**Brief introduction of continuous SQL statement stream processing system**

**Introduction**

Most of client server database systems only support synchronous communication between client and backend database by use of blocking socket. It is believed that this synchronous communication model is much less powerful than asynchronous communication in many aspects such as data transferring efficiency, database fundamental API feature and scalability.

I have developed a world-leading package of secured communication software components written with continuous inline request/result batching, real-time stream processing, asynchronous data transferring, and parallel computation in mind. To demonstrate the power of asynchronous communication, I use popular databases Sqlite and MySQl/MariaDB as samples to implement own database client server transferring system.

This article mainly uses Sqlite and C# demonstration codes for presentation.

**Source codes and samples**

Source codes and samples are located at <https://github.com/udaparts/socketpro>. After cloning it into your computer by GIT, mainly pay attention to the subdirectories umysql and usqlite inside the directory ../socketpro/samples/module\_sample. The first directory contains a set of demonstration MySQL/MariaDB client server applications written with continuous inline request/result batching, real-time stream processing and asynchronous data transferring in mind. Similarly, the second directory is for Sqlite. For details, simply see the sites <https://github.com/udaparts/socketpro/tree/master/samples/module_sample/umysql> and <https://github.com/udaparts/socketpro/tree/master/samples/module_sample/usqlite>.

You can see these samples are created from .NET, C/C++, Java and Python development environments. They can be compiled and run on either Linux or windows platforms.

In regards to SocketPro communication framework, you may refer to its development documentation at ../socketpro/doc/**SocketPro development guide.pdf**

**Asynchronous Sqlite and MySQL/MariaDB are free to you**

It is noted that the two SocketPro server plugins (MySQL:smysql and Sqlite:ssqlite), which are pre-compiled inside directory ../socketpro/bin from UDAPArts, are completely free for you to distribute together with SocketPro client (usocket) and server (uservercore) core libraries. It is planned that an ODBC based plugin will be also created for free in the future.

I wish you can provide us a few suggestions and feedbacks as well as bug report so that these free plugins can be further improved to meet your needs better.

You can compile the two plugins from provided source codes for debugging and learning, but I don’t expect you will distribute your compiled plugin libraries as your compiled plugin libraries will cause an evaluation dialog prompted out at client side randomly.

100% asynchronous data transferring