SQLMonitor: SQL Logging within Omnis

Demonstration vof SQL Logging, SQL Breakpoints, Browsing and Filtering of the SQL-Logs via an included tool

Author

Udo Sonnabend
usonnabend@wigasoft.ch

Programming Omnis since 1986
Start with Omnis 3 on Macinthosh Plus

Company

Wigasoft AG
CH-9014 St.Gallen Schweiz
0041 (0) 71 274 51 31
www.wigasoft.ch

- One of the leading software vendors for healthcare in hospitals and homes
- Software runs on Windows platforms
- Studio v10.22
- Fat Client Software
- Framework is a modified Masterstudio Framework by Marc Smit

Summary

An example library SQLMonitor.lbs is used to show how SQL queries can be recorded directly in Omnis. It contains a monitor with which the log can be inspected, filtered and breakpoints can be set.

The library is attached and is intended as a template and source of ideas and can also be modified as desired.

Test Limits:

- Tool has been used for 15 years, almost no changes
- Omnis Studio version 5 to 10
- On Fat-Clients only
- Windows only, rudimentary on Mac
- On MS SQL Server only, a bit on SQLite
- No RPCs
- No Cursors
- Workers mapped only rudimentary

How it works

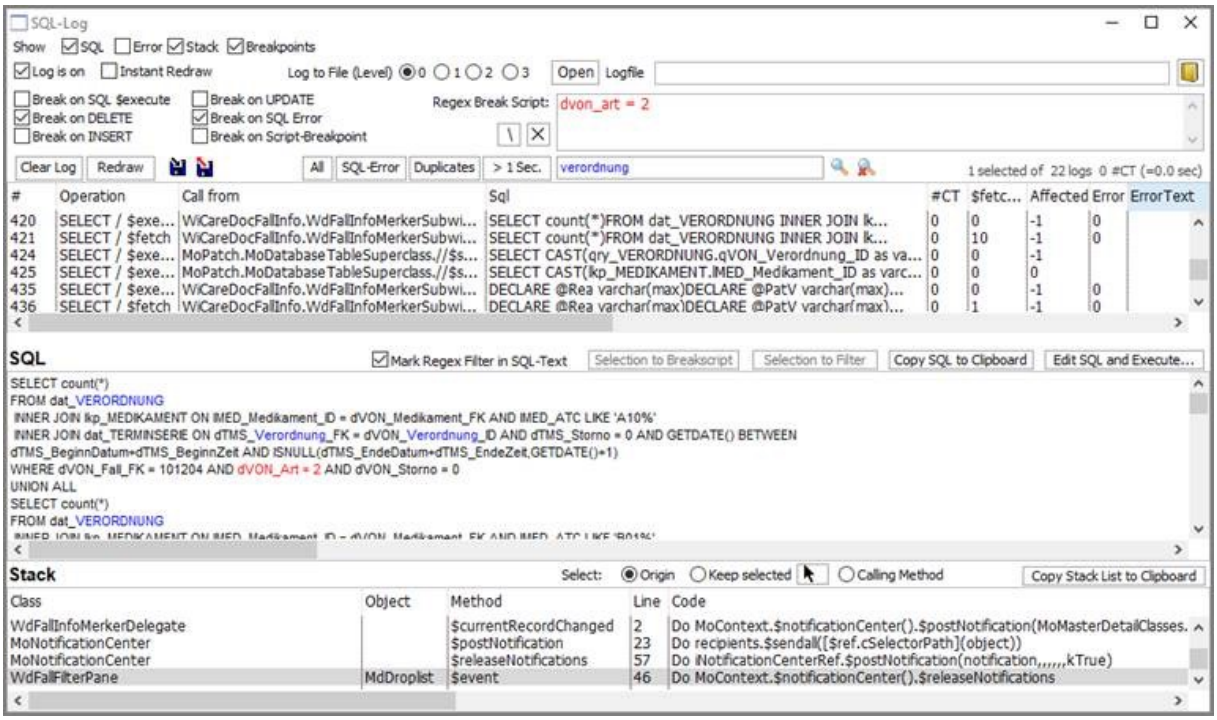
Put the library in the Omnis Startup folder ➔ Menu

- Get the monitor window in the foregrund
- Switch on/off Log
- Delete latest log

- Hold at each command
- Set ‚rough' breakpoints
- Set script breakpoints, regex search

- Customizing of an existing program

Monitor



Areas:

- Display Control
- Set Breakpoint
- Log with filter
- SQL Query
- Error Text
- Call Stack

Display Control

Areas of the monitor window can be faded in and out. The size of the sub-areas can be changed.

- Log list is visible all the time
  - Can be redrawn
  - Can be deleted
  - Can be stored as a file and loaded back
- SQL Text display with labelled filter
  - Label can be switched on and off
  - Query can be edited/tested in a separate editor
- Error: Display with error codes and error text
- Stack: Entire stack at the time of the Query

Logged data

API: Data recorded by the method $addToSQLLog()

- SQL Skript, with replacement of the bind variables by their values, if possible
- Method Stack, at the time oft he log call
- #CT: The duration for the execution of the individual query, fetch
- Number of records concerned
- Errors: error and error text at runtime
- Call From: Legible description of where the enquiry to DB came from (Lib.Class/Method).

The Log/Filter

- Can be filtered:
  - via Regex string (blue)
  - or by special critera, double, slow ones etc.
- Columns can be sorted.
- The log can be saved and reloaded for later comparisons.

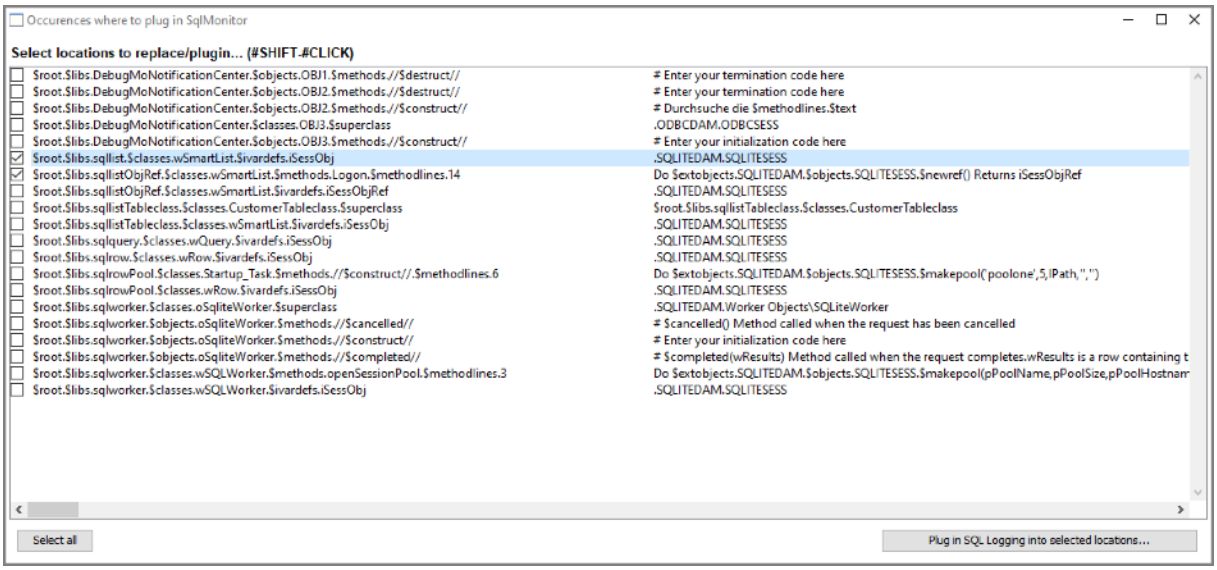
Breakpoints

Execution of the program can be stopped:

- Stop of the program at each SQL operation
- with special commands, e.g. INSERT, UPDATE etc.
- when SQL faults occur
- if the SQL text contains certain keywords (in red) which can be defined via a Regex string.

Plug in SQL Logging into open Libraries...

The tool can be plugged into open libraries at desired locations at your own risk.



The window above shows some IDE SQL samples from Omnis. Locations where logging is to be installed can be selected. The replaced locations are displayed in the Omnis Findlog. Please close Findlog immediately afterwards, see there.

Final Remark

- - This tool is intended to ***outline*** the possibility of how such a log could be built.
- - For the current situation, very large adjustments may have to be made
- - or only use the maintenance of the log via monitor.
- - Plugging in the classes for logging is at your own risk.
- - To log simple lists, the demo also uses unsupported features of Omnis.