

Fox River Watershed Data

Database Conversion (MS access to Sqlite3)

Used mdbtools (linux) to convert MS access (mdb) to Sqlite3

Shell script

mdb2sqlite.sh

```
#!/bin/bash
# Inspired by
# https://www.codeenigma.com/community/blog/using-mdbtools-nix-convert-microsoft-access-mysql

# USAGE
# Rename your MDB file to migration-export.mdb
# run ./mdb2sqlite.sh migration-export.mdb
# wait and wait a bit longer...

now=$(date +%s)
sqlite=sqlite3
fname=$1
sql=${fname/mdb/sqlite}
schema=${fname/mdb/schema}
dir=${fname/.mdb/}-${now}

mkdir $dir

mdb-schema $fname sqlite > $dir/$schema

for i in $( mdb-tables $fname ); do
  echo $i
  mdb-export -D "%Y-%m-%d %H:%M:%S" -H -I sqlite $fname $i > $dir/$i.sql
done

< $dir/$schema $sqlite $sql

for f in $dir/*.sql ; do
  echo $f
  (echo 'BEGIN;'; cat $f; echo 'COMMIT;') | $sqlite $sql
done
echo "Using $dir"
```

Overview website: <http://ilrdss.isws.illinois.edu/fox/>

To acquire the CSV file from MS access we first converted the .mdb database into SQLite. Once we had access to the .sqlite3 file we created a python script to access it, execute SQL statements, and generate a CSV file from the resulting table.

- **Data location:** gltg-source-data/Fox River Watershed Database
 - CSV for latest sqlite3 file: selected_final_foxdb_20171116.csv
 - Lastest MS access db: FoxDB_20171116.mdb
 - Converted sqlite3 file: foxdb_20171116.sqlite3 (please use this one)
- **Database design document:**
 - All documents: http://ilrdss.sws.uiuc.edu/fox/fox_report_phase1.asp?ws=3
 - General description http://ilrdss.sws.uiuc.edu/fox/downloads/Fox_Chapter_4.pdf
 - Section 4.4.1 shows a good example of SQL query
- **SQL statement:**

```

SELECT tblsample.station_id,
       tblstation_information.latitude,
       tblstation_information.longitude,
       tblstation_information.place_name_description,
       tblsample.sample_code,
       tblsample.idloc,
       tblparameter_codes.full_name,
       tblparameter_codes.short_name,
       tblsample.start_date,
       tblresults.result_value,
       tblparameter_group.parameter_group
FROM   ((( (tblresults
           INNER JOIN tblsample
                 ON ( tblresults.sample_code = tblsample.sample_code )
                 AND ( tblresults.idloc = tblsample.idloc ))
        INNER JOIN tblparameter_codes
                 ON tblresults.parameter_code =
                   tblparameter_codes.parameter_code)
       INNER JOIN tblqappgroups
                 ON tblparameter_codes.parameter_code =
                   tblqappgroups.parameter_code)
       INNER JOIN tblqapp_group_codes
                 ON tblqappgroups.qappcode = tblqapp_group_codes.qappcode)
       INNER JOIN tblparameter_group
                 ON tblqapp_group_codes.parameter_group =
                   tblparameter_group.parameter_group)
LEFT JOIN tblstation_information
       ON tblstation_information.station_id = tblsample.station_id
WHERE  (( ( tblparameter_group.parameter_group ) = 9
         OR ( tblparameter_group.parameter_group ) = 10 ))

```

- **Python 3 code:**

```

""" Access a sqlite3 file,
    executes a SQL statement, and
    generates a CSV file for parsing

Variables
-----
data_loc : str
    The location of the database (for example, ./foxdb/foxdb_20171116.sqlite3)
query : str
    The SQL statement

"""
import sqlite3
import pandas as pd

data_loc = ""
query = ""

conn = sqlite3.connect(data_loc)
df = pd.read_sql_query(query, conn)
df['Start_Date'] = pd.to_datetime(df['Start_Date'], errors='coerce') #use this to format date and time to only date
df.to_csv("output.csv", encoding='utf-8', index=False, date_format='%Y-%m-%d')

conn.close()

```

Resources:

- Presentation at Nutrient Monitoring Council that explains what kinds of query [Jong Lee](#) did to extract the data from the database.



NMC-Fox-River-Data.pptx

- **Python sqlite3 module**
 - Python 2: <https://docs.python.org/2/library/sqlite3.html>
 - Python 3: <https://docs.python.org/3/library/sqlite3.html>
- **Python Data Analysis Library**
 - <http://pandas.pydata.org/pandas-docs/stable/>