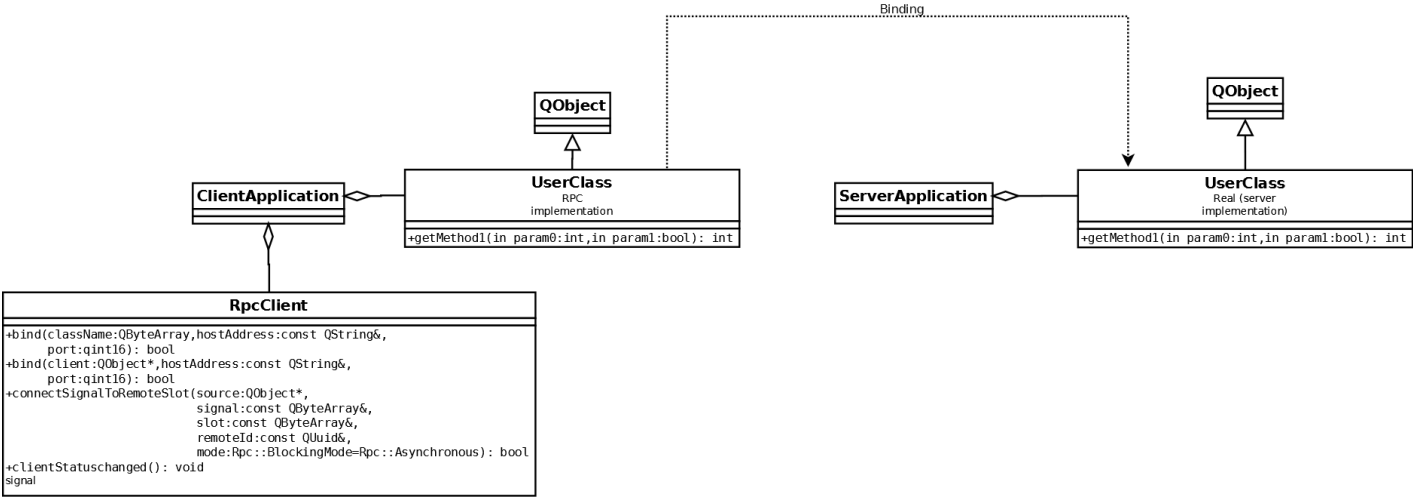


Architecture

The project consists of two libraries, librpcclient and librpcserver which share some base and other classes as well as a tool called roc (Remote Object Compiler). The general usage model is one of client and server, with the client application linking to librpcclient and the remote object instance, i.e. the object on which you wish to remotely invoke methods or connect to slots on, being contained in a server application, running somewhere else on the network or in a different process on the same machine as the client.

The original driver for creating this project was for inter-process communication between QML and Qt C++ applications, in a way that was reusable and portable. The libraries also facilitate distributed applications and other client-server, telemetry and remote control scenarios.

The high level view is shown in UML below. An association between objects of the same type is made between the client and server, for, in this case, RPC binding.



In the signal forwarding use case, the client application may connect signals with a compatible signature from any instance of a local QObject derived class to slots on a remote object which has been bound by the client using the overloaded bind call on [RpcClient](#).