



What is SQL4MICRO(SQLFORMICRO)?

SQL4MICRO is a middleware application that sits between a microprocessor/microcontroller and a Sql engine. It runs under windows, makes use of 3.5 NET framework, multithread and multiuser. It's main use is to "buffer" DataBase activity, unloading from the micro the hard job of direct interfacing the DB Engine. For it's capability of substituting the Micro in those tasks, we call it "Delegator" as well.

How does it work?

It listens for commands on a TCP port (soon serial COM port as well). The client, generally a micro, must send the command to our server thru the LAN, WAN, RS232 or RS422.

The command is only 2 bytes followed by a string of parameters, each delimited by a single character of user's choice. We use the pipe (|), but every very rarely used, not alphanumeric char is ok. In every case SQL4MICRO server answers to the request, with a result code and a number of fields each separated by the same character used for the command.

The parameters for the command can have three format: one direct and two indirect.

The direct mode is to send the actual command content.

The first indirect mode makes use of a named strings' repository in our server, so frequently and fixed strings can be recalled directly by the server using their name.

The second allows to send long parameters(like connection strings and SQL strings) in chunks. We handle an array of 1000, 2Mb wide, elements for each micro connected that can be recalled by it's numeric index. As long the micro remains connected, the array is available for repeated uses after we load its content.

The connection with the DBEngine lasts the time needed for the execution of the query and the result is available with no connection, since it's closed on the fly.

There is also the capability to execute command SQL, so the client can create tables and indexes. In this way also row(record) update, add and deletion is possible.

Each client is handled in a separate task/thread.

Each client can use different query at the same time and read the generated subsets using a command with row and a list of columns recalled using 0 relative position or column's name.

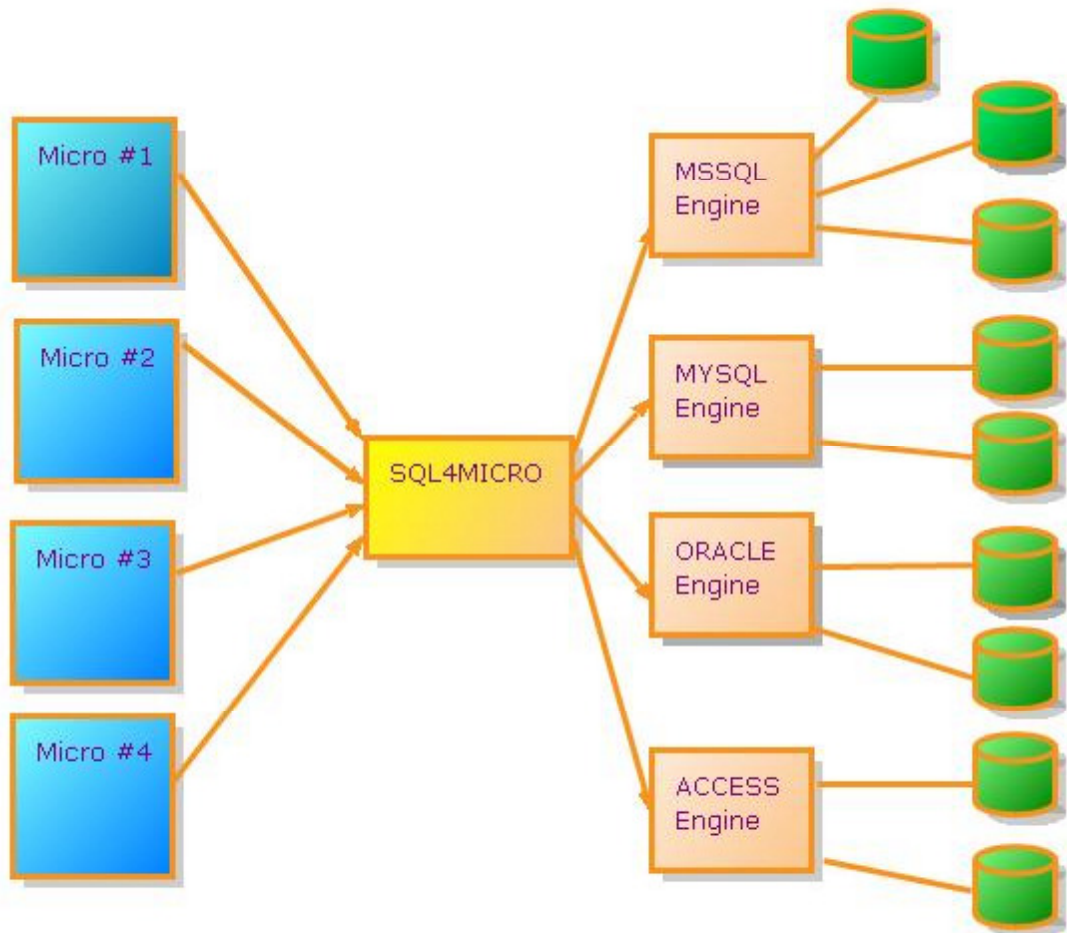
One additional native capability is to handle expressions. A string containing a simple or complex expression can be sent to the server and a result is immediately sent to the client. Data/time can be handled as well. NOW() function returns date&time to the client from the server's clock.

Every operation is logged in separate files for each client, according its IP address and date.

One native capability is to write data in a file, in text format. More files can be written at once.

In brief it's capabilities:

- "Query SQL" execution and result read, with very light memory overhead.
- Execution of "SQL commands".
- Expression parsing and solving.
- LOG of activities divided for client address and date
- Writing of text files, useful for logging data.
- A "user functions' " support will be added at a second time.



What is it for?

Every device which needs centralised data, like settings, could have a vantage with SQL4MICRO. It makes simple to exchange data between devices. Logging and capturing data is made simple and so there is no need to write specialised programs to support different applications or have to maintain two related programs at the same time. The interfaced DataBases could be the same used for general corporate use.

Industrial operations like weighing could be handled by the balance itself.

Time and Attendance Terminals could interface the DB directly, or write the events' record in clear on a text file. Time can be read from the server, for terminal setup.

For informations: info@logicdata.it