

Forum:[Hack Forums](#)

Thread: [DSTD.Network - New, high level networking library - Async - Open source - Beta](#)

hgdfhsdgdgdfgh - December 18, 2011, 5:20 am

Even though this is part of my DSTD library, I decided to create new topic because this is pretty big library.

This is still in beta: its missing some features, existing features may or may not work correctly. But because this is open source, feel free to modify it.

Features

- NET 2.0 C#
- Open source
- Event driven
- TCP
- High level, use Packet classes to represent data
- Easy to use

If you would like to get some feature, please tell

Example server&client

Protocol definition class.

All subclasses of your protocol can be used as packets (if they derive from Packet abstract class)

Code internal class MyProtocol : DSTD.Network.Protocol

```
{

    public class MyPacket : DSTD.Network.Packet
    {
        public string data;

        public override void CreateFromStream(System.IO.BinaryReader streamReader)
        {
            data = streamReader.ReadString();
        }

        public override void SerializeToStream(System.IO.BinaryWriter streamWriter)
        {
            streamWriter.Write(data);
        }
    }
}
```

Server class

Code internal class MyServer : DSTD.Network.EasyPacketBasedTcpServer<MyProtocol>

```
{
    public MyServer() : base(10021)
    {

    }
}
```

```
protected override void OnConnectionReceived(DSTD.Network.EventDrivenTcpPacketConnection
connection)
{
    MessageBox.Show("Server: Received connection, sending hello world");
    connection.SendPacket(new MyProtocol.MyPacket()
    {
        data = "Hello, World!"
    });
}
```

```
protected override void OnDisconnection(DSTD.Network.EventDrivenTcpPacketConnection
connection)
{
    MessageBox.Show("Server: Client disconnected");
}
```

```
protected override void OnPacketReceived(DSTD.Network.EventDrivenTcpPacketConnection
connection, DSTD.Network.Packet packet)
{
    if (packet is MyProtocol.MyPacket)
    {
        MessageBox.Show("Received MyPacket, data: " +
(packet as MyProtocol.MyPacket).data);
    }
}
```

Client (uses TcpClient to connect, I dont think its necessary to create own class for connecting)

```
Code System.Net.Sockets.TcpClient client = new System.Net.Sockets.TcpClient();
client.Connect("localhost", 10021);
```

```
DSTD.Network.EventDrivenTcpPacketConnection connection =
new DSTD.Network.EventDrivenTcpPacketConnection(
client.Client, new MyProtocol());
connection.PacketReceived += connection_PacketReceived;
connection.EnableReceive();
```

..

```
void connection_PacketReceived(object sender, DSTD.Network.PacketReceivedEventArgs e)
{
    MessageBox.Show("Client: got packet, sending reply\r\n");

    ((DSTD.Network.EventDrivenTcpPacketConnection) sender).SendPacket(
new MyProtocol.MyPacket()
{
    data = "Hello to you too"
});
    ((DSTD.Network.EventDrivenTcpPacketConnection) sender).Disconnect();
}
```

Download

Go to DSTD: <https://hackforums.net/showthread.php?tid=2020379> and download & build source code

Deceptive - December 18, 2011, 7:23 am

Hm, let me see if I can get a chat server going, I messed with DragonHunter's NEX, and Omni's "Nexus" networking libraries. :3

hgdfhsdgdgdfh - December 18, 2011, 8:13 am

I just created a small network sample project (TimeProtocol, TimeServer & TimeFetcherTask). Its included in source control on codeplex.

Banks? - December 18, 2011, 9:26 am

(Dec 18, 2011 08:23 AM)Deceptive Wrote: [\[CITE\]](#) Hm, let me see if I can get a chat server going, I messed with DragonHunter's NEX, and Omni's "Nexus" networking libraries. :3
Omni's "Nexus"? I think you mean, Aeonhack's "Omni".

Deceptive - December 18, 2011, 9:33 am

(Dec 18, 2011 10:26 AM)?anksy Wrote: [\[CITE\]](#) Omni's "Nexus"? I think you mean, Aeonhack's "Omni".

No "Omni" is the whole library, Nexus is the networking part, i rite u wong