Introduction to MySQL

FHWA Resource Center

EPA Office of Transportation and Air Quality

EPA-420-B-09-023

Why Learn About MySQL?

- **Custom inputs**
  - MOVES has importers for most common types of inputs, but some MySQL expertise would be helpful for using specialized inputs
- **Working with output**
  - While MOVES can produce summary reports from the GUI, MySQL can be used to produce customized output

MySQL

- Relational Database Management System (RDBMS)
- What is a “relational database”?

MOVES

Relational Databases 101

- A relational database provides a way of storing, querying, and manipulating large amounts of data
- A database contains multiple tables, in which data are stored. Each table contains records (rows) and fields (columns)
- These tables relate to each other based on values in a particular column

4

Relational Databases 101

- **Example: Excel spreadsheet**
  - Each “spreadsheet” is like a database
  - Each “worksheet” within the spreadsheet is like a table
  - However, the rules for database tables are more rigorous

MOVES

Relational Database Rules

- Each table is a set of columns and rows that describe multiple instances of something
- Each field (column) must contain a unique type of data in a specific format
- Each row represents one unique record (tables cannot contain duplicate records)

6

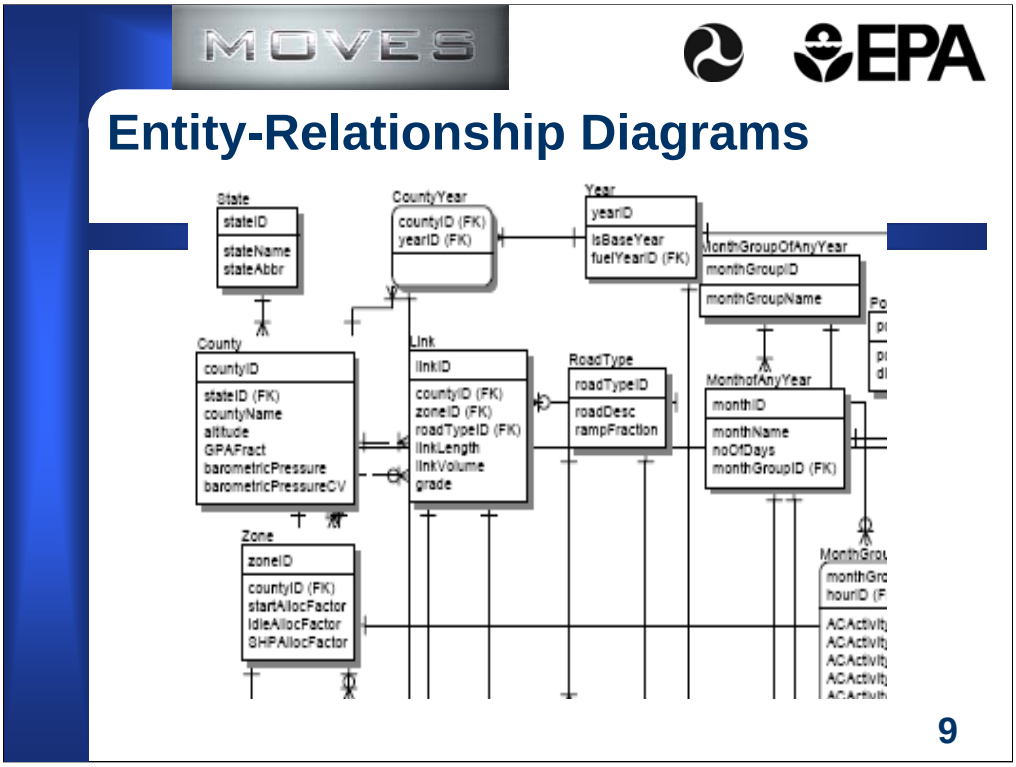
MOVES

Relational Database Rules

- **Each table must contain at least one field with completely unique values to identify a record (row), and so that this record can be related to data in other tables (a key)**
  - Keys are usually numeric for flexibility and speed.
- **NULL values are allowed when data are not known or not relevant. Zeros are not the same as NULL.**
  - Keys cannot be NULL

MOVES Databases

- **The MOVES default database has over 100 different tables that store**
  - Lookup/reference information
  - Conversion/adjustment factors
  - Emissions data
  - Activity data
- **MOVES also uses databases to store user inputs, intermediate results and final output**



The symbols on the connecting lines indicate the type of relationship. The top fields (above the line) in the boxes are the key fields used in joining tables.

MOVES

Types of Relationships

- **One to one**
  - Example: for each month there is only one month code, and for each month code there is only one month
- **One to many**
  - Example: one model year car can be sold in any of 12 months, but during any given month only one model year car is being sold
- **Many to many**
  - Example: any vehicle type can be of any age (any of the 30 age bins)

10

Structured Query Language (SQL)

- **SQL is the language used to work with modern relational databases**
  - It provides the programming syntax (language) to communicate with (query) the database and its tables
- **Like English, an SQL statement is made up of subjects, verbs, and predicates**
  - "SELECT * FROM state WHERE stateid = 32" means "Show me all the data from the "state" table where the state code in the table is 32."

MOVES and MySQL

- **Draft MOVES2009 uses only MySQL version 5.0.27**
  - Newer or older versions will not work.
- **Support for MySQL is available in the help files, the reference manual (C:\MySQL\docs\manual.chm), and the MySQL web site (http://www.mysql.com)**
- **Entity/relationship diagrams for the MOVES default database are located at C:\MySQL\data\(MOVES default database folder)\readme**

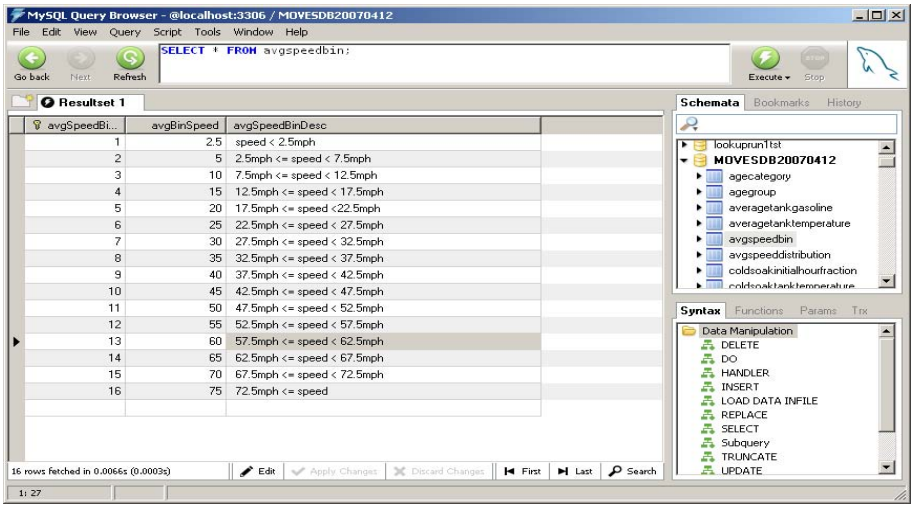
MOVES and MySQL

- **Do you need expertise in relational databases and MYSQL?**
  - Not needed for simple runs
  - Some basic knowledge gives users flexibility to customize outputs and view input databases
  - In-house expert would be helpful for more advanced analysis
- **MySQL training is available from various commercial vendors.**

MySQL Query Browser

- Provided with the MOVES installation package.
- Windows tool for viewing databases, executing queries, and editing tables.
- Resultsets can be exported as csv or Excel files.
- Built-in Help files.
- Records query history, so you can repeat queries without retyping them.
- Tables can be edited directly, rather than using MySQL commands.
  - Table must have a key to be edited directly

MySQL Query Browser



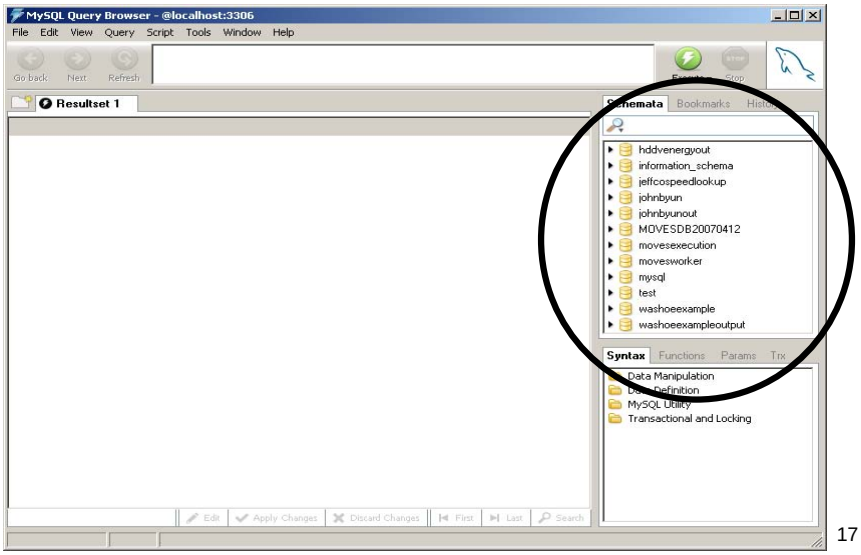
15

The Browser can be used to display the contents of the tables in a spreadsheet-like form.

Exploring MOVES Databases with Query Browser

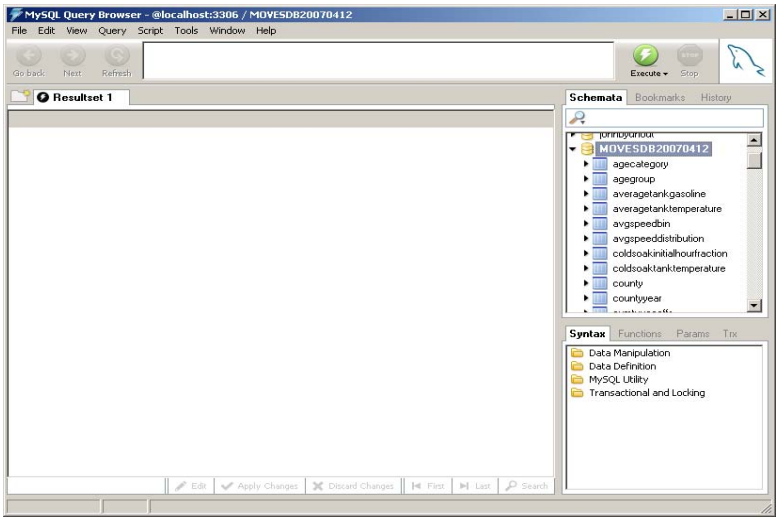
- **We'll demonstrate some simple MySQL commands in Query Browser.**
  - Commands are shown in the slide headings and screenshots.
  - Note that some screenshots show older versions of MOVES and/or output databases you won't have on your computer.
- **To Open Query Browser:**
  - Start/Programs/MySQL/MySQL Query Browser.
  - Make sure "localhost" is specified (might not be after initial installation) and click OK.
    - **Click "ignore" on warning message about schema.**
    - **You can suppress this message for future use.**

MySQL Query Browser opening view with available databases (example)



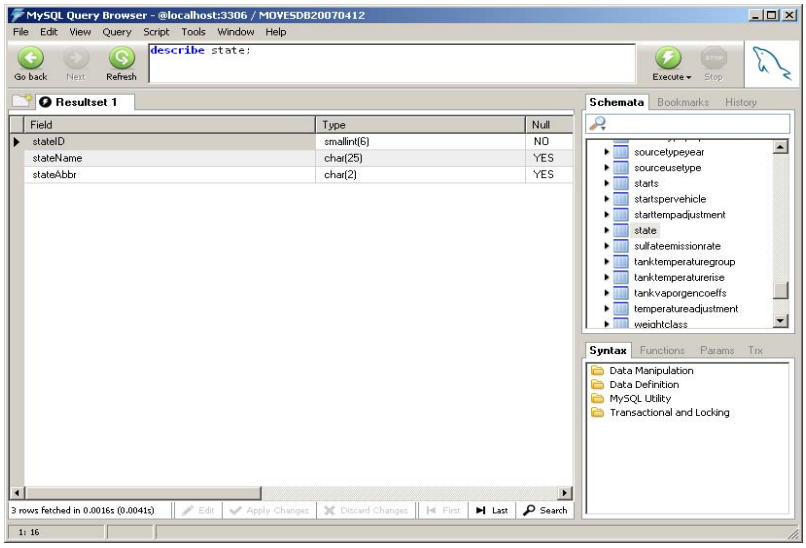
17

Double-clicking on the MOVES default database opens up tables



18

DESCRIBE state;
(or, show columns from state;)

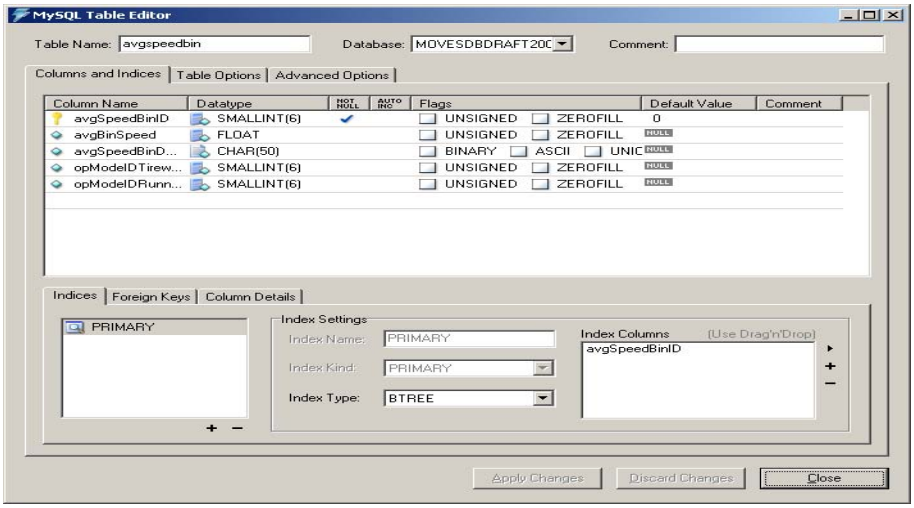


19

Describe command is used to show you the structure of a table.

You can also right click and edit a table, which will display it's structure as well.

Highlighting Table + F2 will also show the table structure.



20

The F2 function key will show the structure of the table.

You can also right click and edit a table, which will display it's structure as well.

MOVES

Commonly Used Field Types in MOVES Databases

- INT (± 8388608)
- SMALLINT (± 32767)
- FLOAT(M,D), (default is (10,2))
- VARCHAR (variable length string between 1 & 255 characters)
- CHAR (fixed length string between 1 & 255 characters)

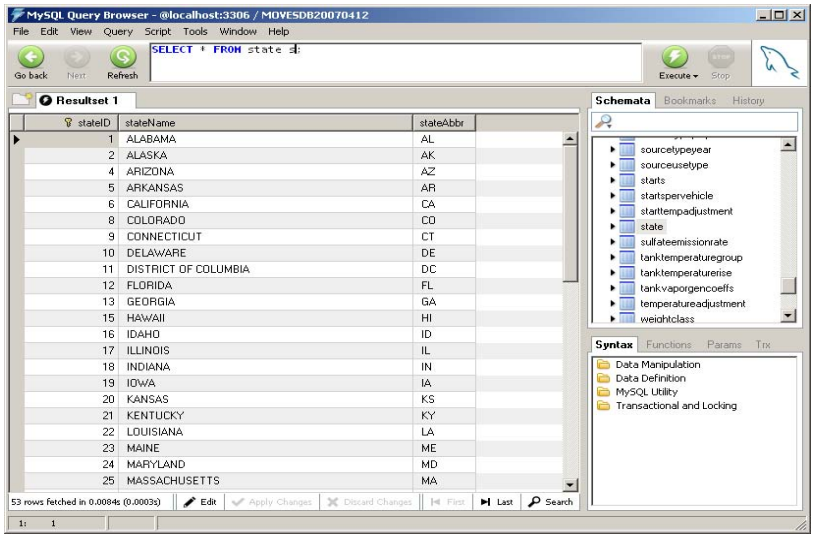
21

MOVES

“SELECT” statements in Query Browser

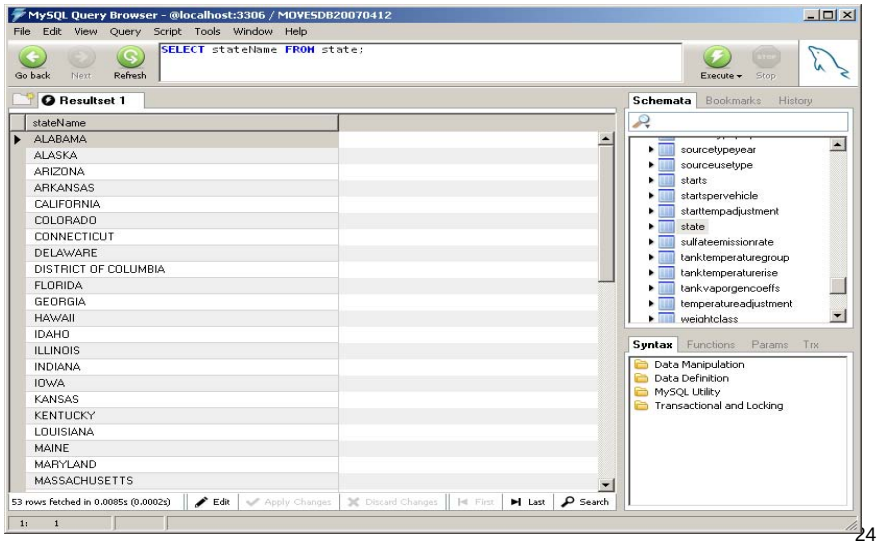
- Double-clicking on table names generates a “SELECT *” (Select All) statement
- Can also drag and drop from the table listing into the results window
- You can include conditions in Select statements to filter data or locate individual data elements

SELECT * FROM state;

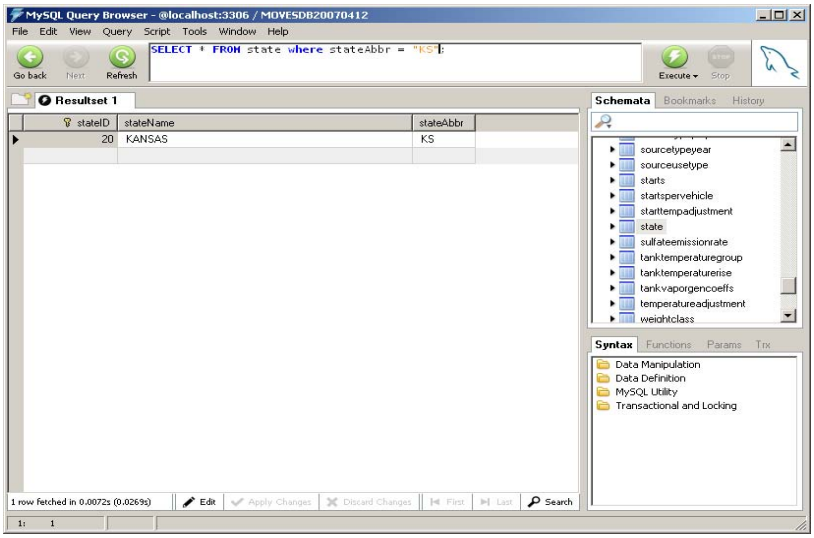


23

SELECT stateName FROM state;

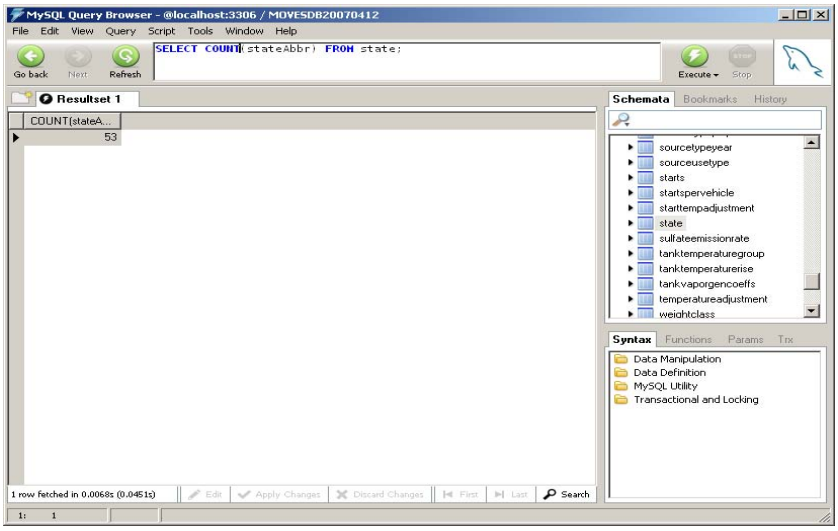


SELECT * FROM state where stateAbbr = "KS";



25

SELECT COUNT(stateAbbr) FROM state;



26

MOVES

Working with MOVES output

- **MySQL commands can be used to sort or sum MOVES output by year, month, pollutant, road type, etc.**
  - Can filter results in MySQL and export to Excel
- **The Output Emissions Detail options can also be used to perform some output aggregation, but once applied, the underlying detail is not saved**
  - MySQL or Excel can be used to aggregate the output after the MOVES run, so that the detail in the native MOVES output is still available if it's needed in the future
  - In some cases, you need aggregation factors (e.g., travel time fractions)—total *emissions* of a given pollutant are additive, but emissions *rates* need to be weighted

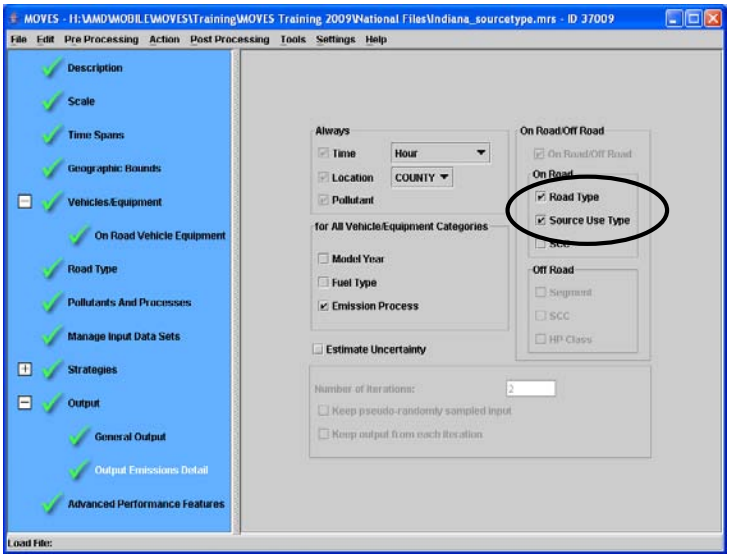
MOVES

Working with output: examples

1) Summing emissions output across roadtypes
2) Filtering MOVES lookup output for desired emissions rates

28

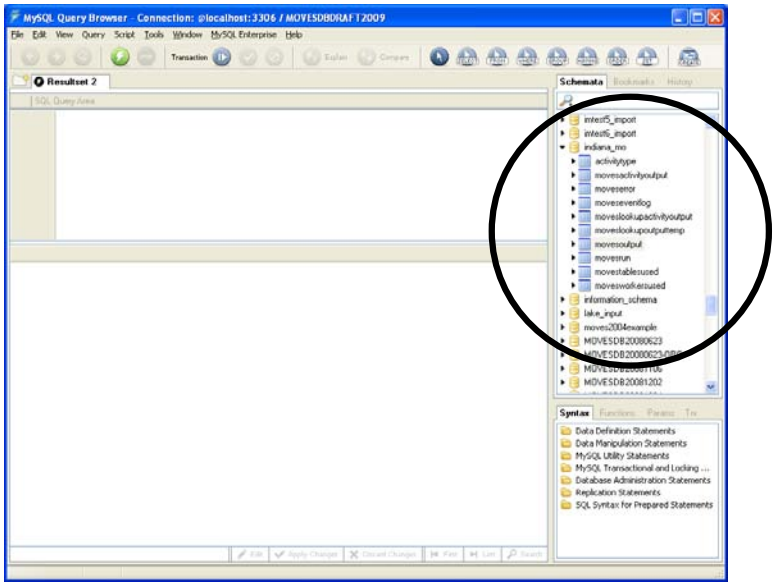
Specifying Detail in Output Emissions Detail screen



29

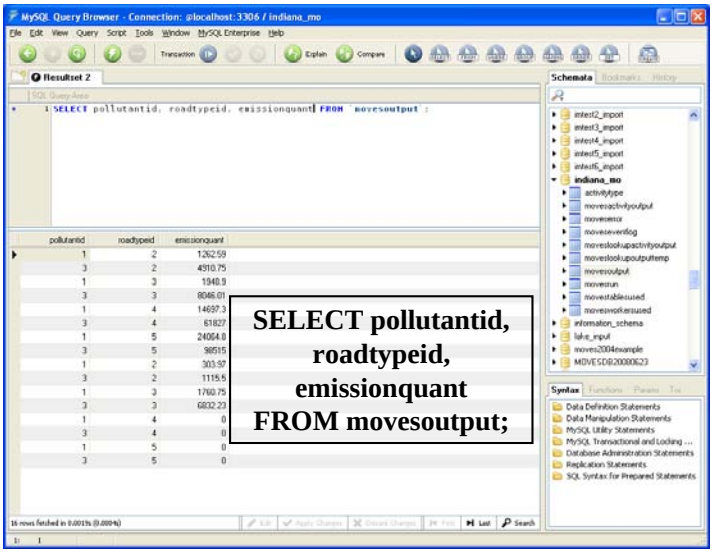
There is only 1 emission process in Indiana.mrs (running), but emission process is selected so that it is identified in the output database rather than being NULL.

MOVES output tables



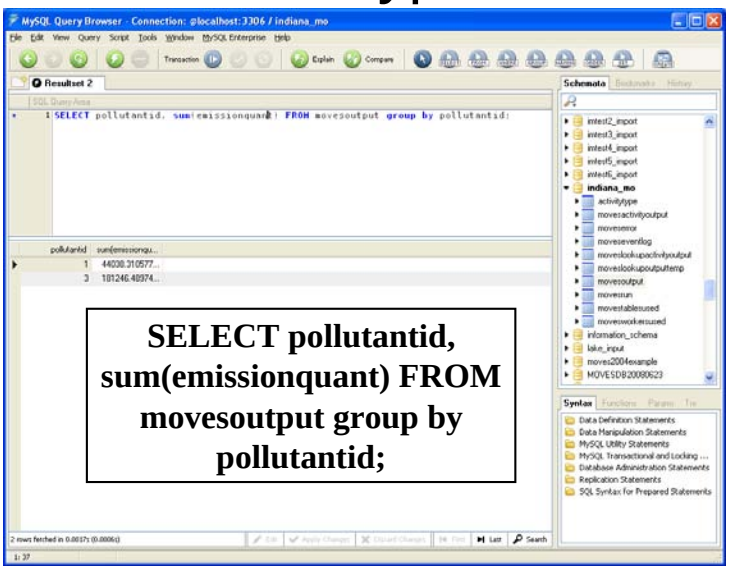
30

Emissions output table, filtered for pollutant, roadtype, emissions



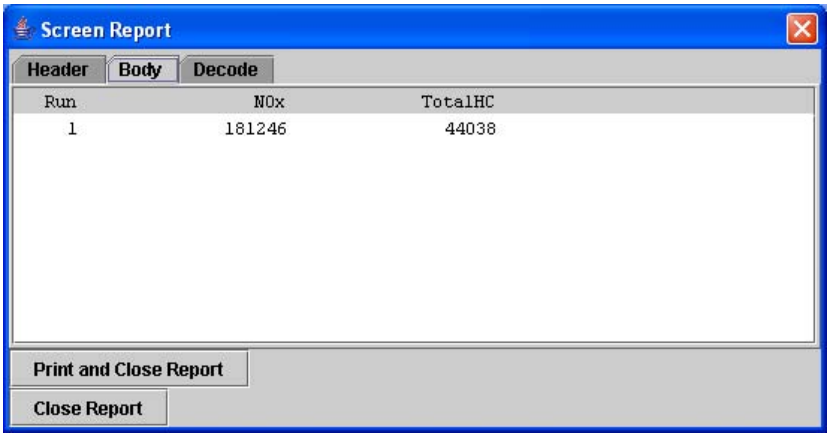
31

Summing emissions across road types



32

MOVES summary report for the same run:



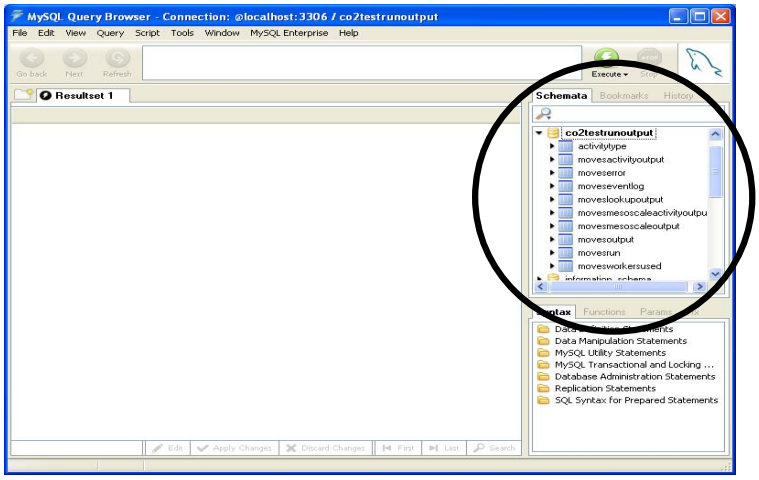
33

MOVES

Other example scripts for summing output

- SELECT MOVESRunID, count(*), sum(distance) FROM MOVESActivityOutput GROUP BY MOVESRunID;
  - (count(*) returns the total number of records summed)
- SELECT MOVESRunID, pollutantID, processID, count(*), sum(emissionQuant) FROM MOVESOutput GROUP BY MOVESRunID, pollutantID, processID;

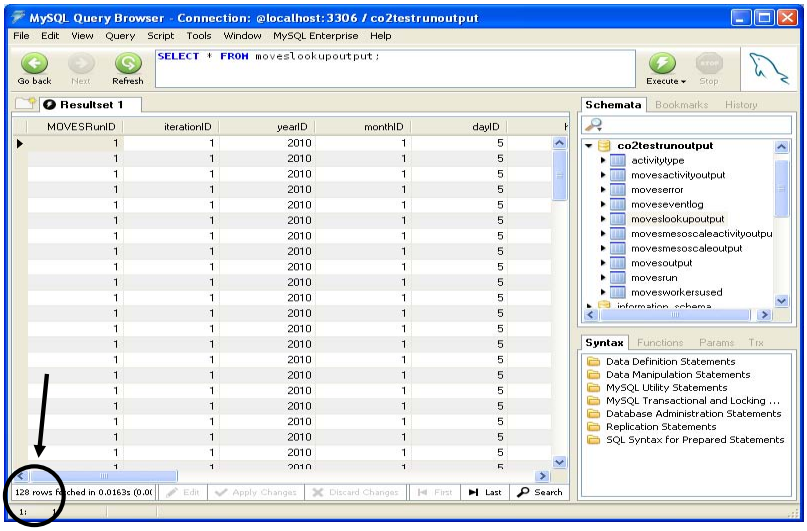
Working with MOVES Lookup Output Tables



35

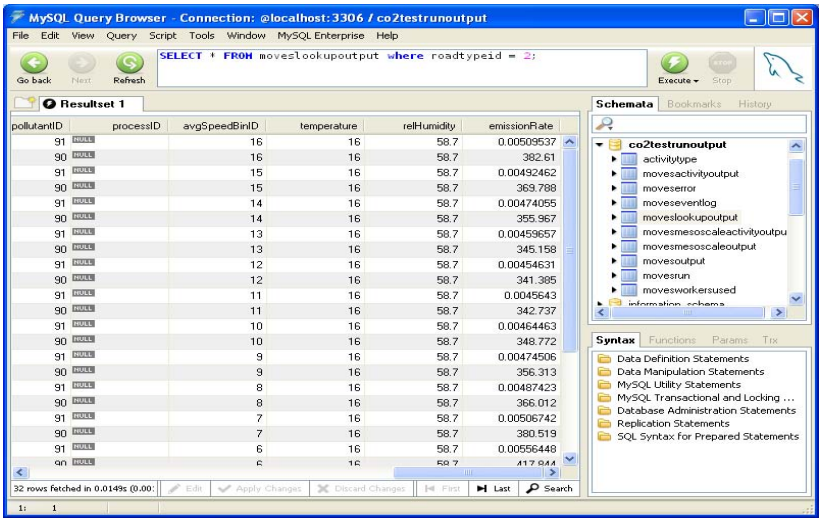
Participants do not have the database used in the following slides.

SELECT * FROM moveslookupoutput; (returns 128 rows in this example)



36

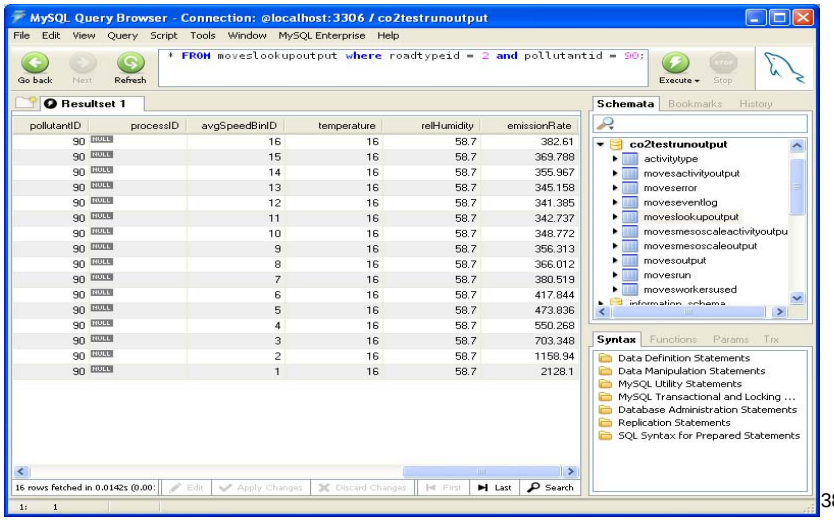
SELECT * FROM moveslookupoutput where roadtypeID = 2; (returns 32 rows)



pollutantID	processID	avgSpeedBinID	temperature	relHumidity	emissionRate
91	NULL	16	16	58.7	0.00509537
90	NULL	16	16	58.7	382.61
91	NULL	15	16	58.7	0.00492462
90	NULL	15	16	58.7	369.788
91	NULL	14	16	58.7	0.00474055
90	NULL	14	16	58.7	355.967
91	NULL	13	16	58.7	0.00459657
90	NULL	13	16	58.7	345.158
91	NULL	12	16	58.7	0.00454631
90	NULL	12	16	58.7	341.385
91	NULL	11	16	58.7	0.0045643
90	NULL	11	16	58.7	342.737
91	NULL	10	16	58.7	0.00464463
90	NULL	10	16	58.7	348.772
91	NULL	9	16	58.7	0.00474506
90	NULL	9	16	58.7	356.313
91	NULL	8	16	58.7	0.00487423
90	NULL	8	16	58.7	366.012
91	NULL	7	16	58.7	0.00506742
90	NULL	7	16	58.7	380.519
91	NULL	6	16	58.7	0.00556448
90	NULL	6	16	58.7	417.944

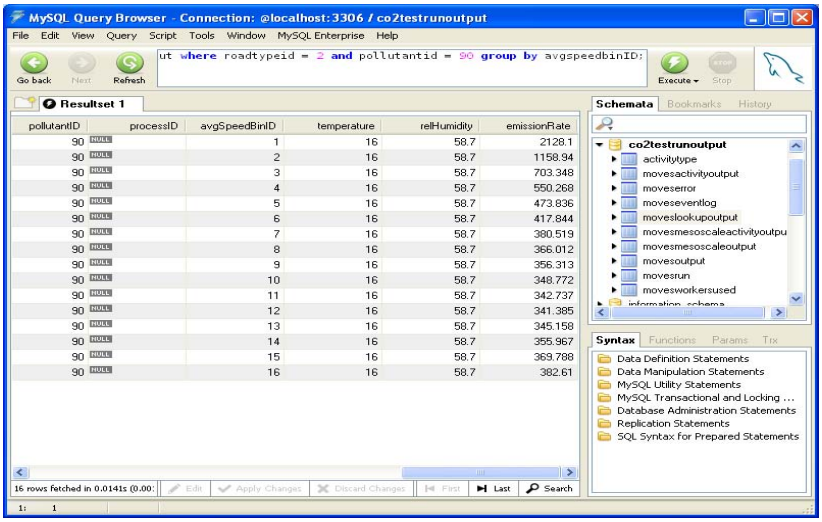
37

SELECT * FROM moveslookupoutput where roadtypeid=2 and pollutantid=90; (Returns 16 rows)



pollutantID	processID	avgSpeedBinID	temperature	relHumidity	emissionRate
90	NULL	16	16	58.7	382.61
90	NULL	15	16	58.7	369.788
90	NULL	14	16	58.7	355.967
90	NULL	13	16	58.7	345.158
90	NULL	12	16	58.7	341.385
90	NULL	11	16	58.7	342.737
90	NULL	10	16	58.7	348.772
90	NULL	9	16	58.7	356.313
90	NULL	8	16	58.7	366.012
90	NULL	7	16	58.7	380.519
90	NULL	6	16	58.7	417.844
90	NULL	5	16	58.7	473.836
90	NULL	4	16	58.7	550.268
90	NULL	3	16	58.7	703.348
90	NULL	2	16	58.7	1158.94
90	NULL	1	16	58.7	2128.1

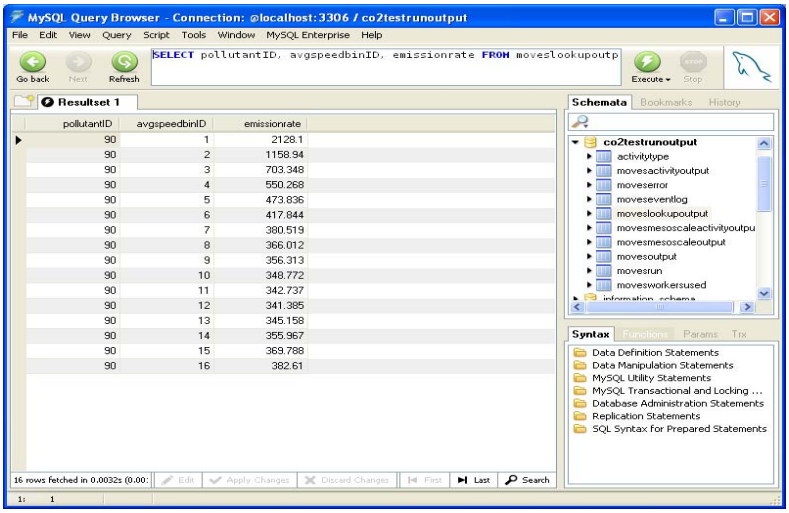
SELECT * FROM moveslookupoutput
where roadtypeID = 2 and pollutantID = 90
group by avgspeedbinID;
(now ordered by speed bin)



pollutantID	processID	avgSpeedBinID	temperature	relHumidity	emissionRate
90	NULL	1	16	58.7	2128.1
90	NULL	2	16	58.7	1158.94
90	NULL	3	16	58.7	703.348
90	NULL	4	16	58.7	550.268
90	NULL	5	16	58.7	473.836
90	NULL	6	16	58.7	417.844
90	NULL	7	16	58.7	380.519
90	NULL	8	16	58.7	366.012
90	NULL	9	16	58.7	356.313
90	NULL	10	16	58.7	348.772
90	NULL	11	16	58.7	342.737
90	NULL	12	16	58.7	341.385
90	NULL	13	16	58.7	345.158
90	NULL	14	16	58.7	355.967
90	NULL	15	16	58.7	369.788
90	NULL	16	16	58.7	382.61

39

SELECT pollutantID, avgspeedbinID, emissionrate FROM moveslookupoutput where roadtypeID = 2 and pollutantID = 90 group by avgspeedbinID;



40

MOVES

Other handy things Query Browser can do

- Export result tables.
- Import data.
- Edit data.
- History function.

41

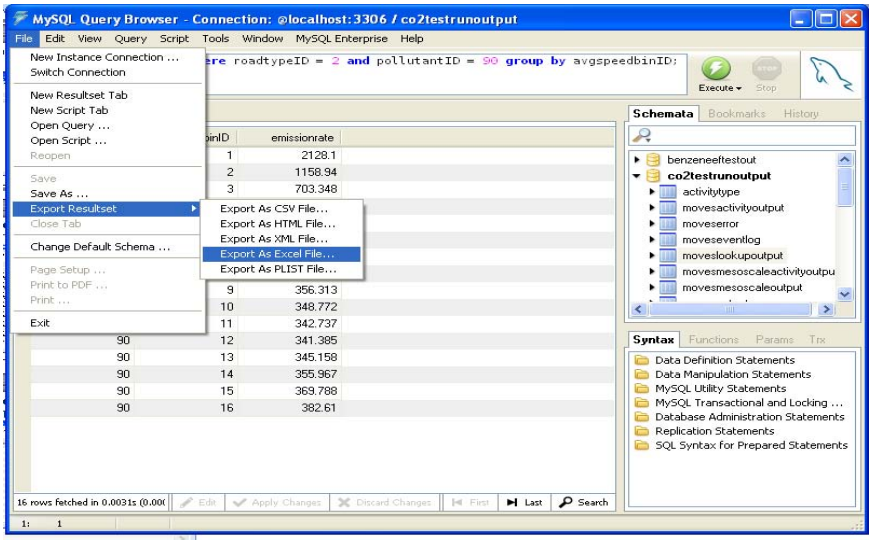
MOVES

Export tables using the Browser

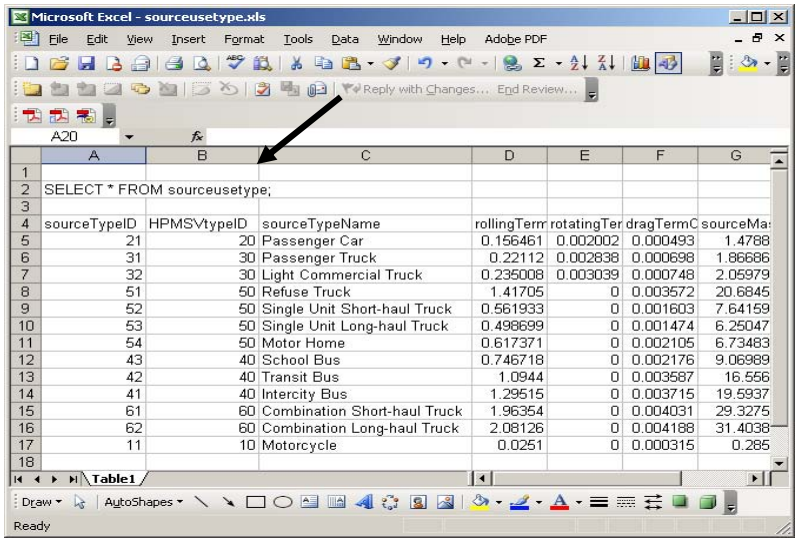
- One useful way to build input databases is to first copy the relevant information from the MOVES default database
- Use SELECT and WHERE statements as necessary to filter the information and get the values that you want
- In Query Browser file menu, select “Export Resultset”—can save as .csv, .xls, or other file formats
- Then you can work with these values in Excel or other programs

42

Export resultsets from Browser



Tip: Copy query statements into exported spreadsheets for future reference



SELECT * FROM sourceusetype;

sourceTypeID	HPMSVtypeID	sourceTypeName	rollingTerm	rotatingTer	dragTermC	sourceMa
21	20	Passenger Car	0.156461	0.002002	0.000493	1.4788
31	30	Passenger Truck	0.22112	0.002838	0.000698	1.86686
32	30	Light Commercial Truck	0.235008	0.003039	0.000748	2.05979
51	50	Refuse Truck	1.41705	0	0.003572	20.6845
52	50	Single Unit Short-haul Truck	0.561933	0	0.001603	7.64159
53	50	Single Unit Long-haul Truck	0.498699	0	0.001474	6.25047
54	50	Motor Home	0.617371	0	0.002105	6.73483
43	40	School Bus	0.746718	0	0.002176	9.06989
42	40	Transit Bus	1.0944	0	0.003587	16.556
41	40	Intercity Bus	1.29515	0	0.003715	19.5937
61	60	Combination Short-haul Truck	1.96354	0	0.004031	29.3275
62	60	Combination Long-haul Truck	2.08126	0	0.004188	31.4038
11	10	Motorcycle	0.0251	0	0.000315	0.285

Load data infile

- **This statement can be used to import information from .csv or .txt (not .xls) files *into* MySQL tables**
- **Example script to update age values:**
  - load data infile 'c:/MOVES/Change/NewAgeDist.csv'
  - Replace into table SourceTypeAgeDistribution
  - fields terminated by ','
  - ignore 1 lines
  - (sourceTypeId,yearId,ageId,ageFraction);

MOVES

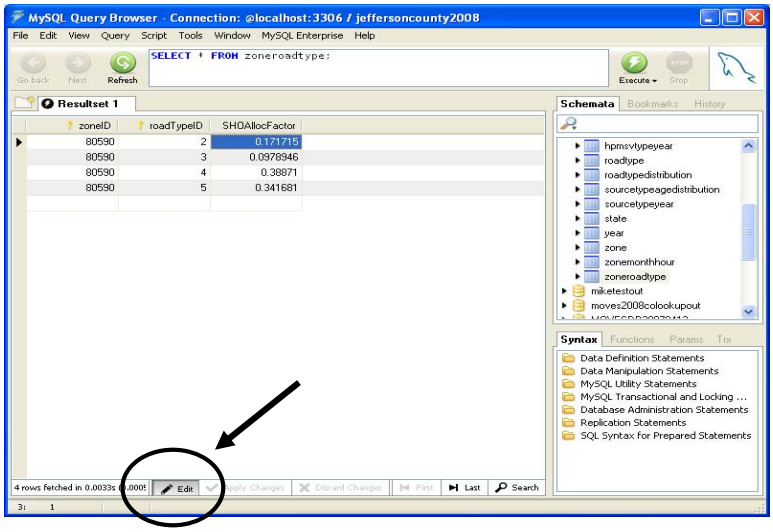
Editing data from within Query Browser

- **Cell values in tables <u>with primary keys</u> can also be edited from within Query Browser itself, rather than importing alternative data:**
  - Highlight a cell
  - Click “Edit” button
  - Make any necessary changes
  - Click “Apply Changes”
  - Click “Edit” button again to close

Editing data from within Query Browser

- **MOVES output tables cannot be readily edited this way, since they do not have primary keys.**
- **Don't edit the default database!!!**
  - Check the title bar at the top of the Query Browser window to make sure that the database you *want* to edit is active

Editing data from within Query Browser



48

Copy default information into new database

- **You can also copy information from the default database directly into another database without first exporting and then importing**
- **This is useful when you want to edit data within Query Browser, and don't have many values to change**
- **Example: insert into myrun.fuelformulation SELECT * FROM MOVESDBDRAFT2009.fuelformulation where fuelformulationid = 1061;**

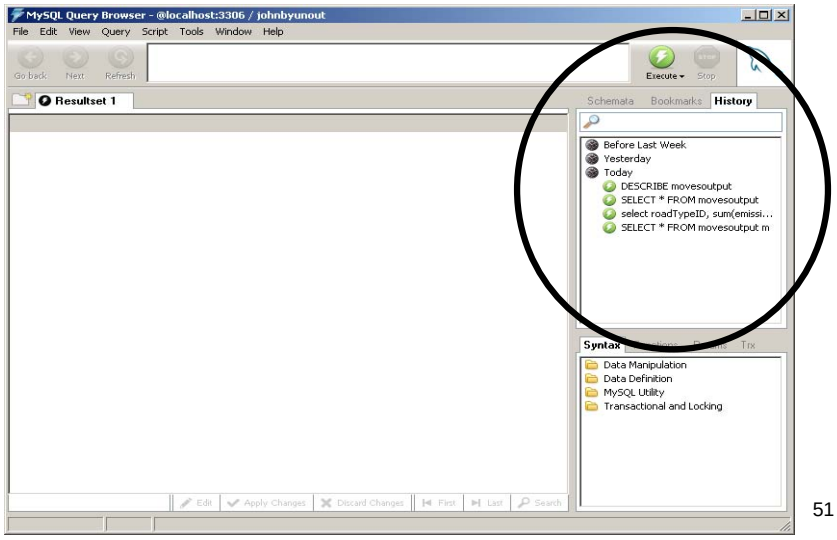
  In this example, "myrun" is the user input database and "MOVESDBDRAFT2009" is the default database (syntax is "databasename.tablename")

MOVES

Using Query Browser's "history" function

- Query Browser has a history function, so that previous statements can be selected and applied, rather than retyping them
- Can also pull up a previous query statement and then modify it to select the desired data

Query Browser history function



51

MOVES

Generating local input databases for MOVES when GUI options don't exist

- MOVES has several GUI options for importing common types of local inputs (County Data Manager, I/M Coverage Editor, etc.)
  - These are covered in other parts of the course
- In some cases, non-default local data would need to be supplied through additional input databases (as opposed to being entered through the GUI)
- Since the tables in these databases must adhere to the MOVES default names and formats, it is safer (and easier) to base input databases on existing tables, rather than creating them from scratch.

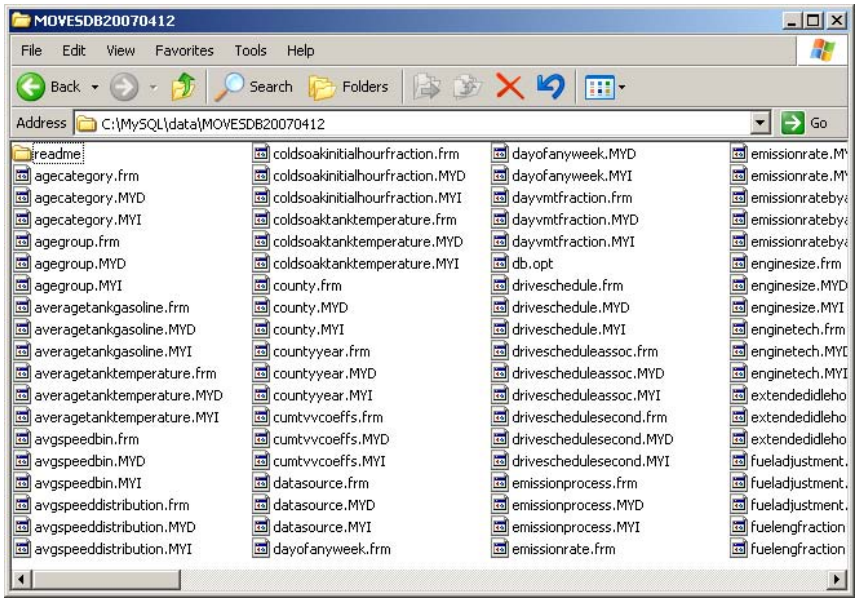
52

MOVES

One way: Copying default tables

- Create a new database in MySQL
- In Explorer, locate the files associated with each necessary table in the MOVES default database, and COPY (don't MOVE) them into the new database folder
- Each table has three associated files (*.frm, *.MYD, and *.MYI), all of which must be copied
- These can then be edited using MySQL commands

53



54

Note that all tables have three parts (frm, myd and myi). All three parts must be copied.

Use the “flush tables;” command to unlock tables that were in use.

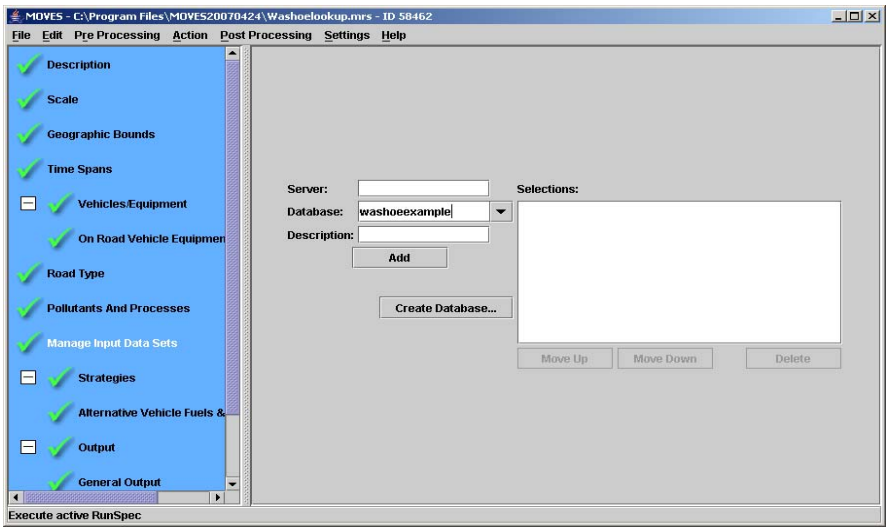
MOVES

Creating a Template Database in MOVES

- A blank database can be created in MOVES from the Manage Input Data Sets screen—this database contains all the tables that the default database does, with the same format, but no data
- This database can then be populated with necessary data

55

Creating input databases



MOVES

Never, Ever, EVER Modify the MOVES Default Database!!!

- **To be safe, ALWAYS create a new database in MOVES, or copy tables from the MOVES default database into a new database, and THEN work on the tables**
  - MOVES will not alert you if you accidentally modify the default database
  - Any changes you make will be applied to all future MOVES runs using this database
  - If you modify the default database, it will also make it extremely difficult to troubleshoot puzzling model results
- **Make a backup copy of the MOVES default database, just in case . . .**