

HowTo NNTP in C#

by Randy Charles Morin

This is the third in a series of articles on Internet programming with Microsoft's new C# programming language. In the first two articles, I wrote two simple TCP/IP classes for SMTP and POP3 clients. In this article, I'm going to write a simple NNTP class.

NNTP is an older fading protocol in the Internet protocol family. The protocol is used to retrieve news from news server, a.k.a. NetNews servers. The protocol works by posting messages into various forums, a.k.a. newsgroups. Then other end-users can read the recent posts in the forums. There also exist protocols for distributing NetNews contents amongst various NetNews servers, allowing thousands of servers to share news and forums. The most popular news server is of course Microsoft's [nntp://news.microsoft.com]. More often than not, you can launch your NetNews client by typing the nntp URL in your browser's address bar.

Listing Error! Bookmark not defined.: Exception Class

```
public class NntpException : System.ApplicationException
{
    public NntpException(string str)
        :base(str)
    {
    }
};
```

I'm still unsure how best to implement exception classes in dot-NET and as such I've remained faithful to my C++ roots. I'm investigating otherwise and might consider writing a brief article on just this subject. We'll see.

Next step is the class declaration. I'm deriving the Nntp class from the TcpClient class in the System.Net.Sockets namespace of the dot-NET framework.

Listing Error! Bookmark not defined.: Nntp Class Declaration

```
public class Nntp : System.Net.Sockets.TcpClient
```

We'll inherit a lot of basic functionality from the TcpClient class.

The first method of our Nntp client class is the Connect method. This method takes a server name that represents the remote NNTP server that will service our requests.

Listing Error! Bookmark not defined.: Connect Method

```
public void Connect(string server)
{
    string response;

    Connect(server, 119);
    response = Response();
    if (response.Substring( 0, 3) != "200")
    {
        throw new NntpException(response);
    }
}
```

We call the Connect method of our base TcpClient class with the server name and port 119. Port 119 is the well-known port for NNTP servers. The server should respond with a 200 status-code indicating that connection was successful.

When you are finished calling methods to the Nntp client object, then you should call the Disconnect method to terminate the connection.

Listing Error! Bookmark not defined.: Disconnect Method

```
public void Disconnect()
{
    string message;
    string response;

    message = "QUIT\r\n";
    Write(message);
    response = Response();
    if (response.Substring( 0, 3) != "205")
    {
        throw new NntpException(response);
    }
}
```

The method will send a QUIT message to the server and the server should respond with a 205 status-code indicating that the it is disconnecting the socket.

When you first instantiate the Nntp object you should call the Connect method and when you are finished you should call the Disconnect method. In between, you can call three method, GetNewsgroups, GetNews and Post, to receive and send news to the NNTP server.

The GetNewsgroups method, receives from the NNTP server all the forums that are supported by the server.

Listing Error! Bookmark not defined.: GetNewsgroups Method

```
public ArrayList GetNewsgroups()
{
    string message;
    string response;

    ArrayList retval = new ArrayList();

    message = "LIST\r\n";
    Write(message);
    response = Response();
    if (response.Substring( 0, 3) != "215")
    {
        throw new NntpException(response);
    }

    while (true)
    {
        response = Response();
        if (response == ".\r\n" ||
            response == ".\n")
        {
            return retval;
        }
        else
        {
            char[] seps = { ' ' };
            string[] values = response.Split(seps);
            retval.Add(values[0]);
            continue;
        }
    }
}
```

The GetNewsgroups method begins by sending a LIST message to the NNTP server. The NNTP server will respond initially with the 215 status-code indicating that it successfully received the LIST message. Then the NNTP server will respond with a series of lines, each representing one forum on the NNTP server. After all the forums are sent, the NNTP server will send one line with a single period, indicating the end of the forum list. The list of forums is returned from the GetNewsgroups method as an ArrayList of strings.

From the list of forums, you can select one forum and receive from the GetNews method all the news for that forum. Call GetNews passing the name of the forum to receive the news postings.

Listing Error! Bookmark not defined.: GetNews Method

```
public ArrayList GetNews(string newsgroup)
{
    string message;
    string response;

    ArrayList retval = new ArrayList();

    message = "GROUP " + newsgroup + "\r\n";
    Write(message);
    response = Response();
    if (response.Substring( 0, 3) != "211")
    {
        throw new NntpException(response);
    }

    char[] seps = { ' ' };
    string[] values = response.Split(seps);

    long start = Int32.Parse(values[2]);
    long end = Int32.Parse(values[3]);

    if (start+100 < end && end > 100)
    {
        start = end-100;
    }

    for (long i=start;i<end;i++)
    {
        message = "ARTICLE " + i + "\r\n";
        Write(message);
        response = Response();
        if (response.Substring( 0, 3) == "423")
        {
            continue;
        }
        if (response.Substring( 0, 3) != "220")
        {
            throw new NntpException(response);
        }

        string article = "";
        while (true)
        {
            response = Response();
            if (response == ".\r\n")
            {
                break;
            }

            if (response == ".\n")
            {
                break;
            }
        }
    }
}
```

```

        if (article.Length < 1024)
        {
            article += response;
        }
        retval.Add(article);
    }
    return retval;
}

```

The GetNews method sends a GROUP message to the NNTP server. The NNTP server will respond with a status-code of 211, indicating success, the numbers of articles in the forum present on the server, the lowest message number for an article in the forum and the highest message number for an article in the forum. The method then repeatedly sends an ARTICLE message requesting each article between the lowest and highest message numbers. The NNTP server responds with a 423 status-code if the article is not present on the server and a 220 status-code if the article is present. When we receive a 423 status-code, then we skip to the next message number. When we receive a 220 status-code, the status line is followed by the content of the message and terminated with the now familiar with line with only one period. As the articles are received they are placed into an ArrayList object and returned from the GetNews function once all messages are received.

Finally, the Post method is used to add new articles to the forums. The post method takes four parameters; the newsgroup name, the subject, the sender and the body of the message.

Listing Error! Bookmark not defined.: Post Method

```

public void Post(string newsgroup, string subject, string from,
string content)
{
    string message;
    string response;

    message = "POST " + newsgroup + "\r\n";
    Write(message);
    response = Response();
    if (response.Substring( 0, 3) != "340")
    {
        throw new NntpException(response);
    }

    message = "From: " + from + "\r\n"
        + "Newsgroups: " + newsgroup + "\r\n"
        + "Subject: " + subject + "\r\n\r\n"
        + content + "\r\n.\r\n";
    Write(message);
    response = Response();
    if (response.Substring( 0, 3) != "240")
    {
        throw new NntpException(response);
    }
}

```

The Post method sends a POST message to the NNTP server. The POST message takes the newsgroup names as its only parameter. The NNTP server should respond with a 340 status-code indicating that you may post. The headers and content of the post can then be sent to the server with a terminating single line containing one period. If the message is received correctly, the NNTP server will respond with a 240 status-code.

Our public methods used two private methods, Write and Response. The Write method sends a string of bytes to the NNTP server.

Listing Error! Bookmark not defined.: Write Method

```
private void Write(string message)
{
    System.Text.ASCIIEncoding en = new System.Text.ASCIIEncoding() ;

    byte[] WriteBuffer = new byte[1024] ;
    WriteBuffer = en.GetBytes(message) ;

    NetworkStream stream = GetStream() ;
    stream.Write(WriteBuffer,0,WriteBuffer.Length);

    Debug.WriteLine("WRITE:" + message);
}
```

It is important that we convert the dot-NET string to a series of bytes using ASCIIEncoding before it is sent to the server. I also call the Debug.WriteLine method to send the output string to the debug console to help with debugging this component.

The second private method is the Response method. The Response method receives one line of input from the NNTP server.

Listing Error! Bookmark not defined.: Response Method

```
private string Response()
{
    System.Text.ASCIIEncoding enc = new System.Text.ASCIIEncoding();
    byte []serverbuff = new Byte[1024];
    NetworkStream stream = GetStream();
    int count = 0;
    while (true)
    {
        byte []buff = new Byte[2];
        int bytes = stream.Read( buff, 0, 1 );
        if (bytes == 1)
        {
            serverbuff[count] = buff[0];
            count++;

            if (buff[0] == '\n')
            {
                break;
            }
        }
        else
        {
            break;
        }
    };

    string retval = enc.GetString( serverbuff, 0, count );
    Debug.WriteLine("READ:" + retval);
    return retval;
}
```

After receiving the single line of bytes, the data is translated into a dot-NET string from its ASCII encoded bytes.

Using our NNTP class is quite trivial. An example follows.

Listing Error! Bookmark not defined.: Usage

```
static void Main(string[] args)
{
    try
```

```
{
    Nntp obj = new Nntp();
    obj.Connect("news.devx.com");
    ArrayList list = obj.GetNewsgroups();
    foreach (string newsgroup in list)
    {
        System.Console.WriteLine("Newsgroup :{0}",
            newsgroup);
    }
    list = obj.GetNews("test");
    foreach (string article in list)
    {
        System.Console.WriteLine("{0}", article);
    }
    obj.Post("test", "Hello",
        "randy@kbcafe.com (Randy Charles Morin)", "Goodbye");
    obj.Disconnect();
}
catch ( NntpException e )
{
    System.Console.WriteLine(e.ToString());
}
catch ( System.Exception )
{
    System.Console.WriteLine("Unhandled Exception");
}
}
```

Instantiate an Nntp object and call Connect passing the NNTP server name. Then we can repeatedly call GetNewsgroups, GetNews and Post to receive and post articles in the forums found on the NNTP server. Finally you call the Disconnect method to release the socket connection to our NNTP server.

The RFC for NNTP is RFC 977 and can be found at the IETF website [<http://www.ietf.org/rfc/rfc0977.txt?number=977>]. You should also review the RFC for the USENET news message format as presented in RFC 850 [<http://www.ietf.org/rfc/rfc0850.txt?number=850>].

About the Author

Randy Charles Morin is the Chief Architect of SportMarkets Development from Toronto, Canada and lives with his wife, Bernadette and two kids, Adelaine and Brayden in Brampton, Canada. He is the author of the KBCafe.com website [<http://www.kbcafe.com>], many programming books and many articles.