**A short task for pre-study of SocketPro communication framework**

This is a short guide document on the development of a few Java classes similar to SocketPro adapter for C++. It is expected that you will understand SocketPro communication framework more clearly after completing this short development task.

1. ***Major C++ adapter files, which you mostly like to refer to, are listed in the below.***

* SocketPro C++ adapter files (..\SocketProRoot\include) for server side:

*scloader.h*

*aserverw.h*

*aserverw.cpp*

* SocketPro C++ adapter files (..\SocketProRoot\include) for client side:

*ccloader.h*

*aclientw.h*

*aclientw.cpp*

* SocketPro C++ adapter files (..\SocketProRoot\include) shared by both client and server sides:

*membuffer.h*

*membuffer.cpp*

*ucomm.h*

*commutil.h*

*definebase.h*

1. ***Two important docs ((..\SocketProRoot\doc) are listed in the below:***

*memory queue.pdf*

*tutorial 1 hello world.pdf*

You are required to read through the above two documents. The top one is the most important for you to convert the two C++ classes *SPA::CUQueue* and *SPA::CScopeUQueueEx* into Java classes.

1. ***Four classes to be converted into Java ones. The four classes are:***

*SPA::CUQueue*

*SPA::CScopeUQueueEx*

*SPA::ClientSide::Internal::CClientCoreLoader*

*SPA::ServerSide::Internal::CServerCoreLoader*

Note that the bottom two java classes **must** be implemented with java native interface (JNI) instead of java native access (JNA) for the best interoperation performance between native and Java codes. The java class for *SPA::CUQueue* must be compatible to C++ or C# adapter implementations as described in the doc memory queue.pdf. You must clearly understand how SocketPro deals with object, date time, ASCII string and Unicode string as well as array.

1. ***Implement Java hello world client/server applications as unit test codes***

You are required to implement the Java version of hello world client and server projects. The two compiled java applications must be compatible to existing C++/C# versions of hello world client and server projects whether they are running on either Windows or Linux platforms. In addition, you are highly encouraged to write unit test codes to verify that the Java version of array, object, date time, ASCII string and Unicode string works compatibly with C++/C# implementations.

Note that you don’t have to implement other classes within C++ adapter, but you are definitely able to implement Java version of hello world client and server projects from the above four Java classes if you understand how our C++ adapter works through SocketPro core native libraries.

At last, you are welcome to implement Java versions of other classes within C++ adapter, but they are **optional**. If you have any questions or concerns, please send us a message and your phone number to [support@udaparts.com](mailto:support@udaparts.com)