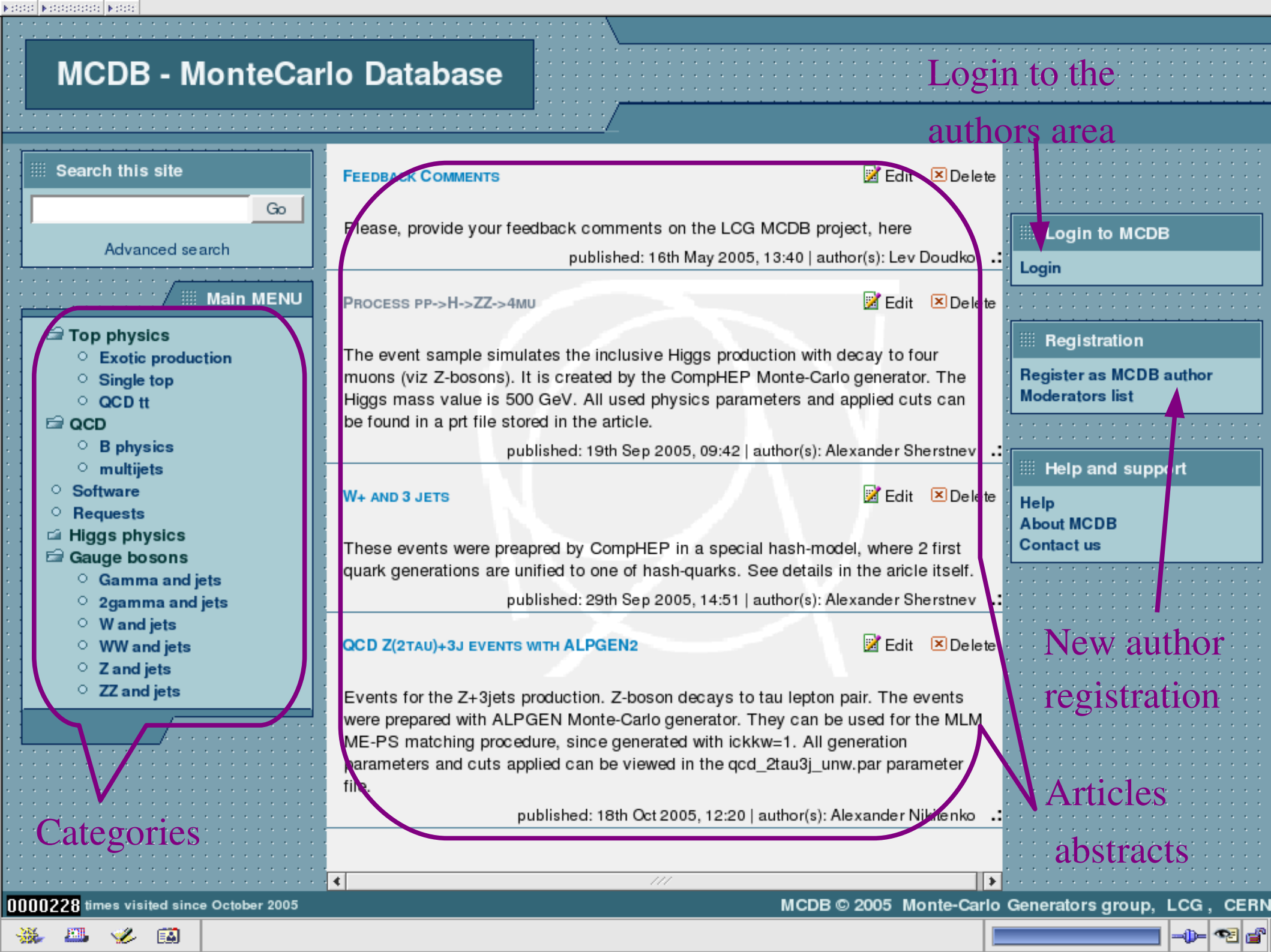
MCDB Search Engine

- *dynamic* query construction wizard (JavaScript/XML/SQL)
- Search by many possible criteria with complicated relations between DB -objects

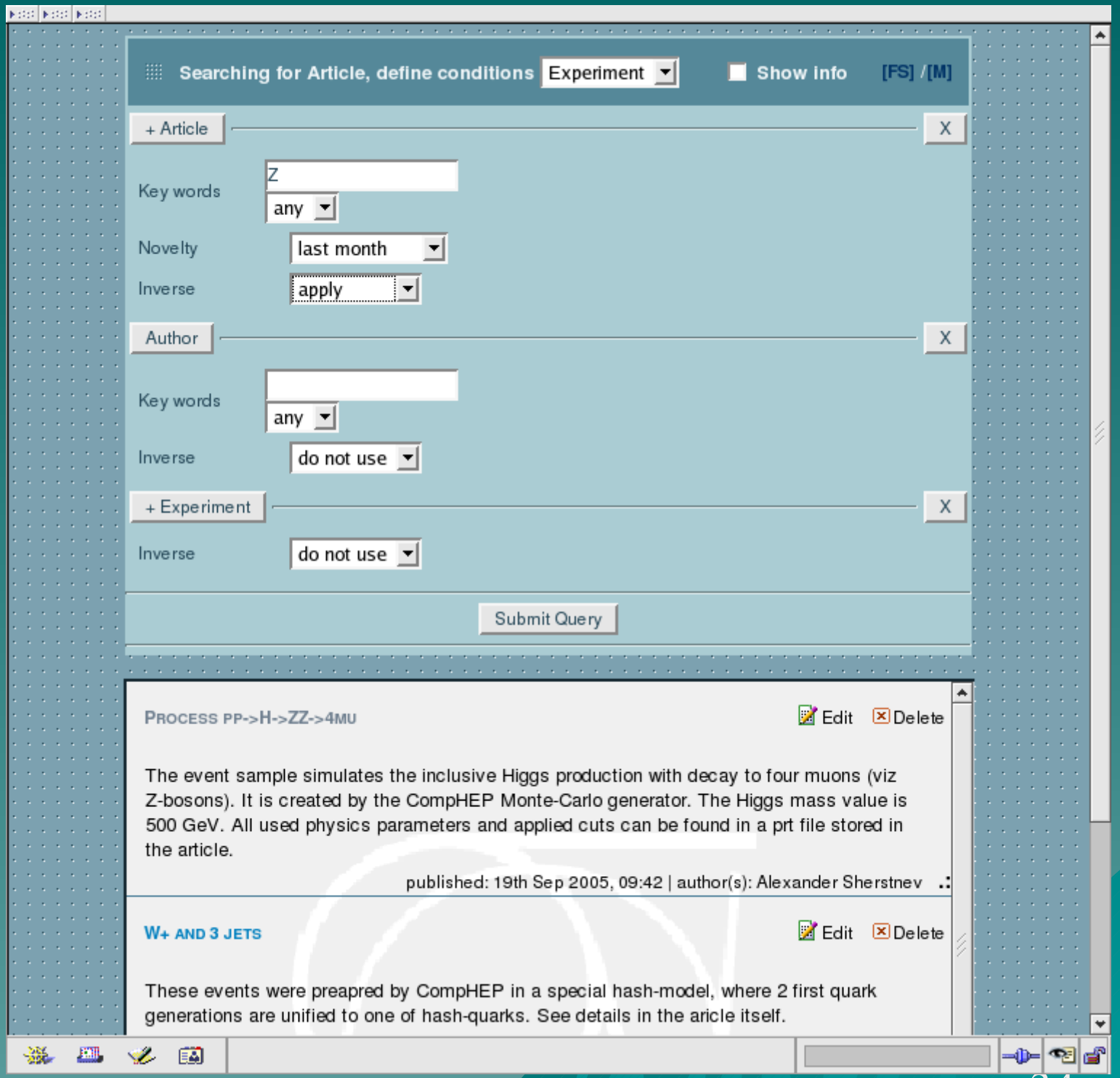


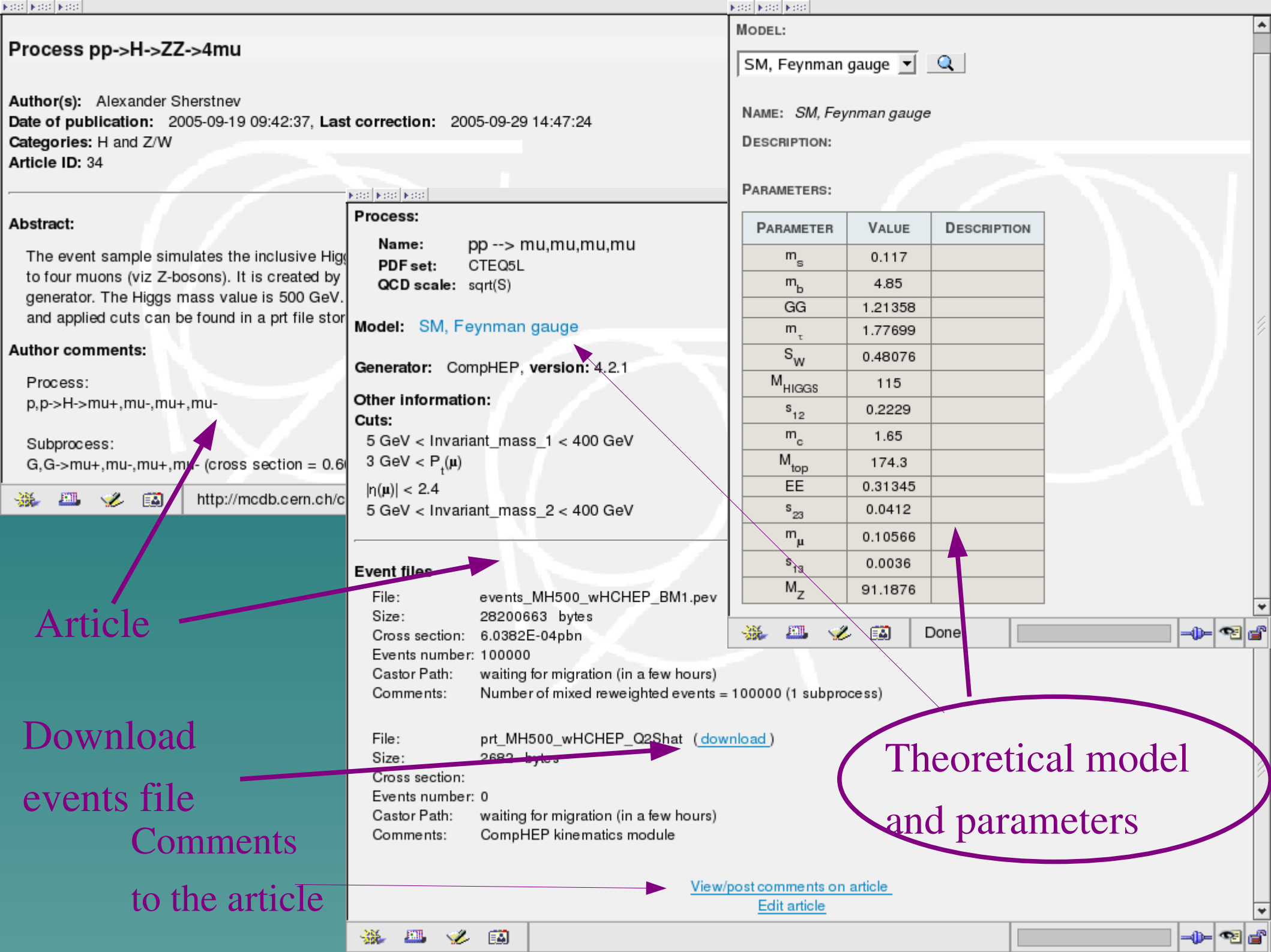
LCG MCDB Subsystems

- XML format of article (part of HepML)
- API to collaboration software
  - Set of C/C++/FORTRAN routines in collaboration environment for the direct access to the MCDB samples during the MC production. Based on HepML and direct URL to files
- HepML, unified XML format of MC events Main stiff point of present development.
  - Do possible an automatic interface to read, document MC samples and use it in the standard way in the collaboration software

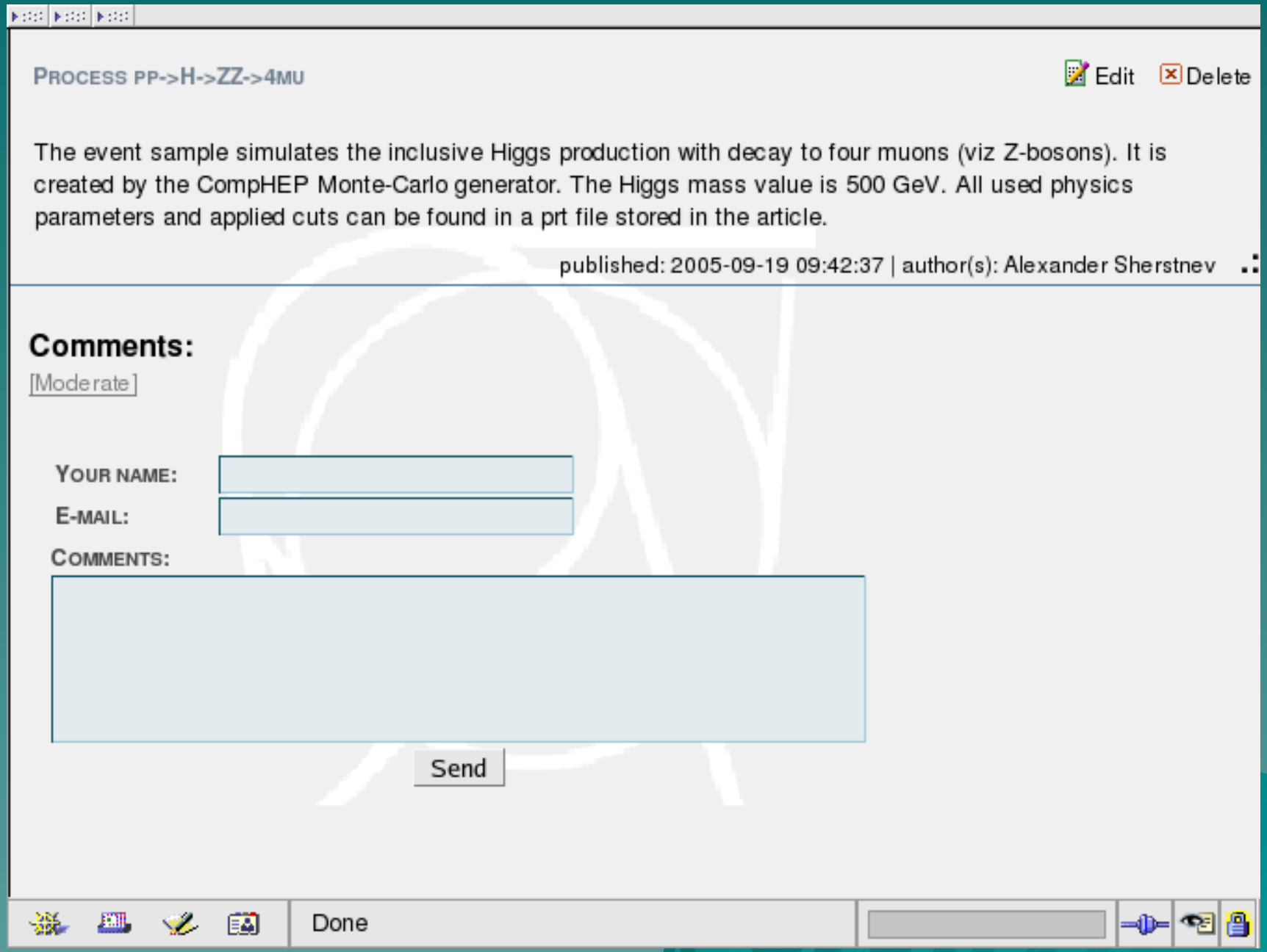

MCDB - MonteCarlo Database
Login to the authors area
Search this site
Go
Advanced search
Main MENU
Top physics
Exotic production
Single top
QCD tt
QCD
B physics
multijets
Software
Requests
Higgs physics
Gauge bosons
Gamma and jets
2gamma and jets
W and jets
WW and jets
Z and jets
ZZ and jets
Categories
FEEDBACK COMMENTS
Edit
Delete
Please, provide your feedback comments on the LCG MCDB project, here
published: 16th May 2005, 13:40 | author(s): Lev Doudko
PROCESS PP->H->ZZ->4MU
Edit
Delete
The event sample simulates the inclusive Higgs production with decay to four muons (viz Z-bosons). It is created by the CompHEP Monte-Carlo generator. The Higgs mass value is 500 GeV. All used physics parameters and applied cuts can be found in a prt file stored in the article.
published: 19th Sep 2005, 09:42 | author(s): Alexander Sherstnev
W+ AND 3 JETS
Edit
Delete
These events were preapred by CompHEP in a special hash-model, where 2 first quark generations are unified to one of hash-quarks. See details in the aricle itself.
published: 29th Sep 2005, 14:51 | author(s): Alexander Sherstnev
QCD Z(2TAU)+3J EVENTS WITH ALPGEN2
Edit
Delete
Events for the Z+3jets production. Z-boson decays to tau lepton pair. The events were prepared with ALPGEN Monte-Carlo generator. They can be used for the MLM ME-PS matching procedure, since generated with ickkw=1. All generation parameters and cuts applied can be viewed in the qcd_2tau3j_unw.par parameter file.
published: 18th Oct 2005, 12:20 | author(s): Alexander Nikitenko
Articles abstracts
Login to MCDB
Login
Registration
Register as MCDB author
Moderators list
Help and support
Help
About MCDB
Contact us
New author registration
0000228 times visited since October 2005
MCDB © 2005 Monte-Carlo Generators group, LCG , CERN

Advanced Search Query




Process pp->H->ZZ->4mu
Author(s): Alexander Sherstnev
Date of publication: 2005-09-19 09:42:37, Last correction: 2005-09-29 14:47:24
Categories: H and Z/W
Article ID: 34
Abstract:
The event sample simulates the inclusive Hig
to four muons (viz Z-bosons). It is created by
generator. The Higgs mass value is 500 GeV.
and applied cuts can be found in a prt file stor
Author comments:
Process:
p,p->H->mu+,mu-,mu+,mu-
Subprocess:
G,G->mu+,mu-,mu+,mu- (cross section = 0.6
http://mcdb.cern.ch/c
Process:
Name: pp --> mu,mu,mu,mu
PDF set: CTEQ5L
QCD scale: sqrt(S)
Model: SM, Feynman gauge
Generator: CompHEP, version: 4.2.1
Other information:
Cuts:
5 GeV < Invariant_mass_1 < 400 GeV
3 GeV < P_t(μ)
|η(μ)| < 2.4
5 GeV < Invariant_mass_2 < 400 GeV
Event files
File: events_MH500_wHCHEP_BM1.pev
Size: 28200663 bytes
Cross section: 6.0382E-04pbn
Events number: 100000
Castor Path: waiting for migration (in a few hours)
Comments: Number of mixed reweighted events = 100000 (1 subprocess)
File: prt_MH500_wHCHEP_Q2Shat (download)
Size: 2682 bytes
Cross section:
Events number: 0
Castor Path: waiting for migration (in a few hours)
Comments: CompHEP kinematics module
View/post comments on article
Edit article
MODEL:
SM, Feynman gauge
NAME: SM, Feynman gauge
DESCRIPTION:
PARAMETERS:
PARAMETER VALUE DESCRIPTION
m_s 0.117
m_b 4.85
GG 1.21358
m_τ 1.77699
S_W 0.48076
M_HIGGS 115
s_12 0.2229
m_c 1.65
M_top 174.3
EE 0.31345
s_23 0.0412
m_μ 0.10566
s_13 0.0036
M_Z 91.1876
Done
Article
Download
events file
Comments
to the article
Theoretical model
and parameters

Users Comments Interface



Documentation

MCDB authors HOW-TO

▲ Register as MCDB author

You can register as a new author following the link "Register as MCDB autho
right side menu on the main page of LCG MCDB.

- Fill out the registration form. All fields are obligatory (and at least one of "
  AFS login" or "DN from LCG certificate" should be specified).
- Submit the form and wait for confirmation e-mail (in a few work days).

▲ Change registration details

Help on LCG Monte-Carlo Data Base

Full HOW-TO for MCDB authors

Brief HOW-TO for new authors of LCG MCDB

If you want to share your new MC sample with other groups, please do the following:

- Register as a new author at the "Register New Author" at the right side menu of the main LCG MCDB web page, then wait for the confirmation e-mail
- Login to the LCG MCDB authors area at the "Login" link at the right side menu
- Choose "Create New Article" at the new right side menu
- Fill the fields in the documentation templates, which will appear (title, generator, theoretical model, cuts, ...)
- Upload your event files in the "Event Files" slice
- Click "Preview/Save" slice and check the box "Publish"

Now, your new article and samples are
Notes: 1. You need valid CERN AFS log

Main Page | Directories | File List | File Members | Search for

cgi-bin

download.cgi File Reference

Go to the source code of this file.

Functions

```
use CGI qw (:standard-private_tempfiles-oldstyle_urls)
use Fcntl qw (:DEFAULT:flock)
use POSIX qw (setsid floor ceil WNOHANG)
use File::Basename qw (basename)
use MCDB::Common qw (:DEFAULT:CGI_SCRIPTS)
if ($download_start)
unless (defined $file)
unless ($cache_id)
if ($location eq 'file')
```

Variables

```
use strict
use constant BUFFER_SIZE = > 131072
use constant REFRESH_TIME = > 10
```

New Authors HOW-TO

(1) Register as a new author, wait for the confirmation e-mail

(2) Login to the LCG MCDB authors area

(3) Choose "Create New Article" in the authors menu

(4) Fill the fields in the documentation templates, which will appear (title, generator, theoretical model, cuts, ...)

(5) Upload your event files in the "Event Files" slice

(6) Click "Preview/Save" slice and check the box "Publish"

Notes:

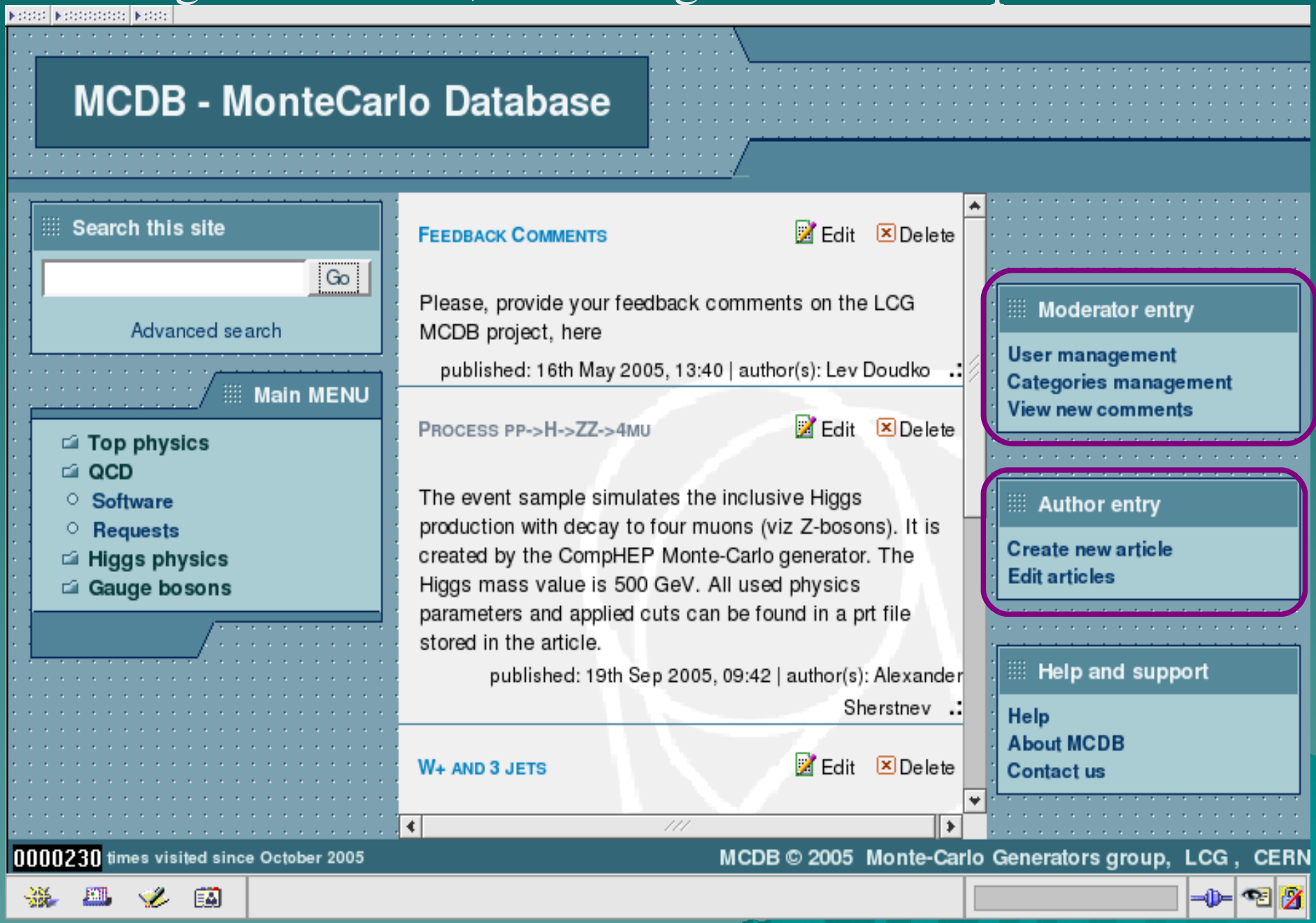
1. Author needs valid CERN AFS login or LCG digital certificate to be authorized;

2. Author can store unfinished articles and resume to correct them in any moment;

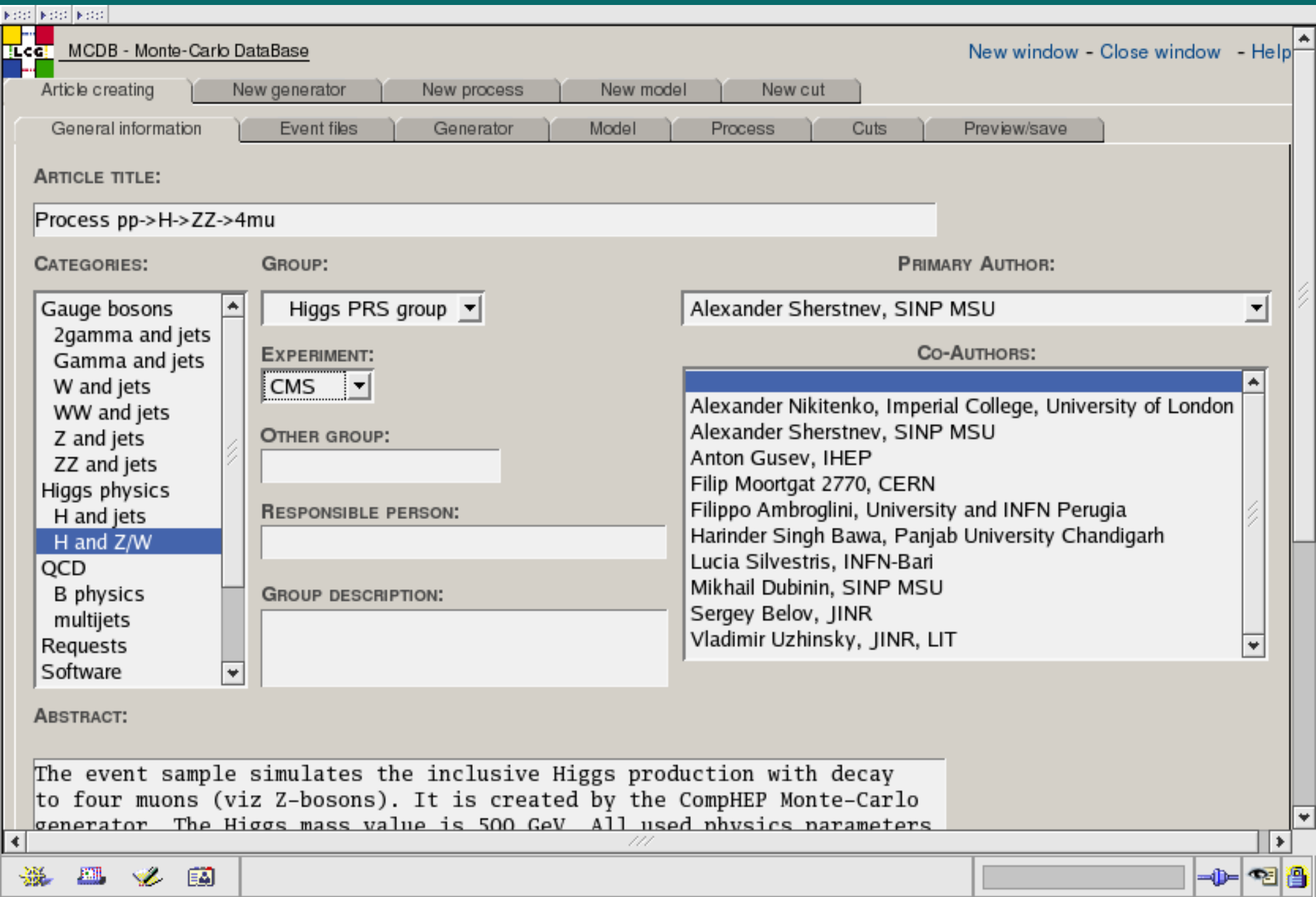
3. Author can edit articles already published on the Web or do the documents publicly inaccessible for a while.

- Form to send a request for the authorization as new LCG MCDB author.
- Necessary only if you want to upload new MC samples

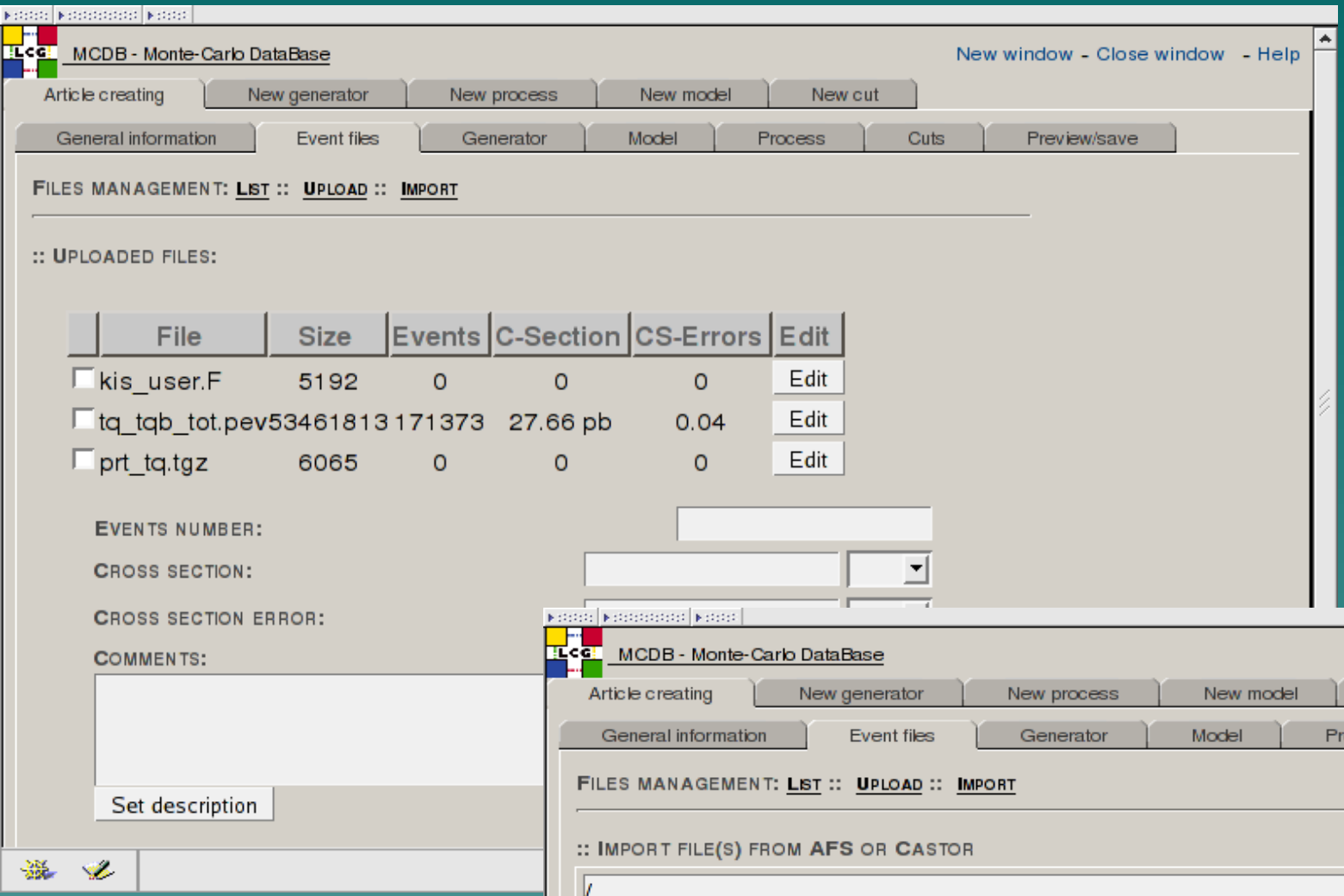
After authorized login to MCDB the additional entries will appear at the right side menu, according to the author permissions

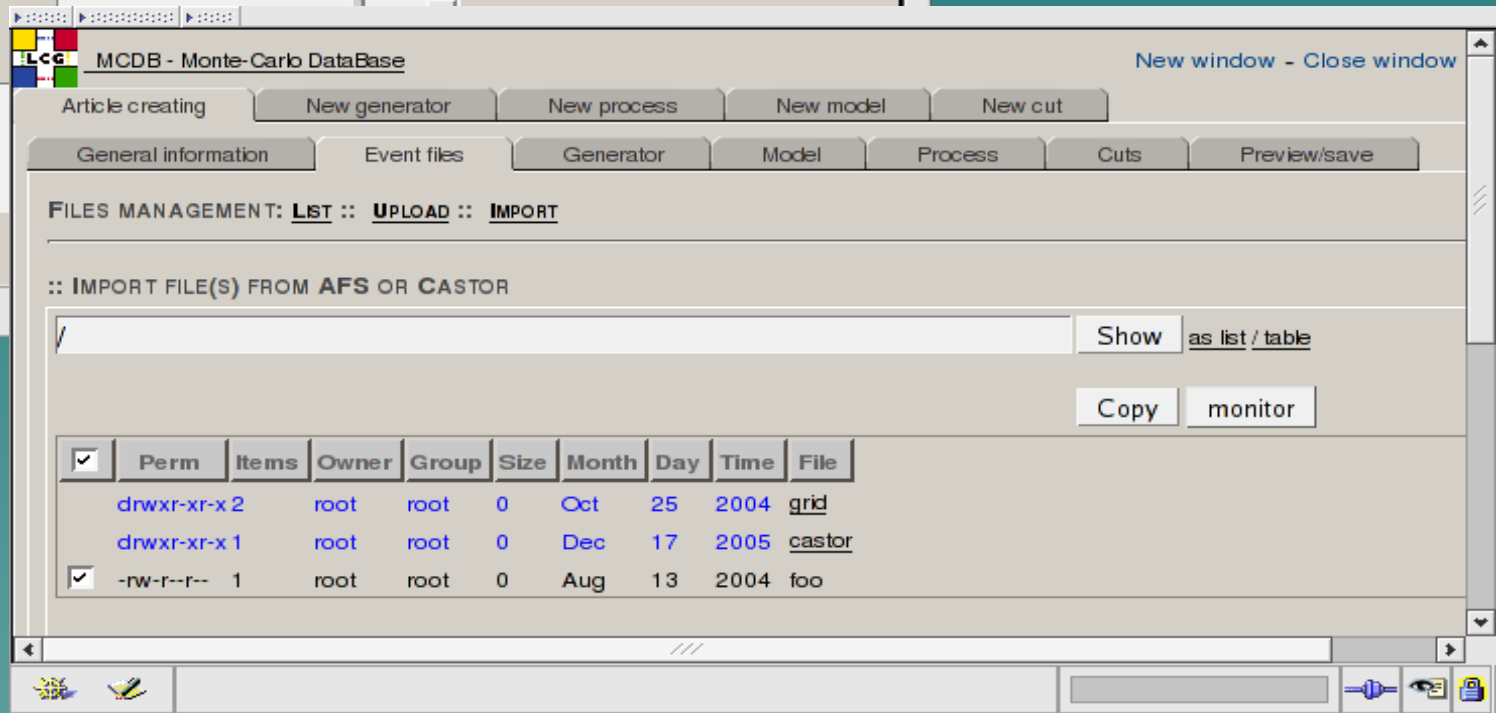


Add/Edit Article link is the gate to the article template system



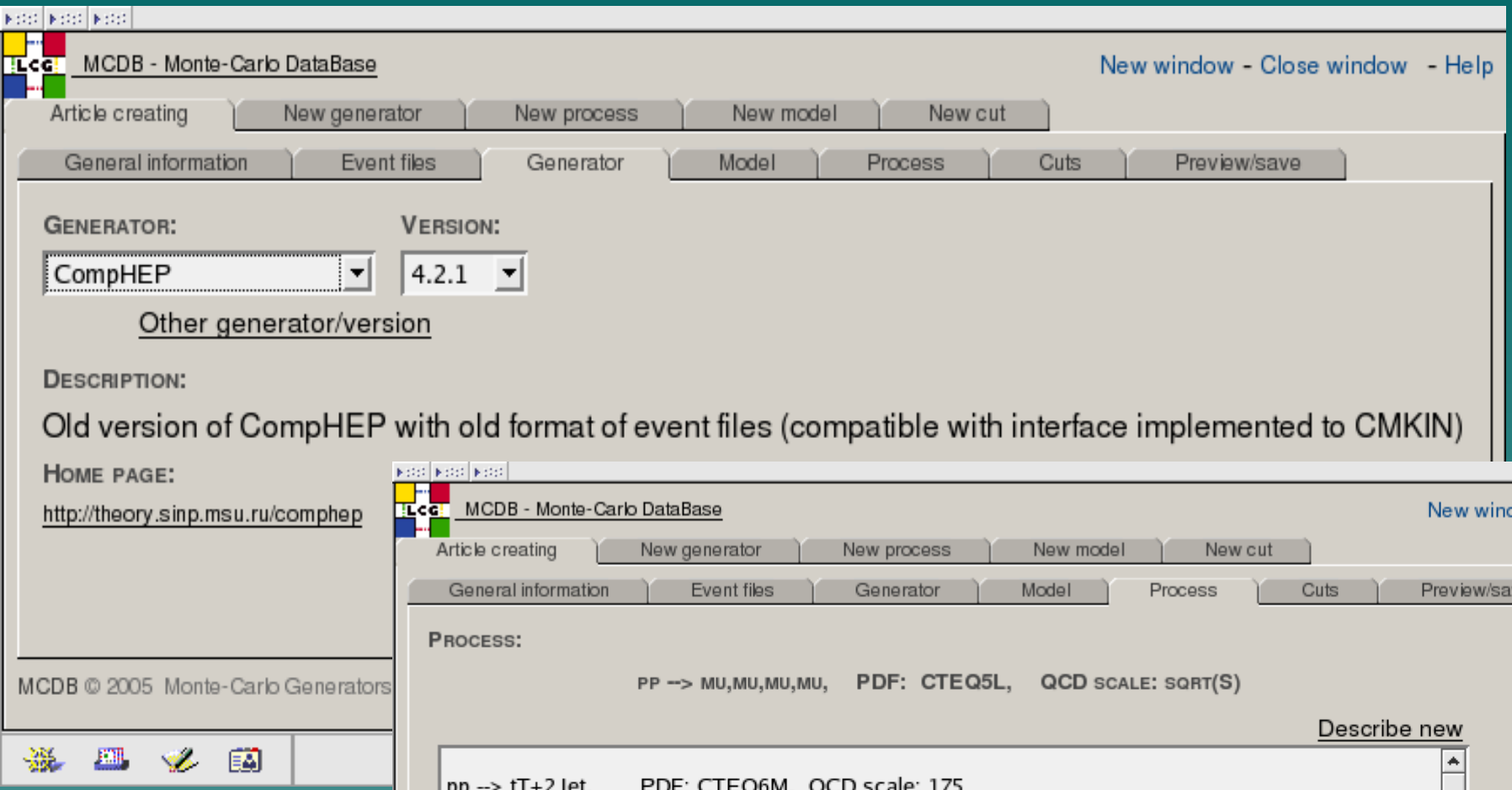
Event files slice to manage event files attached to the article

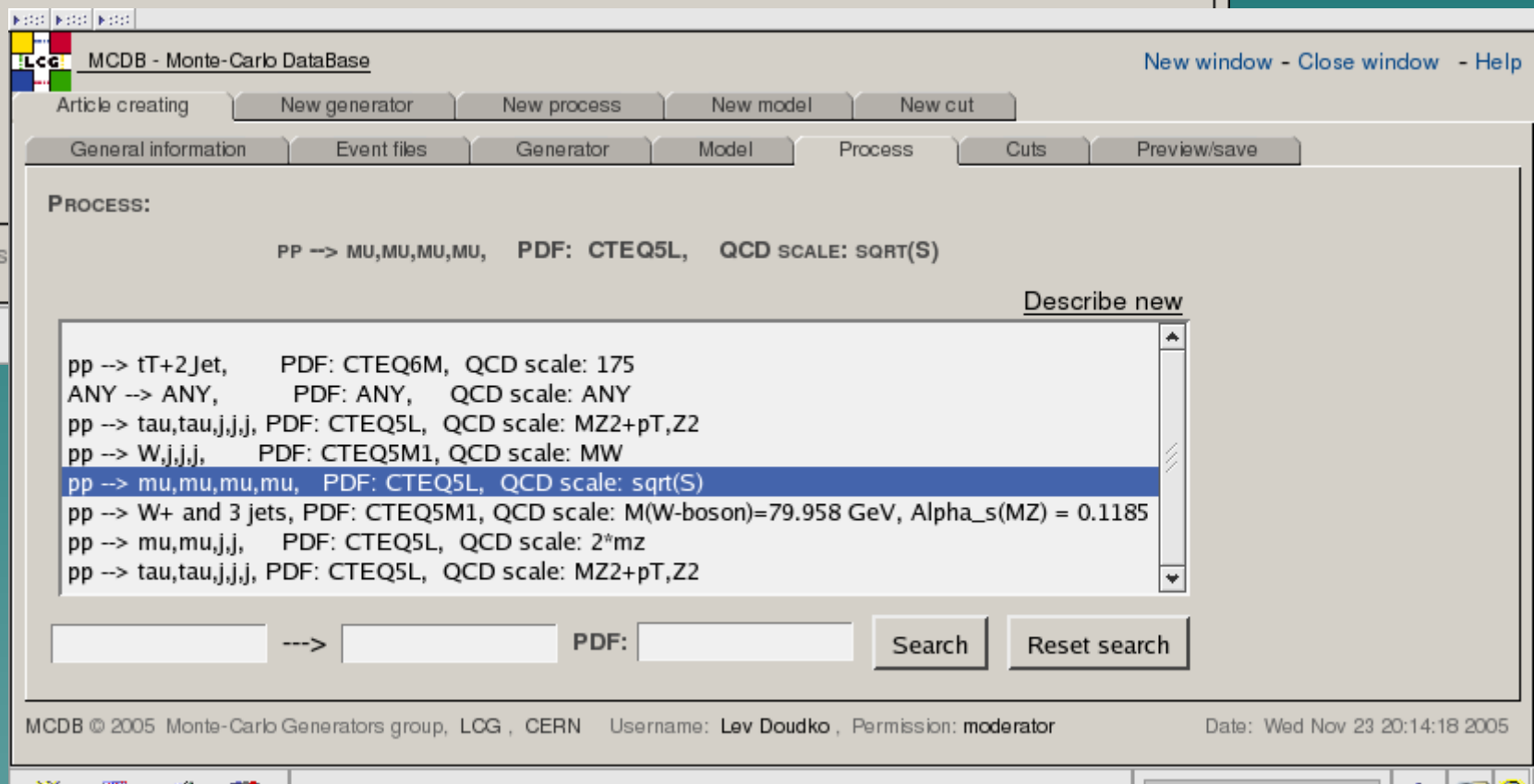




Wild-cards are possible

MC generator and Physics Process description slices





Physics model parameters and applied cuts slices