Subject: Re: U++ needs sockets examples and documentation
Posted by lectus on Wed, 28 Dec 2011 16:09:20 GMT
View Forum Message <> Reply to Message

I know the socket API is simple, but what I really mean is how sockets are used with GUI apps.

For example: Do we need threads? How do make the sockets not block the GUI.
Lots of these questions could be answered with docs and examples (of sockets with GUI).

Also, I'd like to propose something to the U++ Dev Team, if it's not already been done.
In Tcl programming language I can use event-driven programming with sockets and it makes it really simple to not block the GUI and also handle multiple requests.
Here's a sample Tcl code that ilustrates that:

Server.tcl

```
# Read below to understand these 2 functions
proc readit {chan} {
    gets $chan data
    puts "We read $data from the socket."
}

proc writeit {chan} {
    # Now we can write something to the socket safely.
    puts $chan "Some data"
}

proc Serve {chan host port} {
   puts "IP $host connected through port $port on channel $chan."
   # Here is the interesting part. We set an event to the socket, and when it's readbale it'll execute the readit function and pass the channel as arg.
   fileevent $chan readable {readit $chan}
   fileevent $chan writable {writeit $chan}
}

# The line below creates a socket server and transfers control to the callback function
socket -server Serve 1234
# The line below waits in a infinite loop (only needed for console app)
vwait forever
```

Now this code could be used in a GUI app and it would not block the GUI through the use of events. It also handle multiples connects without the use of threads.
I don't know if U++ has anything like this, but it's a good feature to have.