

## GTCP Sample

### Installation

The sample was written in Visual Studio 2002.

You need to unpack the provided archive and open the *GTCP 1.0 > Project.sln* solution file. Then rebuild the solution.

Press **F5** to run both client and server in Visual Studio.

### Configuration

Server IP address is specified in the *Client > App.config* file. You should directly modify *Client > Bin > Debug > Client.exe.config* if you need to change the IP address without recompiling the solution.

There is no need to modify broadcast addresses, default values should work in any LAN.

You should change the port if the port 11000 is already in use in your LAN.

### Design

The sample implements a simple chat. Each client chooses its name after a starting, connects to the server, subscribes to the chat event and allows users to send messages to the server, which, in its turn, re-sends messages to all other clients (except the sender).

Known layer contains the declaration of the chat event provider and chat message receivers. Chat room interface is declared as a separate interface; hence this sample can be easily modified for having several chat rooms.

Server provides a business object bound to the known URI. Server implements Chat and Chat Room functionality. Server sends a message to all clients except the message sender. Server enables the usage of IP multicast by means of Broadcast Engine.

Client implements the event receiver interface and subscribes its listener to the event. Client builds up local transparent proxy pointing to the known URI to access server's business object. Client uses several GTCP features such as determining server restarting and it shows Genuine Channels events. If you close the server and start it again, clients re-subscribe to the chat event and continue receiving chat messages.

In addition, client binds its receiver to the chosen court for receiving chat messages via IP multicast.

### Issue

Broadcast Engine is used for managing the event. You can find more information about Broadcast Engine in Programming Guide and in articles dedicated to events:

<http://www.genuinechannels.com/Content.aspx?id=27&type=1>

<http://www.genuinechannels.com/Content.aspx?id=73&type=1>

If a client is able to receive message via IGMP, it starts receiving them via IGMP automatically. If a client is unreachable for IP multicast, it receives chat messages via IP multicast. You do not care about this in your applications, the best option is chosen by Broadcast Engine automatically.