

ILC Cross Section Database: command line tool

C. Calancha (KEK)
calancha@post.kek.jp

2014, June 13

Motivation (*)

- Last week i presented the web browser interfaz.
 - Text-based web browsers are not supported (tested `lynx` and `w3m`).
 - It seems this is expected since `drupal v7`.
- Having a command line version running at kekcc is desirable on its own.
- It is better an official tool instead of every user collecting the cross sections.
- It's useful.
- It's fun developing this.

Design targets

- Normal user should be available to run it.
 - Not enough privileges to run sql queries on kekcc.
- It should be fast and scalable.
- Easy to use.
- Extensible when new production samples comming.
- It should report clear error messages about wrong input.
- Thought as a tool for interactive session: no intention to implement it within Marlin framework.
- Portability: no `POSIX` restrictions, my target is to be used at *KEK* on kekcc.

(*) For a comprehensive introduction see my previous week report.

ilc-xsec-db

- Include in your PATH the directory where ilc-xsec-db is located:
 - `export PATH=/group/ilc/soft/samples/ilc-xsec-db/bin:$PATH`
- Interfaz accept several options to set parameters and perform finer query.
 - Both `UNIX` and `GNU` option styles are supported.
- Tested on kekcc cluster and it works.
- If no options provided (or called with `'-h'`) documentation on the options is displayed.
 - Please, read this documentation to use this tool efficiently.

How it performs

- list all 'nnh' processes at ECM = 250 GeV:
 - Start: Fri Jun 13 13:24:57.431057000 JST 2014
 - End: Fri Jun 13 13:24:57.638698000 JST 2014
- Wow! that was really fast!
- List all samples contained in the db:
 - Start: Fri Jun 13 13:17:44.978739000 JST 2014
 - End: Fri Jun 13 13:17:51.055819000 JST 2014
- 6 seconds: ok for an interactive session.
- The user interface consume most of the time:
 - From the bash manual: 'BUGS: It's too big and too slow.'

Scalability

- The web browser interfaz presented last week is sql based (scalability not a problem).
- The core implementation of the terminal tool is based on hash tables.
 - Average cost independent of the number of elements stored.
 - Including future TDR samples will not introduce any performance penalties.

Possible extension: argument lists

- Current version only accept single-value arguments (`--name=nnh`)
- One natural extension would be accept something like: `--name=(nnh,4f.sznu_l)`
 - Considering to support this in the future (if i have time).
- With current version, you can 'simulate' such things calling program in a loop:

Example

```
echo ; (  
  names="nnh 4f_sznu_l"  
  ecm="250 500"  
  for n in $names  
  do  
    for e in $ecm  
    do  
      ilc-xsec-db --name=$n --ecm=$e  
    done  
  done  
)
```

Summary

- Developed command line version of my previous ILC cross sections database.
- It runs at kekcc.
- It does not require special privileges.
- It match my design requirements:
 - Fast enough for interactive session.
 - Reliable, easy to use, easy to extend with new samples, scalable.

Plan

- In principle this task is finished.
 - Keep small maintenance for fix eventual bugs found by users.
 - I could extend capabilities if i found the time
 - i.e. adding list arguments support could be nice.
- Plan to comeback to my analysis studies (rare decays).

BACK UP