

UDPwriter

General

Sends a single UDP packet. Choose between various Quest3D data types (Text, Value, Vector, Matrix, and Buffer) as input.

Children

0. IN recipient IP (text, default 127.0.0.1)
1. IN recipient port (value, default 5000)
2. IN source data type selector (value): 0 Text (default), 1 Value, 2 Vector, 3 Matrix, 4 Buffer)
3. IN data source (Text)
4. IN data source (Value)
5. IN data source (Vector)
6. IN data source (Matrix)
7. IN data source (Buffer)

Description

Reads recipient IP address, recipient port and selector value. Prepares a UDP header. Tries to send a copy of the content of the selected data child as UDP packet to the recipient. Returns with number of bytes send or an error code <0. Trial time expiration returns a -2. Other error codes according to Windows socket error specification: <http://msdn.microsoft.com/en-us/library/ms740668>

Tips

- Multiple channels in one application are OK.
- Broadcasting may be used, e.g. directed broadcast IP address X.X.X.255 with standard network mask 255.255.255.0.
- Value, vector and matrix data structures follow DirectX conventions, little endian. No network byte order translation.
- Make sure to call UDPwriter not too often, i.e., slow down with SystemCommand → sleep or If's.
- Handle firewalls beforehand (application may crash during firewall dialogs).
- Buffer can be filled with any content (e.g. SaveChannelToBuffer). A typical usage employs the plug-in OSCmessageEncoder.
- Be aware of UDP service limitations (limited packet size, unreliable)
- Make sure that both sides of the communication use the same data type setting.
- V 5.0 support for x64 only.

Legal note: Permission granted for evaluation and educational purposes only (trial version), commercial use requires an explicit license agreement (full version).

Volume, runtime duration, and expiration restrictions are enforced with trial licenses. Details can be found by reading the output of the Debug Window, and on my web page. Be prepared to cope with this situation in your application.

Contact: quest3d.godbensen.eu

Supported Quest3D Versions:

- 4.3.2
- 5.0 (x64 only), 32 on request

Known Problems:

- none

Revision History:

2012 10 12 minor changes, V 5.0 support, consistent data type numbering
2011 02 01 initial release

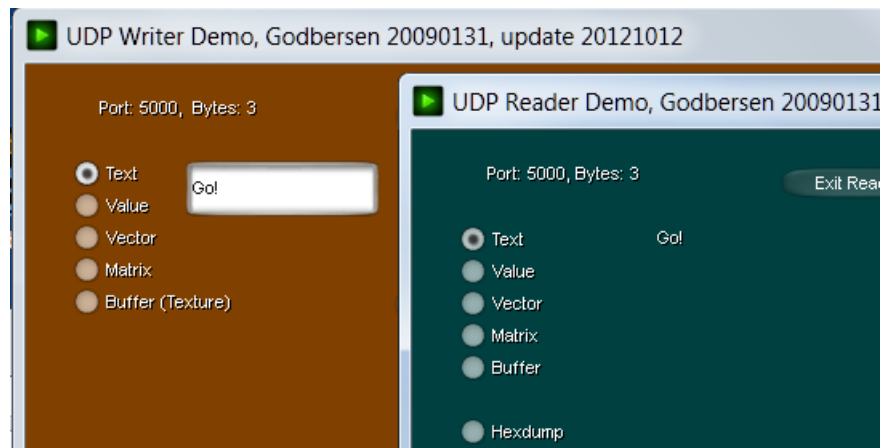
Sample Scenario:

1. UDP Transmission Demo

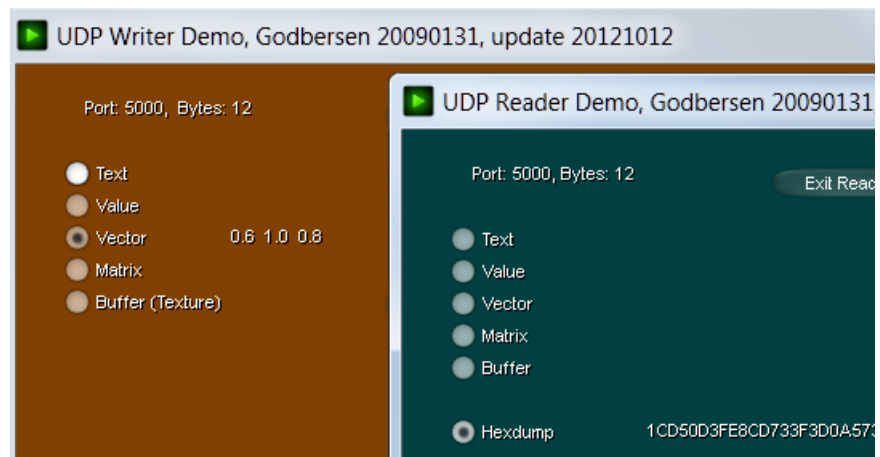
Sample application showing one way transmission (unicast or broadcast) of five different data structures. Start both applications and follow online instructions.

Files: UDPwriterApp500.cgr, UDPreaderApp500.cgr

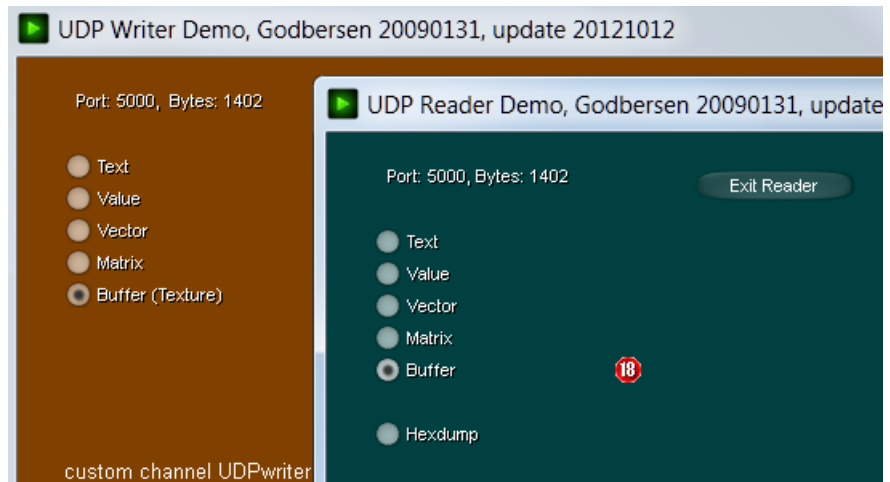
Text 3 Character



Vector (12 Byte), here interpreted as Hexdump



Buffer (1402 Byte),
here used for Texture
transmission



2. UDP Chat Broadcast Demo

Demo of a sample "chat" application. It employs bidirectional transfer of text between many users on different machines in one broadcast domain.

Usage: Adapt IP addresses in source file to your needs. Start same application on different machines.

File: UDPchatBroadcastApp432.cgr

Tutorial:

For testing purposes I recommend this small utility
http://www.hw-group.com/products/hercules/index_en.html