Analysis with MySQL

Shujie Li
With Brad Sawatzky, Tyler Hague

Hall A/C Analysis Workshop
06.25.2018
MySQL introduction
  ○ What is …
  ○ Example
  ○ Create run info table from Logbook
  ○ python/ROOT MySQL connector

MySQL at JLab
  ○ RCDB (run condition database)
  ○ Tritium database
  ○ RCDB for Hall A (work with Brad Sawatzky)
MySQL 101

SQL (Structured Query Language): relational database management system

MySQL: an open source version of SQL

MySQL server: store data, command line interface (already installed on the virtualbox thanks to Ole!)

Advantage:

- Fast, good to store large, structured data
- Commands easy to read / learn
- Allow multiple people access / edit data
- Interface with webpage, python, C++ (ROOT)

MySQL Server -> databases -> Tables -> Columns -> Rows (Entries)
MySQL 101

References:
https://www.ntu.edu.sg/home/ehchua/programming/sql/MySQL_Beginner.html

- Semi colon to end command ;
- Enter to break line;
- Command is not case sensitive;
Log in MySQL

`>> mysql -u <username> -p<password> -h <hostname> <database name>`

Welcome to the MariaDB monitor. Commands end with ; or \\g.
Your MariaDB connection id is 7
Server version: 5.5.56-MariaDB MariaDB Server

Copyright (c) 2000, 2017, Oracle, MariaDB Corporation Ab and others.
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> use workshop;
Database changed
MariaDB [workshop]> show tables;
Empty set (0.00 sec)
MariaDB [workshop]> 1
Create Table

```
>> create table target( name varchar(20) default null, type varchar(20) default null, density float(8,3) not null, primary key (name, type));
```

```
MariaDB [workshop]> show tables;
+-------------------------------+
| Tables_in_workshop            |
|-------------------------------+
| target                        |
|-------------------------------+
1 row in set (0.00 sec)

MariaDB [workshop]> describe target;
+-----------------+-----------------+-----+-----+---------+-----------------+
<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Null</th>
<th>Key</th>
<th>Default</th>
<th>Extra</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>varchar(20)</td>
<td>NO</td>
<td>PRI</td>
<td></td>
<td></td>
</tr>
<tr>
<td>type</td>
<td>varchar(20)</td>
<td>NO</td>
<td>PRI</td>
<td></td>
<td></td>
</tr>
<tr>
<td>density</td>
<td>float(8,3)</td>
<td>NO</td>
<td>NULL</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
3 rows in set (0.00 sec)
```
MariaDB [workshop]> insert into target values ('tritium','gas',0.77);
Query OK, 1 row affected (0.00 sec)

MariaDB [workshop]> insert into target values ('helium-3','gas',0.53),('carbon foil','solid',0.12);
Query OK, 2 rows affected (0.01 sec)
Records: 2  Duplicates: 0  Warnings: 0

MariaDB [workshop]> update target set density=0.077 where name='tritium';
Query OK, 1 row affected (0.00 sec)
Rows matched: 1  Changed: 1  Warnings: 0

MariaDB [workshop]> update target set density=density/10.0 where name='helium-3';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0

MariaDB [workshop]> select * from target;
+-----------------+--------+-------+
| name            | type   | density |
| carbon foil     | solid  | 0.120  |
| helium-3        | gas    | 0.053  |
| tritium         | gas    | 0.077  |
+-----------------+--------+-------+
Edit table Content

MariaDB [workshop]> alter table target add beamtime float(8,3) not null;
Query OK, 3 rows affected (0.02 sec)
Records: 3  Duplicates: 0  Warnings: 0

MariaDB [workshop]> describe target;

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Null</th>
<th>Key</th>
<th>Default</th>
<th>Extra</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>varchar(20)</td>
<td>NO</td>
<td>PRI</td>
<td></td>
<td></td>
</tr>
<tr>
<td>type</td>
<td>varchar(20)</td>
<td>NO</td>
<td>PRI</td>
<td></td>
<td></td>
</tr>
<tr>
<td>density</td>
<td>float(8,3)</td>
<td>NO</td>
<td></td>
<td>NULL</td>
<td></td>
</tr>
<tr>
<td>beamtime</td>
<td>float(8,3)</td>
<td>NO</td>
<td></td>
<td>NULL</td>
<td></td>
</tr>
</tbody>
</table>

4 rows in set (0.00 sec)

MariaDB [workshop]> update target set beamtime=100 where type='gas';
Query OK, 2 rows affected (0.00 sec)
Rows matched: 2  Changed: 2  Warnings: 0

MariaDB [workshop]> update target set beamtime=10 where type='solid';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0

MariaDB [workshop]> select * from target;

<table>
<thead>
<tr>
<th>name</th>
<th>type</th>
<th>density</th>
<th>beamtime</th>
</tr>
</thead>
<tbody>
<tr>
<td>carbon foil</td>
<td>solid</td>
<td>0.120</td>
<td>10.000</td>
</tr>
<tr>
<td>helium-3</td>
<td>gas</td>
<td>0.053</td>
<td>100.000</td>
</tr>
<tr>
<td>tritium</td>
<td>gas</td>
<td>0.077</td>
<td>100.000</td>
</tr>
</tbody>
</table>

3 rows in set (0.00 sec)
Display Table Content

**MariaDB [workshop]>** select sum(beamtime) from target where type='gas';

```
<table>
<thead>
<tr>
<th>sum(beamtime)</th>
</tr>
</thead>
<tbody>
<tr>
<td>200.000</td>
</tr>
</tbody>
</table>
```

1 row in set (0.00 sec)

**MariaDB [workshop]>** select name, density from target order by density desc;

```
<table>
<thead>
<tr>
<th>name</th>
<th>density</th>
</tr>
</thead>
<tbody>
<tr>
<td>carbon</td>
<td>0.120</td>
</tr>
<tr>
<td>foil</td>
<td>0.077</td>
</tr>
<tr>
<td>tritium</td>
<td>0.053</td>
</tr>
<tr>
<td>helium-3</td>
<td></td>
</tr>
</tbody>
</table>
```

3 rows in set (0.00 sec)
Python Connector Example:

Create_table.py:

Create a table called “runinfo”

Log2db.py (need the ‘requests’ module)

- Talk to logbook API to get start/end of run content
- Read desired information, write into database

Always CLOSE connection in the end of scripts!
Import/export SQL file

Export:

`mysqldump -u... -p... workshop runinfo > workshops.sql`

Import:

`mysql> use workshop;`
`mysql> source workshop.sql;`
Import/export CSV file

-- need to create table structure in advance, use the command below or run create_table.py

mysql> create table cer_L(run_number int not null, pmt_id int(2) unsigned not null, pedestal int(4) not null, gain int(4) not null, primary key (run_number, pmt_id));

-- load the file:

mysql> LOAD DATA INFILE "cerL.csv" INTO TABLE cer_L COLUMNS TERMINATED BY '\t' OPTIONALLY ENCLOSED BY '"' ESCAPED BY '"' LINES TERMINATED BY '\n' IGNORE 1 LINES;
ROOT Example:
https://root.cern.ch/doc/v608/group__tutorial__sql.html

```cpp
#include "TSQLServer.h"
#include "TSQLResult.h"
#include "TSQLRow.h"

......

TSQLServer* Server = TSQLServer::Connect("mysql://localhost/workshop", "username", "password");
TString query(Form("select target from runinfot where run_number=%d", runnum));
TSQLResult* result = Server->Query(query.Data());
TSQLRow *row;
row = result->Next();
string tempname;
tempname = row->GetField(0);
Server->Close();
```
Using SQL at JLab

Hall D:

- **RCDB (run condition database)**
  - Web interface: [https://halldweb.jlab.org/rcdb/](https://halldweb.jlab.org/rcdb/)
- **CCDB (Calibration database)**

Hall A:

- **G2p**: [https://hallaweb.jlab.org/experiment/g2p/mysql/](https://hallaweb.jlab.org/experiment/g2p/mysql/)
Coming Soon: RCDB for Hall A/C

- Work in-progress with Brad and the Hall D RCDB team
- Use the existing RCDB platform maintained by Dmitry Romanov from Hall D
- Well-designed structure, customized connector, web interface
- One database for all experiments, no re-inventing wheels
Coming Soon: RCDB for Hall A/C

- Work in-progress with Brad and the Hall D RCDB team
- Use the existing RCDB platform maintained by Dmitry Romanov from Hall D
- Well-designed structure, customized connector, web interface
- One database for all experiments, no re-inventing wheels
Thanks Hall D RCDB team for the support!