
Database/MySQL in analysis

Alexandre Camsonne
Hall A Jefferson Laboratory

Hall A analysis workshop
December 12th 2007

Overview

- Introduction to Mysql
- Use of Mysql in data taking and analysis
- Examples
- Conclusion

Mysql

- Free open source
- Relational database
- Client/Server
 - Sorting
 - Advanced query
 - Correlating information
 - Information dispatching : all analysis have same constants variables
 - Information display

 - But one needs a server !

API

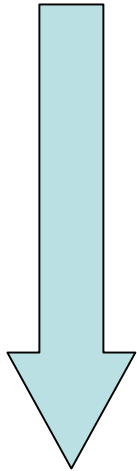
- Widely supported
- Application programming interface
 - C,C++
 - ROOT : TSQLServer
 - Perl DBI
 - Tcl (recent version)
 - Php
 - Excel, OpenOffice
 - Python...

Analysis workflow

- Data taking
 - Run informations
 - Data taking conditions
 - Data quality information
- Calibration
 - Calibrations coefficient
 - Offsets, gain
- Analysis
 - Cuts
 - results

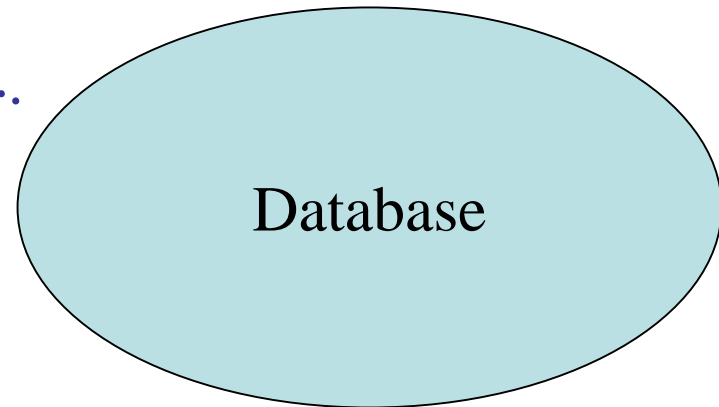
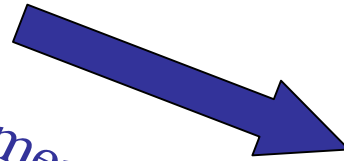
Analysis workflow

Data taking



Replay

EPICS : momentum, angle, ...
Pedestals, HV ...
Information by run



Database

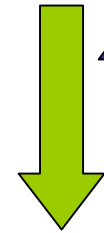
Kinematic



Charge, deadtime
calibration



Kinematic
calibrations



Results

Analysis

Data taking

- Run informations
 - Managing replay
 - Sorting by run types, target, momentum...
 - Monitoring : HV
 - Run quality : manual flagging of runs
- Implementation
 - Semi-online : in run start and run end
 - Offline from
 - Perl script parsing text files generated by DAQ
 - HALOG : same as previous but from HALOG
(can regenerated database offsite)
 - ELOG or database driven logbook

Calibration

- Calibrations constants
 - pedestal
 - PMT gain
 - Timing offsets
 - db_run.dat
- Implementation
 - Regenerate current DB from database
 - Scripts
 - Starting point [Werner Boeglin](#)
 - Direct query to database
 - root mysql interface : GEn, DVCS

Analysis

- Use all previous data
- Use from root to generate cuts
- Can store informations useful for cross section
 - THaNormAna

- Job management
 - File location
 - Replayed runs

- Results display

Display

- HTML generation
 - [LEDEX](#) (not using Mysql but same idea)
- PHP
 - Experiments
 - [Compton](#)
 - Gen
 - Small Angle GDH
 - [E04-018](#)
 - CSR
 - Great for searching
 - Security issues : now cannot run from user's directory

Mysql in experiments

- Compton Polarimeter
- Small Angle GDH
- HAPPEX
- DVCS
- LEDEX, E04-018, CSR

GE_n

- EPICS tables
- Run list with data quality
- <http://hallaweb.jlab.org/experiment/E02-013/DB/>
- http://hallaweb.jlab.org/experiment/E02-013/wiki/tiki-download_wiki_attachment.php?attId=23&page=GE_nDB

Happex

- EPICS tables
- Run list with manual data quality check during data taking
- Cuts for the analysis based on database

Small angle GDH

- HALOG search
 - <http://www.jlab.org/%7Eesinghj/runsummary/search.php>
- EPICS tables from CODA file
- Run list with manual data quality check during data taking
 - <http://www.jlab.org/%7Eesinghj/runsummary/>
- Jaideep Singh's software
<http://hallaweb.jlab.org/software/tools/jaideep.html>

DVCS

- Base class with detector informations
 - Load data at beginning of run analysis
 - Run selection
 - Detector geometry
 - Kinematic : angle, momenta
 - Pedestals, reference pulse for each channel
 - PMT gain characteristics
 - Timing and positions offsets, calibration to energy
- Class for data access :
 - [Data structure](#)
 - Database access :
 - <http://www.jlab.org/~munoz/soft/DB/index.html>
 - <http://www.jlab.org/~munoz/soft/html/doc/TDVCSDB.html>

Coulomb sum rule

- Run liste
 - Generated from HALOG with perl script
 - Located on adaq at :
/adaqfs/home/adaq/e05110/perl/e05110.pl
 - Display with PHP
 - http://hallaweb.jlab.org/experiment/E05-110/exp_home
 - Need to add automatic updating at start/end of run
- Use in macro : example of list of runs for NaI calibration.
It put all runs at a given momentum to have enough statistics.

Coulomb sum rule

```
TString query;  
  query = Form ("select * from runliste where abs(momentum-  
%f)<0.002 and run <20000  and abs(energy-%f)<1 and abs(angle-  
%f)<1. order by stamp",p0,energy,angle);  
  //query = Form ("select * from runliste where  
run=366",p0,energy,angle);  
  TSQLServer *  
serv=TSQLServer::Connect("mysql://server","username","");  
  TSQLResult * result=serv->Query(query.Data());  
  TSQLRow * row;  
row = result->Next();
```

Query a
kinematic

Coulomb sum rule

```
while (row!=0)
{
  run = atoi (row->GetField(0));
  cout<<"Run number :"<<run<<endl;
  fname=Form("/adaql10/work1/E05110/fromfarm/onlyphys_%d.root",run);
  ch->Add(fname.Data());
  split=1;
  lafile=Form("onlyphys_%d_1.root",run);
  goodname=gSystem->FindFile(dirname.Data(),lafile);
  while (gSystem->FindFile(dirname.Data(),lafile))
  {
    goodname=gSystem->FindFile(dirname.Data(),lafile);
    ch->Add(goodname.Data());
    split++;
    lafile=Form("onlyphys_%d_%d.root",run,split);
    cout<<goodname.Data()<<endl;
  }
  row = result->Next();
}
```

Generate file list

Implementation in Hall A

- Coordinate efforts
- Include development in standard analyzer
- Official dedicated server

- Work on scripts for integration into DAQ

Importance of data structure

- Selection of variables to record
 - Difficulties to have flexible software and table structure
- Very general structure
 - Flexible
 - But slow
- Possibility to generate tables from other tables with scripts
- Need some thoughts
- Hall B calibration data structure

Conclusion

- Use of database can be useful and time saving
- Mysql : preferred choice since widely available
- Many implementations done for experiments

- Would be nice to have something for all experiments but my need a bit of user support