

Saturday, February 11, 2017

Backward Compatibility of Windows 10 Mobile

This post discusses through a personal experience whether a Windows Phone 8.1 (WP81) app can run well on Windows 10 Mobile (W10M) devices.

It's my understanding that there isn't an official statement from Microsoft that WP81 apps continue to run on W10M devices.

When submitting your WP81 app to the store, you can select whether you want to make that app available for WP81 devices, for W10M devices, or both.

Additionally, you can choose to let Microsoft decide whether to make your app available for any future device families. It's not clear how Microsoft decides on it though. They likely run some tests and analysis internally for evaluation purposes before deciding whether to make that app available for any future device.

If your WP81 app doesn't work on W10M it does make sense not to make available for W10M devices till that app is fixed.

But is it possible that a WP81 app does not work on W10M devices?

Recently, I made my WP81 app available both for WP81 and W10M devices. Months after that app is out, when I was checking the Store Dashboard, I saw the app was crashing on a number of devices. Surprisingly, all the crashes occurred on W10M devices even though the app downloaded to WP81 devices twice as many times. Yet, no crash reported on WP81 devices.

It looks we can tell that WP81 app that can run well on WP81 devices can crash on W10M.

When I looked into the crashes I found crashes at places when the app interacts with other app. Particularly, my app crashed when using share feature enabling the app to send a photo to another app.

I couldn't reproduce the crash on any of my W10M devices. However, some time ago I remember seeing that the built-in Photo app crashed when sharing multiple photos with other app. It's kind of similar scenario and that's noting to do with my app.

It might be even possible that W10M devices running a WP81 app differently than WP8.1 devices running that app. That's why Microsoft probably doesn't say W10M is compatible with WP8.1 albeit they clearly try to make W10M to be backward compatible. It's just they can't guarantee backward compatibility.

No comments:

Labels: [Backward Compatibility](#), [Windows 10 Mobile](#), [Windows Phone 8.1](#)

Saturday, January 7, 2017

Casio Updated Their Smartwatch A Year Later

Casio just announced their new smartwatch that is an updated version of [WSD-F10](#) released a year ago. They have many lineups such as the very successful G-SHOCK however WSD-F20 belongs to the Pro Trek lineup.

This time the watch has built-in GPS receiver which is reasonable because many of Casio's other watches have GPS for ages. Casio is the company has been taking slow but surely steps

towards the success. They don't rush out with something unstable and even when they do experiment release they do it in the most careful way.

That's why FSD-F20 doesn't have any major upgrade.

There are articles on the news portal that missing the heart-rate monitor. If you know Casio you know they are not going to add it just because they could. They will add heart rate monitor if they see there is a good reason to add into the Pro Trek line. How many Pro Trek Casio watches do have heart rate monitor? Heart rate monitor technology has been available for ages. Casio just don't want to add something that doesn't fit into the objectives of the product line. Secondly, they don't add if the technology has a risk that would damage the reputation of Casio. I'm not sure if a heart rate monitor sensor can work for 20 years unlike most of the Casio watches.

The asking price \$500 is a bit high according to the news. Casio have more expensive watches and they know less, and on the top of that they are just digital watches. For example, look at the Frogman. What you pay for is the quality. I'm sure you heard that someone returned their Microsoft Band, Fitbit, or other smart devices. Have you heard about someone returning their Casio watch because it got broken?

No comments:

Labels: [Casio WSD-F20](#), [Smartwatch](#)

Sunday, October 30, 2016

The Challenges of Developing a Photo App for Mobile Devices: Memory Limits

Couple of months ago I started developing a photo app as a side project. The kick-off was ambitious but then I needed to admit I underestimated the complexity of the project as a casual mobile developer.

I was targeting Windows Phone and using managed language but it's quite possible I'd have encountered similar experience with other platforms.

Development of a mobile app poses challenges and in this post I'm writing about one specific challenge I encountered.

The amount of memory an app can use is limited and it depends on the memory size of the device. If the device has 512 MB RAM the app can use only 185 MB of memory. If the app hits the memory limit it will crash.

One might assume if the app is written in managed language the unused memory will be regularly freed up and the app won't hit the memory limit.

In my experience, it was easy to introduce a memory leak that gradually consumed the memory till the point the app hit the memory limit and so it crashed.

When the app is running it already using certain amount of memory. In one scenario the available memory was less than the amount of memory the app tried to allocate. As a result, the app hit the memory limit and so it crashed.

That point, I felt like being in the demoscene back in late '90s when there were the competitions with specific limits. I knew what I wanted to do but I needed to make sure what I do doesn't violate the limits.

A photo app stores each pixel of the photo in the memory. A single pixel can occupy four bytes describing the color. So an image of 3552x2000 requires 27 MB of memory.

The calculation looks like this: $((3552*2000*4)\div 1024)\div 1024 = 27$

One might think that 27 MB of memory in this example is little taking into account that 185 MB is the limit. It's not that simple.

The photo app might need to keep multiple copies of the photo in the memory. For example, the app keeps both the original and edited photos in the memory.

The photo app might need to keep multiple photos in the memory. For example, the app might insert one photo on the top of another. The photo app might need to show a different image to the user than it holds in the memory. So both photos need to be kept in the memory.

Other than that, the photo app needs to keep other things in the memory like temporary buffers, pages of app, history, etc. When these are all added up it's fairly easy to hit the memory limit.

The memory consumption of the app might not be exactly the same between two devices even when the devices have the same amount of RAM. So the memory consumption of the photo app not that just has to be lower than the limit but it's advisable to leave a safety margin to be on the safe side.

Different devices have different amount of RAM and unless you want to optimize your code to the low-end device you need to deal with devices with varying amount of RAM.

I used Visual Studio's built-in Performance and Diagnostics Tools to ensure there are no memory leaks. I also made a measurement how much memory is needed for certain resolution of photos. Then, I set a limit that the app won't open large images that would lead to excessive memory usage and ultimately to crash. I wrote a memory manager class that checks the amount of memory available and adjusts the internal configuration of the app accordingly. As a consequence, certain functionalities—such as zooming into the photo—can take the advantage of more memory.

No comments:

Labels: [Memory Limits](#), [Photo App](#), [Windows Phone 8.1](#)

Thursday, January 7, 2016

Smartwatch from Casio with Dual Display

"There will only be chances later for those who take action now." -- Kazuo Kashio, 2008

I love Casio watches, heck Casio electronics, since my childhood which was 25+ years back. These days there are four Casio watches in my watch box that I regularly wear according to where to go. It's difficult to convey my opinion in an objective way about Casio WSD-F10 that will be reportedly released in March and April later this year.

I just saw the Casio press conference on YouTube. Casio say WSD-F10 created and designed specifically for outdoors and it's 50m water resistant. I would not hesitate to wear it outside or in sea knowing it's built by Casio the maker of G-SHOCK watches. I just haven't had considerable problem with any of my Casio watches. Perhaps the worst thing was that I needed to replace the strap, once in 90's and once in 2000's I guess.

I saw the photo and the watch just looks so cool taking into account it's a smart watch. I still love my Rangeman, and I'm keeping an eye on the Bluetooth watches hoping they will support

Windows Phone one day soon (I'm honestly sad about this but have hopes).

WSD-F10 has got DUAL DISPLAY. This is a big thing because the monochrome display can provide continuous time display. Also when you use the monochrome display rather than the color the battery can last many days or weeks. Casio say the battery can extend till 1 month -- I guess if you don't use the apps.

It has got sensors like compass, accelerometer, gyroscope, barometer. No thermometer? That would be useful, too. I got barometer in my Rangeman and it's pretty practical as you can predict weather changes. In Singapore, for example, I know if the pressure is going up high, then dropping fast it's likely will be raining soon, say, in the next hours.

It's got Android Wear OS on it that I'm not so happy about but Casio can't develop their own OS from scratch as they need to keep their focus on the watch.

Pre-installed apps like map can be really useful I guess specially with the use of compass.

It seems it doesn't have GPS built-in, but whether it can use to the phone's GPS via bluetooth is not clear.

It has got WiFi and bluetooth built in and it's used for notifications when the watch is linked with the smartphone. If it could be linked with Windows Phone that would be even better.

You can remotely operate Casio camera which means Casio take smartwatch seriously. At the moment, my heart however goes for the conservative Bluetooth G-SHOCK that sadly needs update to work with Windows Phone.

No comments:

Labels: [Casio WSD-F10](#), [Smartwatch](#)

Monday, December 14, 2015

Reproducible Bug in MS-DOS Mobile

MS-DOS Mobile has a bug that causes an unexpected termination of the app when it is triggered.

The bug can be reproduced by following the steps below.

- Open MS-DOS Mobile version 1.0
- Type "cd windows" and press enter
- Type "cd.." and press enter
- Again, type "cd.." and press enter to observe the app terminates

It's not obvious whether the app crashes or terminates other way when the bug is triggered because in both cases the app exits.

No comments:

Labels: [Bug](#), [MS-DOS Mobile](#)

Sunday, February 22, 2015

VirtualBox and Windows Phone Emulator (Hyper-V)

VirtualBox requires Hyper-V disabled but Windows Phone Emulator requires Hyper-V enabled. Since Hyper-V is either enabled or disabled I configured my system to have a boot menu for Hyper-V disabled.

I created the new boot menu with the following commands.

```
C:\windows\system32>bcdedit /copy {current} /d "Without Hyper-V"
The entry was successfully copied to {186a5a87-1050-11e4-8250-54ee7520341c}.
```

```
C:\windows\system32>bcdedit /set {186a5a87-1050-11e4-8250-54ee7520341c}
hypervisorlaunchtype off
The operation completed successfully.
```

So I select between the menus during bootup depending on whether I want to use VirtualBox or do Windows Phone development.

No comments:

Saturday, February 21, 2015

Displaying Binary Characters

The wrong way to display an array of random characters in the console (standard output) is like this.

```
foreach (Byte by in buffer) {
    System.Console.Write(Convert.ToChar(by));
}
```

It only displays ASCII characters (0-127) correctly. If you want your code to display ANY characters correctly you may use something like this.

```
Console.OpenStandardOutput().Write(buffer, 0, buffer.Length);
```

Here is the complete example code.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Stdout
{
    class Program
    {
        static void Main(string[] args)
        {
            Random random = new Random(777);
            Byte[] buffer = new Byte[100];

            random.NextBytes(buffer);

            // It only displays ASCII characters (0-127) correctly
            foreach (Byte by in buffer) {
                System.Console.Write(Convert.ToChar(by));
            }

            System.Console.WriteLine();

            // It displays any characters (0-255) correctly
            Console.OpenStandardOutput().Write(buffer, 0, buffer.Length);

            System.Console.WriteLine();
        }
    }
}
```

You can download the source code [here](#).

No comments:

Wednesday, February 18, 2015

Getting the distinct Values of the dictionary

Recently, I needed a solution to return the distinct Values of the dictionary I worked with. In my solution I used the Distinct method with a specified IEqualityComparer. The code looks like this.

```
namespace EqualityComparer
{
    class Program
    {
        static void Main(string[] args)
        {
            Dictionary<int, int> dict = new Dictionary<int, int>();

            dict.Add(1, 1);
            dict.Add(2, 2);
            dict.Add(3, 2); /* Dupe */
            dict.Add(4, 4);
            dict.Add(5, 5);
            dict.Add(6, 5); /* Dupe */

            foreach (KeyValuePair<int, int> element in dict.Distinct(new DictionaryEqualityComparer()))
            {
                System.Console.WriteLine("{0} {1}", element.Key, element.Value);
            }
        }
    }

    public class DictionaryEqualityComparer : IEqualityComparer<KeyValuePair<int, int>>
    {
        public bool Equals(KeyValuePair<int, int> x, KeyValuePair<int, int> y)
        {
            return x.Value == y.Value;
        }

        public int GetHashCode(KeyValuePair<int, int> x)
        {
            return x.Value.GetHashCode();
        }
    }
}
```

The complete source code can be downloaded from [here](#).

No comments:

Sunday, January 25, 2015

Lumia 925 Does Not Turn On - RESOLVED

I had purchased a Lumia 925 in one of Carphone Warehouse branches in Oxford when I had been working in the UK. Couple of months ago I moved to Singapore bringing my Lumia with me. Unfortunately, the following happened.

Today (25/January/2015) it has gone completely dark and it doesn't turn on anymore. Apparently the phone unexpectedly switched off when running the GPS tracking in the official Health & Fitness app. The battery was about at 20% charge.

Since then, I'm not able to turn the phone on. It doesn't response when I put it on charger. It is completely unresponsive. Also, I tried soft reset (volume down + power buttons held down simultaneously) but it didn't have any effect. No vibration, the phone doesn't feel warm when it's on charger. The laptop doesn't detect if it's connected to it.

I just visited the only one Nokia customer care in Singapore (12 Kallang Avenue, Aperia, #03-23) and they suggested to contact the place in the UK I bought this phone from. They say their system cannot handle my case because my Lumia is considered as exported as I didn't purchase it in Singapore but in the UK.

Just sent email to Carphone Warehouse in the UK and will update the article when I hear back from them.

UPDATE Sunday, 25/January/2015. Carphone Warehouse came back with the response that my "device has a valid manufacturers warranty until the 27th June 2015." They also suggest "As you now live outside the UK it may be easier for you to contact Nokia directly to see if they provide a courier service for repairs"

Since it's Sunday, it seems they are not available via phone -- will try tomorrow. In the meantime I also filled a form of my problem on Nokia's [Support page](#).

UPDATE Monday, 26/January/2015. I got a ticket 1-15857619274 at Nokia Support but so far, no-one came back to me apart from an auto-reply.

I was able to call Nokia Customer Care UK but what they suggested is to send the phone to a friend in the UK who can send it to Nokia UK. I can't send the phone directly from Singapore to their UK service because they don't take the responsibility if the phone lost.

Same at Carphone Warehouse the store I purchased the phone. They need someone who take the phone in the store for me so they could forward it to Nokia service. They don't accept phone on the post.

Since I don't know anyone who could bring it for me either to Nokia Service or Carephone Warehouse I'm in trouble.

[@LumiaHelp](#) try to help me -- I appreciate that.

UPDATE Tuesday, 27/January/2015.

Nokia Care Singapore came back with a long-winded email on the ticket 1-15857619274. Basically, they recommend me to send the phone back to the UK to be repaired there, and they suggest to coordinate with them. The problem is, as I wrote above, that Nokia UK doesn't accept my phone via post from Singapore.

Some promising news. [@LumiaHelp](#) is willing to help and they asked for more info.

This morning someone from the Nokia Care called me to confirm the details of the problem. They say they try to help me and they asked for the proof of the purchase. On my request Carphone Warehouse were quick to issue a document of the purchase, and I have forwarded it to Nokia Care. There is a new ticket created 1-15859081472.

UPDATE Thursday, 29/January/2015. As for ticket 1-15859081472, Nokia confirmed that they received the proof of the purchase and they forwarded it to the relevant department for further checking.

UPDATE Monday, 2/February/2015. Promising News. Today, Nokia Care Singapore called me that they repair my Lumia under warranty. So I took the phone and now look forward to their call.

UPDATE Saturday, 7/February/2015. Yesterday, I received the SMS that my phone is repaired and it's ready for collection. I went to pick it up, and now it's with me and it's going well. They re-flashed the phone's software.

Acknowledgement

Thank you [LumiaHelp](#) to forward the message to the relevant department, and arrange to get my phone fixed here in Singapore even I understand the warranty is within UK.

Thank you for [Nokia Singapore Customer Care](#) (12 Kallang Avenue, Aperia, #03-23 Singapore 339511) to handle my case patiently. I especially appreciate the attitude of the Nokia staff who kept me up-to-date via phone regularly (Reference: 1-15859081472). And of course the service team who not just fixed the phone but I got my Lumia back cleaned.

UPDATE Thursday, 12/February/2015. I had reception issues with the newly repaired Lumia. On low-signal there was no reception at all. Also, the service (both voice and 4G) was unreliable after the repair, specially when the phone was under pressure such as using heavily 4G. So on 10th Tuesday I took back to Nokia Care Singapore.

UPDATE Monday, 16/February/2015. Sunday I got the SMS from Nokia Care Singapore that my phone is ready for collection and I excitedly went to collect it. It looks the reception issue is now resolved, and luckily I didn't encounter any more issue. Applied Lumia Denim update just released. :)

No comments:

Monday, January 12, 2015

Lenovo X1 Carbon For App Development

It's been already a few days since I replaced my old laptop with a new one. This one is able to run the emulator smoothly and so I can work on app development without the need to connect my [phone](#).

It's branded as Lenovo X1 Carbon Generation 2. Initially, I considered Dell XPS to get from their official shop but I was surprised to see the one displayed had burn-in on its screen. This helped me to stick to Lenovo. Also, the laptop I was using for three years was Lenovo as well. I was happy with it apart from their [customer care](#).

I was taking a bit of risk when buying X1 Carbon though. How I gonna deal with the adaptive keyboard? Later I just figured out this doesn't seem to be an issue as I just need to adjust its settings.

The other thing I needed to adjust is the trackpad. I disabled few of the functionalities so now it's smoother to use, no accidental clicking, etc.

What I'm not happy about is that the screen glare is too much. My old laptop just didn't have screen glare. This however has the touch screen and I guess the screen is built with a different technology that can made the screen to glare.

Battery is impressive with its about 8 hours lasting time. It can let me to sit long at the balcony developing apps. :-)

To be able to run the phone emulator from Visual Studio the virtualization feature has to be enabled in the bios. This was not enabled by default.

No comments:

Saturday, January 3, 2015

Experiences With Lumia 925

I got my Lumia 925 from the UK on the week of UK release which happened one and a half years ago. This is a great amount of time to feel confidence in writing about my experiences.

Since I started this blog about Windows Phone Development I think it's somewhat relevant to share my experiences here.

Despite the fact I had issues with my Lumia I like the direction where Windows Phones going.

The most outstanding thing is the camera including both the camera as the hardware itself and the camera software shipped with the phone. Many people praised the photos taken with it.

It's got a clean design and to customize the phone to your needs became very simple. It's easy to reach the options and to launch Apps. No need to worry about updates as your Apps got the updates automatically in the background.

There is an App that shows you what are the Apps that generate the most of the Internet traffic. Also, the other App shows you the Apps that consume the most power.

The issues started coming around the time the phone received the Windows Phone 8 to 8.1 update. I don't know if there is any relation between the bugs and the update or it's just a coincidence.

The first major freeze definitely happened weeks after WP 8.1 update. The phone got completely frozen with black screen. It didn't respond to the touch screen or the side buttons including the power button. I thought my Lumia completely died but since there was a Nokia shop nearby I just took there to show them. One of their staff pressed a special key combination to initiate reset and the phone had restarted without losing data.

Unfortunately, few times, I got the kind of freeze that required to reset the phone.

Other time, the freeze manifested itself in other form. The phone was responsive to the side buttons but was not to the touch screen.

I had a habit of keeping the Apps running in the background when not using them so I decided to close the Apps after the use. By doing so rate of freezes I experienced reduced significantly. Without having evidence other than the pure observation my theory is the Apps can expose some bug in lower level.

That's not all though.

I had other issue with it and this issue doesn't seem to be related if Apps running in the background. The symptom is the following. If the phone is placed on flat surface it doesn't ring (but vibrate) when it's called. Interestingly, if I put a pen under it it's ringing on incoming call.

I didn't bring it to any Nokia service as the hassle doesn't currently justify -- the symptoms doesn't have a significant impact on the daily use nevertheless that should not be the point.

No comments:

Wednesday, December 31, 2014

Introduction

I'm already writing a Windows [blog](#) which is proven to be useful for many people including me. In the foreseeable future, however, I expect to have new experiments that cannot fit into the existing blog and so I created this one.

It's about to record my experiment with Windows Phone, mostly, in development point of perspective.

In the recent weeks I already had initial experiments how to get into developing apps for WP and created a [page](#) that is a reference of links I mostly used and surely will.

As a warm up I decided to write a hexadecimal calculator app which is sometimes useful to have

it on your phone. ;-) I'll just post when it's already in the store.

Signing off,
Attila
Singapore