CEDAR, part II

HepData, HepML and HepForge

Andy Buckley

Institute for Particle Physics Phenomenology
Durham University, UK

HERA-LHC Workshop 2006, CERN, 2006-06-08
Outline

1. Re-introduction to CEDAR
2. HepData — the HEP reaction database
3. HepForge — a lightweight development environment for HEP
4. Summary
Re-introduction to CEDAR
I expect Jon Butterworth will have already said this, but... CEDAR is an e-Science project with several sub-projects:

- **JetWeb**: Monte Carlo generator tuning
- **HepData**: archival of published experimental data
- **HepML**: XML formats for data sets and MC config
- **HepForge**: development environment for HEP software
- **HepCode**: centralised repository of pheno code/programs
HepData — the HEP reaction database
HepData’s history

- Searchable \(\sim30\) years’ worth of scattering data, PDFs etc.
- Stored in hierarchical BDMS database: very little modern support
- Data available as text files, PAW kumacs or GIF images
- Db stores pretty much everything as a string
- Adding records requires writing a Fortran routine
- Mirrors to SPIRES, PDG info
HepData in CEDAR (1)

HepData is undergoing active development:

- Migration to relational MySQL database
- Database sanitising e.g. uniform units...
- Database improvements e.g. axis-level properties, more searchable quantities...
- Using HepML for I/O: XSLT transformations to data formats like HTML and AIDA
HepData in CEDAR (2)

And more...

- Java object model for data abstraction: interface for JetWeb
- Java/JSP-based Web front-end using Apache Tomcat
- JBoss Hibernate used to abstract the object-relational binding
- Similar system planned for XML persistency
- Grid authentication for expts. to submit data directly (modulo sanity checking)
HepData HepML

HepML is a set of XML-based data format defns. for HEP

- XML representation for generator configs and HepData data sets (and more to come?)
- (I’ll just mention the HepML data schema here)
- To be backed up with Python, Java (and C++?) interfaces
- Version for comment has been released. Please check it out: http://hepforge.cedar.ac.uk/hepml/
- Experiments will submit data to HepData: we need your comments on what HD and HepML should allow you to do. Abstracted interfaces to HepML (e.g. ROOT routine)?
<xml version="1.0" encoding="UTF-8">
<hepml xmlns="http://www.cedar.ac.uk/hepml/hepdata/0.1/">
  <data timestamp="2006-04-07 13.09.27">
    <paper irn="3326047" paperId="3552">
      <dataset datasetId="1">
        <comment>
          FITTED, BACKGROUND SUBTRACTED, PEAK OMEGA CROSS SECTION,
        </comment>
        <property name="RE(Q=MU)" value="E+ E- --&gt; MU+ MU-" />
        <property name="RE(Q=HAD)" value="E+ E- --&gt; PI+ PI0 PI-" />
        <xaxis header="SQRT(S) IN GEV" xaxisId="1">
          <bins>
            <bin pointId="1" value="0.7726" />
            <bin pointId="2" value="0.7776" />
            ...
          </bins>
        </xaxis>
      </dataset>
    </paper>
  </data>
</hepml>
HepML fragment (2)

...  
<yaxis header="SIG(Q=HAD) IN MUB" yaxisId="1">  
  <points>  
    <point pointId="1" value="0.269" />  
    <point pointId="2" value="0.74" />  
    ...  
    <point pointId="7" value="0.065" />  
  </points>  
  <axiserror norm="percent" source="sys" plus="11.0" minus="11.0"  
    description="FROM NUCLEAR ABSORPTION CORRECTION" />  
  ...  
  <pointerrors norm="abs" source="stat">  
    <pointerror plus="0.096" minus="0.096" pointId="1" />  
    <pointerror plus="0.192" minus="0.192" pointId="2" />  
    ...  
    <pointerror plus="0.036" minus="0.036" pointId="7" />  
  </pointerrors>  
</yaxis>
HepData behind the scenes

Diagram:
- HepData database
- Persistency system
- Query system
- Object model
- Web interface
- HepML
- JetWeb

Andy Buckley
CEDAR, part II
Durham University
HepData demos (1)
Query interface (breaking the rules — spot the SQL!)

HepData query testing

Empty fields will match any value of the corresponding quantity:
- Paper (IR):
- Reaction:
- Observable:
- Detector:

Results
Click on the paper or dataset link to see the data.

SQL query (for debug)

```
SELECT DISTINCT paper.IR as paper_IR, paper.dataset as dataset, paper.description as description, reaction as reaction, observable as observable, detector as detector FROM HepData.paper JOIN HepData.reaction ON paper.IR = reaction.IR JOIN HepData.observable ON paper.IR = observable.IR JOIN HepData.detector ON paper.IR = detector.IR WHERE paper.IR IS NOT NULL AND paper.dataset IS NOT NULL AND paper.description IS NOT NULL AND reaction IS NOT NULL AND observable IS NOT NULL AND detector IS NOT NULL
```

Durham University
HepData demos (2)

HepData → HepML \( \xrightarrow{\text{XSL}} \) HTML

Andy Buckley
CEDAR, part II
HepData demos (3)

HepData → HepML $\xrightarrow{XSL}$ HZTool Fortran

HepData HZTool header generator

Reformatted HepML for HZTool

Andy Buckley
CEDAR, part II
HepData summary

- HepData re-engineering in progress: demos at http://hepdata.cedar.ac.uk
- Not everything is finalised yet...
- Your chance to make sure that it can do everything you want!
- HepML is also incomplete: metadata is not yet fully handled, for example
- Do we need to provide special data set support like correlation matrices?
- Get involved!

Andy Buckley

CEDAR, part II
HepForge — a lightweight development environment for HEP
Software engineering for small HEP projects

- Everyone has written code that might be re-used. . .
- But: “energy barrier” to tidying up / managing it as a project
- Small projects don’t have resources for nice things like:
  - Use std. build/install systems e.g. autotools, libtool
  - Independence of build/run environment (e.g. /cern!)
  - “Formal” quality control, feedback and bug tracking
  - Version control systems (Subversion, CVS)

HepForge aims to reduce this barrier and encourage more publically released, well-developed HEP code
HepForge

- Collaborative development system for HEP
- Online at http://hepforge.cedar.ac.uk
- Features including:
  - Shell access with full set of dev tools
  - Web space (with several convenient features)
  - Subversion/CVS version control (+ Web viewers)
  - Wiki and bug tracker
  - Mailing lists for developers and users
  - Downloads manager
- We’ve done the “boring bits”!
- Many users already: Herwig++, Pythia6, LHAPDF... and ~25 others
Who should be interested in HepForge?

- Who’s it for? Small–medium size projects
- *Probably* not experiments! But not necessarily…
- *You* should:
  - Stand-alone code preferred, since it aids *re-use*
  - Use standard methods, e.g. support *make install*
  - Intention to document properly
  - Support users
  - (General responsible development, essentially!)

In return, HF provides powerful *software development tools* with a *minimal learning curve*
HepForge tour

Home page

CEDAR HepForge

HepForge is a development environment for high energy physics software development projects. Some of the benefits offered by HepForge are:

- Shell account with up to date development tools
- Web page hosting
- CVS and Subversion code management systems
HepForge tour

Project list (1)

Here is the current list of projects using HepForge to do their development. (We will be introducing keyword sorting of projects in time.)

- **ExHuME**: C++ generator of central Exclusive Hadronic Monte-carlo Events
- **FastNLO**
- **FeynML**
- **Fortran Herwig**: Fortran-based Monte Carlo event generator with parton shower
- **HepData**: HepData database and Web interface development project
- **HepForge**: Project to build and maintain the HepForge system!
HepForge tour

Project list (2)

- **HepML**: Proposed interchange formats for MC parameters and HepData records
- **HepTeX**: Collection of HEP-specific TeX/LaTeX packages
- **Herwig++**: IO and steering utilities for HZTool.
- **HZSteer**: Data-MC comparison histogramming
- **JetWeb**: Web-based system for MC event generator validation
- **Jimmy**: Multiple interactions implementation for Herwig
- **KtJet**: C++ implementation of the kT jet clustering algorithm
- **LHAPDF**: Les Houches Accord PDF library and interface
- **Professor**: Tuning tool for MC generators
- **PyFeyn**: Python library for programatically drawing Feynman diagrams
- **Rivet**: C++ re-implementation of the HZTool MC validation tool
- **RivetGun**: Generator interfacing system for Rivet
- **RunMC**: C++ front-end of Monte Carlo models
- **ThePEG**: Platform for using and building C++ event generators
HepForge tour
Project VC listing

HepForge version control

Current revision: 516 (of 516)
Jump to directory revision: 516

Files shown: 0

<table>
<thead>
<tr>
<th>File</th>
<th>Rev.</th>
<th>Age</th>
<th>Author</th>
<th>Last log entry</th>
</tr>
</thead>
<tbody>
<tr>
<td>hdbdmsmigrate/</td>
<td>433</td>
<td>2 months</td>
<td>whalley</td>
<td>changes since move to svn</td>
</tr>
<tr>
<td>hcdomon/</td>
<td>498</td>
<td>12 days</td>
<td>buckley</td>
<td>Moving DBMatch back due to problems with pro</td>
</tr>
<tr>
<td>hdmigrate/</td>
<td>499</td>
<td>12 days</td>
<td>buckley</td>
<td>Moved DBMatch back for convenience</td>
</tr>
<tr>
<td>hdmode/</td>
<td>512</td>
<td>8 days</td>
<td>buckley</td>
<td>Added hasZeroSize() method for determining whether a file is empty</td>
</tr>
<tr>
<td>hdxml/</td>
<td>488</td>
<td>12 days</td>
<td>buckley</td>
<td>Making a new package for HepData HepML writer</td>
</tr>
<tr>
<td>hepdata/</td>
<td>516</td>
<td>4 days</td>
<td>buckley</td>
<td>Adding errors as expected...</td>
</tr>
</tbody>
</table>

hepforge@cedar.ac.uk
Powered by ViewCVS 1.0-dev
HepForge tour
Project downloads listing (also personal copy via SSI)

<table>
<thead>
<tr>
<th>Name</th>
<th>Version</th>
<th>Filename</th>
</tr>
</thead>
<tbody>
<tr>
<td>packages</td>
<td>1.0</td>
<td>packages-1.0.tar.gz</td>
</tr>
<tr>
<td>runmc_win32</td>
<td>4.0</td>
<td>runmc_win32-4.0.tar.gz</td>
</tr>
<tr>
<td></td>
<td>3.3</td>
<td>runmc_win32-3.3.tar.gz</td>
</tr>
<tr>
<td>runmc</td>
<td>4.0</td>
<td>runmc-4.0.tar.gz</td>
</tr>
<tr>
<td></td>
<td>3.3</td>
<td>runmc-3.3.tar.gz</td>
</tr>
</tbody>
</table>
HepForge tour

Documentation: user guide (note scrollbar!)

HepForge user guide

This is the guide to the services provided by HepForge. Please check here and in the FAQ if you have a problem with the system. If you can’t find an answer there, email us at hepforge@cedar.ac.uk.

Shell account and filesystem

Your HepForge account gives you full Unix shell access to our server, with user privileges. You will have personal workspace (if you want to use it) at http://www.cedar.ac.uk/∼<username> which can be accessed by creating a personal publichtml directory containing HTML files: mkdir ~/publichtml; touch ~/public_html/index.html should do the trick. These pages will be visible on the Web at http://www.cedar.ac.uk/∼<username>/.

Project information

Each project’s file space contains a directory called project-config in which various
HepForge tour

Project Web page: LHAPDF

LHAPDF the Les Houches Accord PDF Interface

Home

LHAPDF provides a unified and easy to use interface to modern PDF sets. It is designed to work not only with individual PDF sets but also with the more recent multiple "error" sets. It can be viewed as the successor to PDFLIB, incorporating many of the older sets found in the latter, including pion and photon PDFs. In LHAPDF the computer code and input parameters/grids are separated thus allowing more easy updating and no limit to the expansion possibilities. The code and data sets can be downloaded together or individually as desired. From version 4.1 onwards a configuration script facilitates the installation of LHAPDF.

Contents:
- Installing LHAPDF.
- List of all available PDF sets.
- Online user manual.
- A wrapper for C++.
- A little bit of theory.

Downloads:
- Latest released version:
  - 4.2 (full): lhapdf-4.2.tar.gz
  - 4.2: (no pdfsets): lhapdf-4.2-nopdf.tar.gz
HepForge tour
Project Web page: Herwig++

The Herwig++ Event Generator

Overview

Herwig++ is a completely new event generator, written in C++. It is built on the experience gained with the well-known event generator HERWIG. The aim is to provide a multi-purpose event generator with similar or improved capabilities (like angular ordered parton evolution and the cluster hadronization model). At some point the ongoing development of the Fortran version will terminate and Herwig++ will take over.

Herwig++ is based on ThePEG and CLHEP.

Download

Herwig++ 2.0 beta

The Herwig++ 2.0 beta release has been tested with CLHEP-2.0.2.2 and ThePEG-2006-01-31.

The release note.
HepForge tour

Project wiki

Herwig++ Quick User Guides

Installation and running

- HerwigInstallation
- FirstRun

Writing new modules

- NewMatrixElement
- NewDecayer?
- NewAnalysisHandler?

Misc Examples

- Analysis with PAW
HepForge tour
Project bug tracker: milestones

Roadmap

**Milestone: JetWeb Accessing HepData**
Due in 2 months

98%  
Closed tickets: 5  Active tickets: 1

Also a [CEDAR Milestone](#).

**Milestone: JetWeb and HepData development version**
Due in 3 months

100%  
Closed tickets: 3  Active tickets: 9

Andy Buckley
CEDAR, part II
HepForge tour
Project timeline (integrated with SVN)

Timeline

10/02/06:
- 17:34 Changeset [516] by buckley
  Adding errors as expected.
- 15:19 Changeset [515] by buckley
  Making progress, thanks to new use of XSLT 2.0 rules and the Saxon...

07/02/06:
- 15:57 Changeset [514] by buckley
  Getting H2Tool? Fortran headers working
HepForge tour
Project bug listing

JetWeb Accessing HepData

<table>
<thead>
<tr>
<th>Ticket</th>
<th>Summary</th>
<th>Component</th>
<th>Version</th>
<th>Type</th>
<th>Owner</th>
<th>Created</th>
</tr>
</thead>
<tbody>
<tr>
<td>#2</td>
<td>Create XSL transformer for data HepML \rightarrow HzTool Fortran headers</td>
<td>hepdata</td>
<td>1.0</td>
<td>task</td>
<td>buckley</td>
<td>01/11/05</td>
</tr>
</tbody>
</table>

Object model and db persistency refactoring

<table>
<thead>
<tr>
<th>Ticket</th>
<th>Summary</th>
<th>Component</th>
<th>Version</th>
<th>Type</th>
<th>Owner</th>
<th>Created</th>
</tr>
</thead>
<tbody>
<tr>
<td>#42</td>
<td>Refactor the object model / db interaction</td>
<td>model</td>
<td>1.0</td>
<td>task</td>
<td>buckley</td>
<td>17/01/06</td>
</tr>
<tr>
<td>#54</td>
<td>Improve error representation in the object model</td>
<td>model</td>
<td>1.0</td>
<td>task</td>
<td>buckley</td>
<td>01/02/06</td>
</tr>
<tr>
<td>#46</td>
<td>Consider using SQL double for data values</td>
<td>migration</td>
<td>1.0</td>
<td>enhancement</td>
<td>buckley</td>
<td>21/01/06</td>
</tr>
<tr>
<td>#26</td>
<td>Separate HepData model from migration code</td>
<td>hepdata</td>
<td>1.0</td>
<td>defect</td>
<td>buckley</td>
<td>25/11/05</td>
</tr>
<tr>
<td>#53</td>
<td>Use enums for error type</td>
<td>model</td>
<td>1.0</td>
<td>enhancement</td>
<td>buckley</td>
<td>01/02/06</td>
</tr>
<tr>
<td>#56</td>
<td>Separate stat/sys and</td>
<td>model</td>
<td>1.0</td>
<td>enhancement</td>
<td>buckley</td>
<td>02/02/06</td>
</tr>
</tbody>
</table>

Andy Buckley
CEDAR, part II
HepForge tour
Project bug details (1)

Ticket #2 (task)

Create XSL transformer for data HepML -> HzTool Fortran headers

Status: assigned

Reported by: buckley
Assigned to: buckley (accepted)
Priority: major
Milestone: jetWeb
Component: hepdata
Version: 1.0
Keywords: Cc: jmb

HZTool requires a Fortran header file for each paper. The existing XSLT stylesheet for the HepML -> Fortran transformation is very incomplete and needs work.
HepForge tour

Project bug details (2)

Attachments

Change History

17/01/06 17:55:28: Modified by buckley

- description changed.
- milestone set to JetWeb Accessing HepData.

I've re-written the XSL transformer classes somewhat. The next steps are:

- abstract the re-formatter classes to be more generic: HepMlReformatter, HepMlsPltReformatter, HepMLTextReformatter etc. (need better names)
- provide a superclass/interface common to

Andy Buckley

CEDAR, part II
HepForge tour

The HepForge registration form! Think about it...
Features still to come

- Keyword and category project metadata
  (for HepCode & general user convenience)
- Web interface to project metadata
  (keywords, description, “pretty” project name...)
- FAQ, news, \texttt{\LaTeX} ... filters
- SSL-encryption of protected project areas
- And many others (see HepForge’s own HF project!)
Registration

Please think about using HepForge!

- Requirements:
  - has to be for a re-useable HEP project
  - not for processor-intensive use
  - commitment to document and support your project
  - encouraged to use standard build procedures etc.

We can and will help with this!

Visit [http://hepforge.cedar.ac.uk](http://hepforge.cedar.ac.uk) to register
HepForge summary

- Designed to be easy to use, pick those features useful to you and ignore the rest
- Designed to encourage modular, reusable, well-documented HEP software
- If you have a project which fits those descriptions, think about using HepForge, it could save you a lot of hassle
- Herwig, Pythia, ThePEG, SoftSusy, LHAPDF, Hoppet, FastNLO, ExHuME... already do!
- (Plus all the CEDAR sub-projects, of course)
Summary
CEDAR is primarily a generator tuning system, combining JetWeb and HepData.

HepData in its new incarnation will use HepML and other standard interfaces.

We need user (expt) input about what you want us to be doing for LHC data!

HepForge is available for HEP software development now!

Eventually will be used to implement the HepCode repository.

Feedback has been very positive: system is powerful but very easy to use.

Consider HepForge for your re-useable HEP code!
HepForge backend (1)
For the interested. . .

- Python-based with shell scripts for acc. management
- Various Subversion things, e.g.
  - re-write of CVS: almost complete UI compatibility
  - support for symlinks, metadata, dirs, `mv/cp`!
  - anon read access over HTTP; rw dev access over SSH
  - use it! Migration from CVS is easy.
- ViewVC with multi-site hack
HepForge backend (2)

For the interested...

• Trac bug tracker and wiki:
  • SQLite backend and SVN integration
  • Moin-compatible wiki
  • excellent tool!

• Web system with post-processing scripts
  • HTML Tidy
  • transparent header and footer handling
  • code highlighting, Markdown...

• Download manager (personalisable)