The MariaDB CONNECT Storage Engine

Serge Frezefond

http://serge.frezefond.com

@sfrezefond
Who am I?

- Serge Frezefond
- Principal Sales Engineer @ SkySQL
- Joined MySQL Ab in 2006
- Worked for MySQL@Sun and MySQL@Oracle until July 2011
Goal of the CONNECT Storage Engine: BI on various targets

Most of the data in companies is in various external datasources (many in non relational database format):

- relational databases: Oracle, SQL Server...
- Dbase, Firebird, SQLite
- Microsoft Access & Excel
- Distributed mysql servers
- DOS, FIX, BIN, CSV, XML
- stored per column...

Not targeted for OLTP
Data is processed by an ETL
  – Change in the data model (denormalization...)

Agregates are computed
  – Need to be defined and maintained

Might need to move data out of RDBMS to other kind of datastore
  – OLAP, Column store, Hadoop/Hbase ...

Specific tools are used to query the data

**IT is involved to maintain this machinery**
The CONNECT Storage Engine

• What is the CONNECT storage engine?
  – A storage engine that enables MariaDB to use external data as they were standard tables in the server
  – Data is not loaded into MariaDB

• History of the CONNECT storage engine
  – Developed by Olivier Bertrand, an ex IBM database researcher
  – The idea dates back in 2004 and Olivier has been in touch with MySQL and MariaDB since
The MySQL Plugin Architecture

• Plugin Architecture is a major differentiator of MySQL
• Datastores can interact with the MySQL SQL layer
• Allow advanced interaction
• Specific Create Table parameters (MariaDB)
• Auto-discovery of table structure (MariaDB)
• Condition push down
• Allow join with other storage engines
  – InnoDB / MyISAM tables
The CONNECT Storage Engine
CONNECT Engine Usage

- Integrates/access data directly in many non-MariaDB formats
- Simplifies the ETL procedures in Business Intelligence and Business Analytics
- Simplifies the export/import of data from/to MariaDB, to/from other data sources
- More powerful than CSV, FederatedX and Merge Engines
- FILE privilege is required
The CONNECT Storage Engine implements advanced features

- **Condition Push down**
  - Used with ODBC and MySQL to push condition to the target database. Big perf gain
    
    set optimizer_switch='engine_condition_pushdown=on'

- **Support MariaDB virtuals columns**
- **Support of special columns:**
  - Rowid, fileid, tabid, servid
- **Extensible with the OEM file type**
- **Catalog table for table metadata(ODBC ...)**
CONECT
File table type

- DOS, FIX, BIN, FMT, CSV, INI, XML
- Support virtual tables (DIR)
- Large tables support (>2GB)
- Compression - gzip format
- Memory file mapping
- Add read optimized indexing to files
- Multiple CONECT tables can be created on the same underlying file
- Indexes can be shared between tables
<?xml version="1.0" encoding="ISO-8859-1"?>
<BIBLIO SUBJECT="XML">
  <BOOK ISBN="9782212090819" LANG="fr" SUBJECT="applications">
    <AUTHOR>
      <FIRSTNAME>Jean-Christophe</FIRSTNAME>
      <LASTNAME>Bernadac</LASTNAME>
    </AUTHOR>
    <TITLE>Construire une application XML</TITLE>
    <PUBLISHER>
      <NAME>Eyrolles</NAME>
      <PLACE>Paris</PLACE>
    </PUBLISHER>
    <DATEPUB>1999</DATEPUB>
  </BOOK>
</BIBLIO>
create table xsampall (
    isbn char(15) field_format='@ISBN',
    authorln char(20) field_format='AUTHOR/LASTNAME',
    title char(32) field_format='TITLE',
    translated char(32) field_format='TRANSLATOR/@PREFIX',
    year int(4) field_format='DATEPUB')
engine=CONNECT table_type=XML file_name='Xsample.xml'
tabname='BIBLIO' option_list='rownode=BOOK,skipnull=1';
### XMLTable Type

**Query Result**

```sql
select isbn, subject, title, publisher from xsamp2;
```

<table>
<thead>
<tr>
<th>ISBN</th>
<th>SUBJECT</th>
<th>TTITLE</th>
<th>PUBLISHER</th>
</tr>
</thead>
<tbody>
<tr>
<td>9782212090819</td>
<td>applications</td>
<td>Construire une application XML</td>
<td>Eyrolles Paris</td>
</tr>
<tr>
<td>9782840825685</td>
<td>applications</td>
<td>XML en Action</td>
<td>Microsoft Press</td>
</tr>
</tbody>
</table>

Can also generate HTML
XCOL Table Type

<table>
<thead>
<tr>
<th>Name</th>
<th>childlist</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sophie</td>
<td>Manon, Alice, Antoine</td>
</tr>
<tr>
<td>Valentine</td>
<td>Arthur, Sidonie, Prune</td>
</tr>
</tbody>
</table>

CREATE TABLE xchild (  
mother char(12) NOT NULL flag=1,  
child varchar(30) DEFAULT NULL NULL flag=2)  
ENGINE=CONNECT table_type=XCOL  
tabname='children' option_list='colname=child';
select * from xchild;

mother    child
Sophie    Manon
Sophie    Alice
...

select count(child) from xchild;    returns 10
### OCCUR Table Type

<table>
<thead>
<tr>
<th>Name</th>
<th>dog</th>
<th>cat</th>
<th>rabbit</th>
<th>bird</th>
<th>fish</th>
</tr>
</thead>
<tbody>
<tr>
<td>John</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Bill</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Mary</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

... create table xpet (  
    name varchar(12) not null,  
    race char(6) not null,  
    number int not null )  
engine=connect table_type=occur tabname=pets  
option_list='OccurCol=number,RankCol=race'  
Colist='dog,cat,rabbit,bird,fish';
### OCCUR Table Type

```sql
select * from xpet;
```

<table>
<thead>
<tr>
<th>Name</th>
<th>race</th>
<th>number</th>
</tr>
</thead>
<tbody>
<tr>
<td>John</td>
<td>dog</td>
<td>2</td>
</tr>
<tr>
<td>Mary</td>
<td>dog</td>
<td>1</td>
</tr>
<tr>
<td>Mary</td>
<td>cat</td>
<td>1</td>
</tr>
<tr>
<td>Lisbeth</td>
<td>rabbit</td>
<td>2</td>
</tr>
</tbody>
</table>

...
<table>
<thead>
<tr>
<th>Who</th>
<th>Week</th>
<th>What</th>
<th>Amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>Joe</td>
<td>3</td>
<td>Beer</td>
<td>18.00</td>
</tr>
<tr>
<td>Beth</td>
<td>4</td>
<td>Food</td>
<td>17.00</td>
</tr>
<tr>
<td>Janet</td>
<td>5</td>
<td>Beer</td>
<td>14.00</td>
</tr>
<tr>
<td>Joe</td>
<td>3</td>
<td>Food</td>
<td>12.00</td>
</tr>
</tbody>
</table>

...  

create table pivex  
Engine=connect table_type=pivot  
tablename=expenses;
```
select * from pivex;
```

<table>
<thead>
<tr>
<th>Who</th>
<th>Week</th>
<th>Beer</th>
<th>Car</th>
<th>Food</th>
</tr>
</thead>
<tbody>
<tr>
<td>Beth</td>
<td>3</td>
<td>16.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Beth</td>
<td>4</td>
<td>15.00</td>
<td>0.00</td>
<td>17.00</td>
</tr>
<tr>
<td>Beth</td>
<td>5</td>
<td>20.00</td>
<td>0.00</td>
<td>12.00</td>
</tr>
<tr>
<td>Janet</td>
<td>3</td>
<td>18.00</td>
<td>19.00</td>
<td>18.00</td>
</tr>
</tbody>
</table>

...
Connect Storage Engine
VEC table / Column store

- 1 or per column file
- Indexes work
- IOs optimization, reads only columns that are requested by the query
CONNECT Storage Engine
ODBC table type

Allow to access to any ODBC datasource.
  – Excel, Access, Firebird, SQLite
  – SQL Server, Oracle, DB2

- Supports insert, update, delete and any other commands
- Multi files ODBC: consolidated monthly excel datasheet
- Access to ODBC and UnixODBC data sources
- WHERE conditions are push to the ODBC source
create table customers engine=connect
table_type=ODBC block_size=10	
 tabname='Customers'
Connection='DSN=MS Access Database;DBQ=C:/Program Files/Microsoft Office/Office/1033/FPNWIND.MDB;'
ODBC database access
From a linux box

• UnixODBC must be used as an ODBC Driver manager.

• The ODBC driver of the target database must be installed
  – For Oracle, DB2
    – install Oracle Database instant Client with ODBC suplement
ODBC access database any command to ODBC target

create table crlite (  
  command varchar(128) not null,  
  number int(5) not null flag=1,  
  message varchar(255) flag=2)  
engine=connect table_type=odbc  
connection='Driver=SQLite3 ODBC Driver;Database=test.sqlite3;NoWCHAR=yes'  
option_list='Execsrc=1';
ODBC Database Access
Any command to ODBC target

```
select * from crlite where command = 'update lite set birth = "2012-07-14" where ID = 2';
```

Can be wrapped in a procedure:
```
create procedure send_cmd(cmd varchar(255))
select * from crlite where command = cmd;
call send_cmd('drop tlite');
```
CONNECT Storage Engine
MYSQL table type vs. Federated(X)

- Condition LIMIT push down
- Implements condition push down
- Autodiscovery of table structure
- Can define the subset of columns we want to see and type conversion
- Access local or remote MySQL tables
Connect Storage Engine
MYSQL table type (a proxy table)

same syntax as federatedx:

create Table lineitem1
ENGINE=CONNECT TABLE_TYPE=MYSQL
connection='mysql://proxy:pwd1@node1:3306/dbt3/lineitem3';
create Table lineitem1
ENGINE=CONNECT TABLE_TYPE=MYSQL
SRCDEF='select l_suppkey, sum(l_quantity) qt from
dbt3.lineitem3 group by l_suppkey'
connection='mysql://proxy:pwd1@node1:3306/dbt3/lineitem3';
Table list table: Collection of tables seen as one

- Tables can be from different storage engines (Not only MyISAM tables)

- Tables may have different column structure

- Underlying tables can be remote / Distributed architecture (ODBC, MySQL)
TBL Table Type (// Merge)

Node 0
- LOCAL
- MYSQL / ODBC

Node 1
- col1
- col2

Node 2
- col1
- col2
- col3

Node 3
- col1
- col2
- col3
- col4

MYSQL table

ODBC table

Table
- Type
- (// Merge)

Muti tables table (like merge)
- Different structure, not myisam only,
- remotely distributed tables
Parallel execution on distributed sharded tables

Node 0

Node 1

Node 2

Node 3

MySQL table

ODBC table

MySQL / ODBC

TBL

col1  col2

col1  col2  col3

col1  col2  col3  col4
Importing /exporting MySQL data in various formats

Importing file data into MySQL tables
– Here for example from an XML file:
  – `create table biblio select * from xsampall2;`

Exporting data from MySQL: Here if we export to XML format:

```
create table handout engine=CONNECT table_type=XML
file_name='handout.htm' header=yes
option_list='name=TABLE, coltype=HTML, attribute=border=1; cellpadding=5'
select plugin_name handler, plugin_description description from
  information_schema.plugins where plugin_type = 'STORAGE ENGINE';
```
Ideas / Roadmap

- Alter table improvement
- ODBC type improvement
- MySQL table type improvement
- Batch key access (MRR/BKA)
- Partition based TBL type (Like Spider)
- Adaptative query ( // MySQL Cluster) ?
- JSON File format
- Transactional / XA support
CONNECT is open source
You can help

• It is 100 % open source
• Sources on MariaDB launchpad
• Open Bug database
• Public Roadmap
• Test cases are released
• Improvement request / worklog
• Well Documented

Try it
Conclusion

- The MariaDB Connect Storage Engine:
  - Open MariaDB to BI and data analysis
  - Simplify heterogeneous data integration
  - Brings real value to MariaDB users
  - Illustrates openness of MariaDB community
  - Supported by SkySQL / MariaDB
Serge Frezefond serge.frezefond@skysql.com
@sfrezefond
http://serge.frezefond.com

Documentation:
https://mariadb.com/kb/en/connect/

MySQL is a registered trademark of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.
SkySQL is not affiliated with MySQL.