MonoTorrent - Bug # 376: Peer.Equals(other) compares only IP, not ports

Status:	Closed	**Priority:**	Normal
Author:	Alexander Bogdanov	**Category:**	
Created:	05/16/2010	**Assigned to:**	
Updated:	10/03/2010	**Due date:**	

Subject: Peer.Equals(other) compares only IP, not ports

Description: There was:

```
@// FIXME: Don't compare the port, just compare the IP
 if (string.IsNullOrEmpty(peerId) && string.IsNullOrEmpty(other.peerId))
 return this.connectionUri.Host.Equals(other.connectionUri.Host);@
```

which (to my mind) should be

```
@ if (string.IsNullOrEmpty(peerId) && string.IsNullOrEmpty(other.peerId)){
return this.connectionUri.Host.Equals(other.connectionUri.Host) &&
      this.connectionUri.Port == other.connectionUri.Port;
} @
```

so there was difference between peers, as they can be behind NAT, or it could be 2 instances of torrent-client on one PC.

History

05/16/2010 05:57 PM - Alexander Bogdanov

That's in MonoTorrent.Client\Peers\Peer.cs file @ line 154

(by the way, can't see a way to edit bug, so sorry for code-highlighting too)

10/03/2010 02:52 PM - Alan McGovern

- Status changed from New to Closed

The reason for this is to stop malicious peers initiating multiple connections to a single client. Every connection will have a different port so without this check you couldn't prevent them from opening (and keeping open) as many connections as they want and potentially preventing the targeted client from downloading anything.

Yes, this does mean that two clients behind the one public IP are treated as being the same. Maybe there should be a way to allow this in a setting, but I don't know if it's worth it.