How to get the device IP addresses on Windows Phone

This article explains how to get the IP address(es) currently assigned to a Windows Phone 8 device and how to find out which interface they belong to.

Introduction

Knowing the IP address(es) allocated to your phone is useful to let another device connect to it - i.e. your device listens for connections to this IP address using a StreamSocketListener (basically a TCP Host Port). It can also be helpful for network diagnostics.

Getting IP Addresses

Most information on network connectivity is exposed through the NetworkInformation class. It's GetHostNames() method will give you a collection of `HostName` objects. Rather counter-intuitively the IP address is provided in the `String` property `DisplayName` and not in the `IPInformation` property.

Devices frequently have more than one attached IP address. In most use-cases you will be interested in getting the IP address attached to the Wifi interface to allow for local connections. For this you will now have to check each of the `HostName`'s `IPInformation` (or more specifically the associated `NetworkAdapter` accessible through it). The network adapter has a IanaInterfaceType property which allows you to identify the type of connection. For Wifi, the ID we are looking for is 71. The complete list of possible values can be found in MSDN here.

Getting IP Address on the Emulator

The Emulator correctly reports it's connection to the host's local network as an Ethernet type connection (rather than as a Wifi or Cellular connection on a real device). Therefore when implementing a method to check for an IP address you should also include a check for the `IanaInterfaceType` 6. Unfortunately there is more than one of those on the Emulator - usually the last one in the list is the one you use to access the device from the hosting PC (e.g. using telnet or a browser).

Sample code

The following code shows how to query for the list of IP addresses allocated to the device within the local host network (Wifi, or ethernet on Emulator).

```
        public static string FindIPAddress()
        {
            List<string> ipAddresses = new List<string>();
            var hostnames = NetworkInformation.GetHostNames();
            foreach (var hn in hostnames)
            {
                //IanaInterfaceType == 71 => Wifi
                //IanaInterfaceType == 6 => Ethernet (Emulator)
                if (hn.IPInformation != null &&
                    (hn.IPInformation.NetworkAdapter.IanaInterfaceType == 71
                    || hn.IPInformation.NetworkAdapter.IanaInterfaceType == 6))
                {
                    string ipAddress = hn.DisplayName;
                    ipAddresses.Add(ipAddress);
                }
            }

            if (ipAddresses.Count < 1)
            {
                return null;
            }
            else if (ipAddresses.Count == 1)
```


(C) Copyright Nokia 2014. All rights reserved.

```
            {
                return ipAddresses[0];
            }
            else
            {
                //if multiple suitable address were found use the last one
                //(regularly the external interface of an emulated device)
                return ipAddresses[ipAddresses.Count - 1];
            }
        }
```


(C) Copyright Nokia 2014. All rights reserved.