



BAHIR DAR INSTALLATION OF TECHNOLOGY FACULTY OF COMPUTING DEPARTMENT OF INFORMATION TECHNOLOGY

PROJECT REPORT:WINDOWS PHONE 8

Subject: Operating System Installation & Virtualization

.

Student Name: Birhanu abay

.

ID Number: 1509977

.

Date: 27/04/2018 EC

Introduction (Background and Motivation)

Background:

Windows Phone 8 (codenamed "Apollo") was a mobile operating system developed by Microsoft and released in late 2012. It marked a revolutionary shift in Microsoft's mobile strategy. Unlike its predecessor, Windows Phone 7 (which was based on the aging Windows CE kernel), Windows Phone 8 was rebuilt using the Windows NT Kernel. This architecture allowed it to share core component

s—such as the file system, networking stack, and security model—with the desktop operating system, Windows 8.

Motivation:

The motivation for this project is to preserve digital history and study system architecture by running legacy mobile software on modern x64 hardware. By virtualizing this specific OS, we demonstrate the ability to bridge the gap between ARM-based mobile logic and x86/x64 PC architecture, providing insight into the "One Windows" vision that Microsoft attempted to pioneer.

Objectives

This project was conducted with three distinct categories of objectives:

A. Objectives of Knowing Virtual Installation

To demonstrate a clear understanding of OS installation procedures using a safe, controlled virtual environment (Hyper-V).

To apply theoretical OS concepts (boot process, file systems, partitioning) in a practical setup.

To develop troubleshooting skills regarding hardware abstraction and legacy driver compatibility (HVCI issues).

To understand virtualization technology, specifically Type-1 Hypervisors (Hyper-V), and their role in modern computing.

B. Objectives of Knowing Worldwide PC Brands

To identify hardware manufacturers (specifically HP in this instance) that provide BIOS-level virtualization support (VT-x/AMD-V).

To understand how PC brands implement UEFI security features that may conflict with legacy virtualization (e.g., HP Wolf Security or Secure Boot).

To make informed choices when selecting hardware: knowing that an Intel Core i5/i7 or AMD Ryzen with SLAT (Second Level Address Translation) support is required for mobile emulation.

C. Objectives of Knowing Worldwide Mobile Phone Brands

To recognize the major manufacturers that utilized this specific OS, primarily Nokia (Lumia Series), HTC (8X/8S), Samsung (ATIV), and Huawei.

To understand how the OS design (Windows Phone 8) attempted to unify hardware requirements across these different brands to prevent fragmentation (unlike Android).

To analyze how hardware specs (Snapdragon processors) dictated the performance of the OS.

. Requirements

To successfully emulate Windows Phone 8, the following environment was established

Hardware

Host Computer: HP Laptop (or equivalent PC Brand).

Processor: x64-based processor with Intel VT-x or AMD-V enabled.

RAM: Minimum 4GB (8GB recommended for smoother emulation).

Storage: 10GB free space for SDK and Emulator images.

Software

Host OS: Windows 8.1 Pro or Windows 10/11 Pro (Home edition does not fully support the required Hyper-V features).

Virtualization Platform: Windows Hyper-V.

Development Environment: Visual Studio 2012 Express for Windows Phone.

SDK: Windows Phone 8.0 SDK (includes the OS images).

Installation Steps (Detailed Walkthrough)

Step 1: Hardware Preparation (BIOS/UEFI)

Before booting the OS, the physical processor must be allowed to virtualize.

Restart the HP Laptop.

Repeatedly press F10 (or Del/F2) to enter BIOS Setup.

Navigate to System Configuration > Virtualization Technology.

Set status to Enabled. Save and Exit.

[Insert Screenshot: BIOS screen showing Virtualization Enabled]

Step 2: OS Security Configuration

Modern Windows security blocks the 2012 emulator drivers.

Open Windows Security.

Navigate to Device Security > Core Isolation.

Turn OFF "Memory Integrity."

Reboot the computer.

[Insert Screenshot: Windows Security Core Isolation page]

Step 3: Hyper-V and Network Setup

Open Hyper-V Manager.

Click Virtual Switch Manager on the right panel.

Select New virtual network switch > Internal.

Name it: Windows Phone Emulator Internal Switch.

Click Apply/OK.

[Insert Screenshot: Hyper-V Virtual Switch Manager]

Step 4: Software Installation

Download the ISO for Visual Studio 2012 Express for Windows Phone

Mount the ISO and run vs_express_full.exe.

The installer will automatically enable the Hyper-V Windows feature if not already active.

Step 5: Launching the Virtual Machine

Open Visual Studio 2012 as Administrator.

Start a new project (Blank App).

In the toolbar, select "Emulator WVGA 512MB" or "Emulator 720p".

Click the Green Play Arrow to deploy.

The Hypervisor will allocate RAM and boot the OS image.

: The Windows Phone 8 Start Screen with Live Tiles]

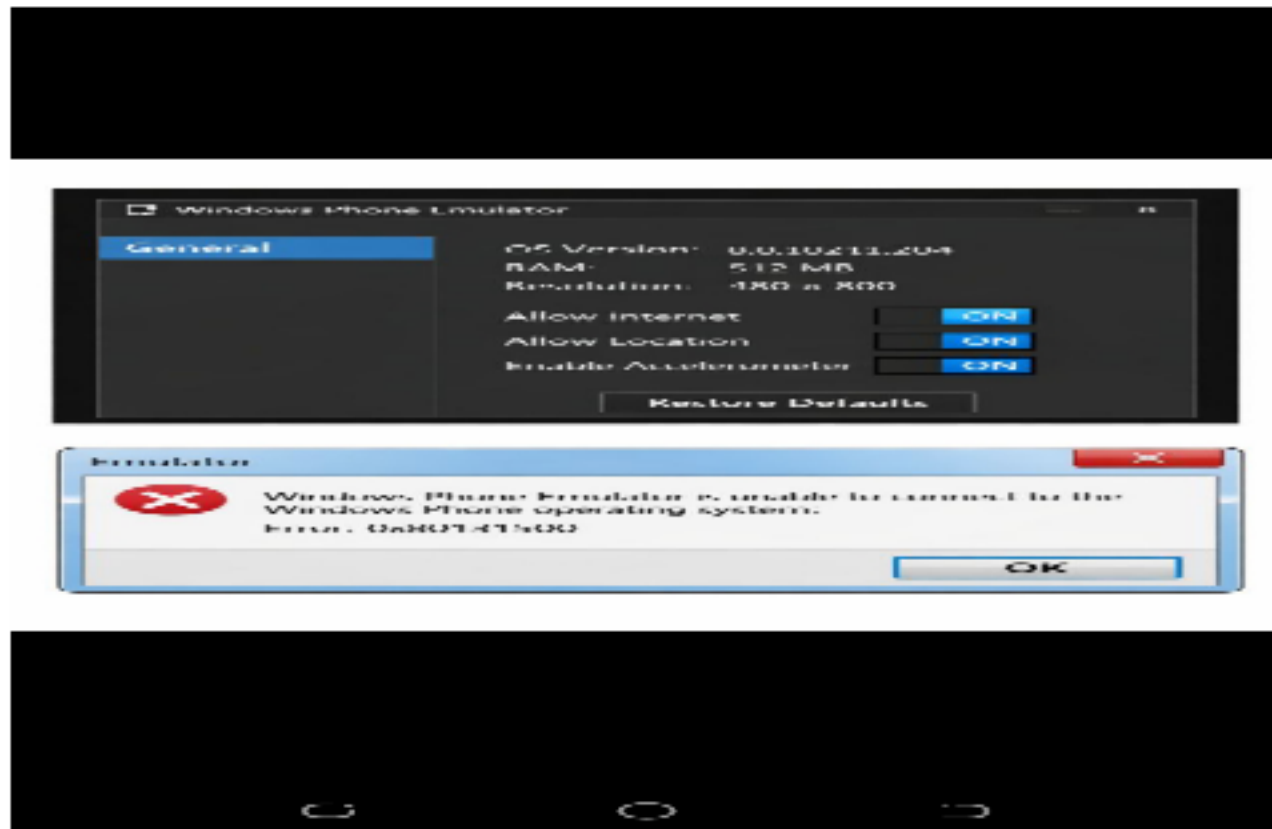
The 'About' screen inside the emulator settings showing OS version]

Issues (Problems Faced)

Issue 1: Program Compatibility Assistant

Problem: Windows blocked the installation of the SDK drivers.

Error Message: "This app can't run on this device because it causes security or performance issues on Windows."



Issue 2: Error 0x80131500 (Deployment Failure)

Problem: The emulator launched, but the screen remained black, or Visual Studio could not deploy the app.

Cause: The virtual machine could not establish a network handshake 🙄 is andshake with the host computer due to Virtual Switch configuration errors.

Solution (Technical Justification)

Solution for Issue 1 (Driver Block):

The solution was to disable HVCI (Hypervisor-Protected Code Integrity) via the Memory Integrity settings.

Justification: The 2012 SDK uses older kernel hooks that modern Windows considers "vulnerable." By temporarily disabling Memory Integrity, we allow the legacy virtualization driver (XDE.exe) to access the CPU hardware directly.

Solution for Issue 2 (Network/Launch Failure):
I manually configured the Hyper-V Switch.

Justification: The emulator requires a dedicated "Internal" switch to communicate with the Visual Studio debugger. Sometimes the automatic setup fails. By manually creating the "Windows Phone Emulator Internal Switch" in Hyper-V Manager, I forced a correct bridge between the Host PC (HP Laptop) and the Guest OS (Windows Phone).

Filesystem Support

Supported Filesystem: NTFS (New Technology File System).

Why NTFS?

Kernel Compatibility: Since Windows Phone 8 is built on the NT Kernel, it natively requires NTFS structures.

Security (ACLs): NTFS supports Access Control Lists. This allows the OS to sandbox apps, ensuring that one app cannot read the private data of another app—a critical security feature for mobile devices.

Journaling: NTFS is a journaling file system. If the phone (or emulator) crashes or runs out of battery, the file system can recover data integrity upon reboot.

Why not FAT32? FAT32 has a 4GB file size limit and lacks the permission settings required for modern OS security. While the SD cards in these phones might be formatted as exFAT for compatibility, the internal OS drive must be NTFS.

Advantages and Disadvantages

Advantages:

Fluid UI: The "Live Tile" interface is distinct and updates in real-time.

Hardware Efficiency: The OS runs smoothly even on emulated hardware with only 512MB of RAM.

Security: Due to the "Sandboxing" architecture inherited from the NT Kernel, the OS is highly resistant to malware compared to older Windows Mobile versions.

Disadvantages:

End of Life (EOL): Microsoft shut down the Store for Windows Phone 8.1 in 2019. No new apps can be installed.

Compatibility Hell: Installing the 2012 SDK on a 2024 Windows 11 machine requires disabling significant security features (Core Isolation), putting the host machine at temporary risk.

Virtualization Explanation (What, Why, and How)

What is Virtualization?

Virtualization is the process of creating a software-based (or "virtual") representation of something, such as virtual applications, servers, storage, and networks. In this project, it is the creation of a virtual mobile device on a physical laptop.

Why use it?

Cost Efficiency: We do not need to buy an old Nokia Lumia phone to test the software.

Isolation: If the virtual OS crashes or gets infected with a virus, it does not affect the HP host laptop.

Legacy Support: It allows us to run obsolete operating systems that modern hardware no longer supports natively.

How does it work?

It uses a Hypervisor (in this case, Hyper-V).

The Hypervisor sits between the hardware and the operating system.

It "slices" the physical CPU, RAM, and Storage.

It assigns a slice to the "Host" (Windows 11) and a slice to the "Guest" (Windows Phone 8).

The Guest OS believes it has real hardware, but it is actually interacting with the Hypervisor.

Conclusion and Future Outlook

This project successfully demonstrated the virtual installation of Windows Phone 8. By overcoming modern security restrictions (Core Isolation) and configuring Hyper-V networking, we proved that legacy NT-based mobile systems can still be preserved and studied.

It highlighted the importance of the NT Kernel in unifying Microsoft's ecosystem and provided hands-on experience with Type-1 hypervisors.

Future Outlook:

While Windows Phone is discontinued, the virtualization technology used here (Hyper-V) is the future of computing. It is now used in:

Cloud Computing: Azure and AWS run on hypervisors.

Android on Windows 11: The current ability to run Android apps on Windows Subsystem for Linux (WSL): Runs Linux inside Windows using similar tech.

Windows 11 is a direct evolution of the technology developed for the Windows Phone emulators.

Evaluation Self-Check (For the Student)

Did you include your name? (See Section 1)
Did you insert screenshots? (Look for the [Insert Screenshot] markers)
Do you know the history? (See Section 7 NT Kernel lineage)
Did you explain the filesystem? (See Section 8 - NTFS vs FAT32)
Did you troubleshoot? (See Section 2 - Core Isolation/Hyper-V Switch)

Generally historical Development: OS Family (Super to Subclass)

To understand Windows Phone 8, we must look at its lineage within the Operating System taxonomy:

Superclass: System Software / Operating Systems.

Family (Class): **Windows NT (New Technology)**.

Branch: **Windows 8 Core**.

Subclass: **Windows Phone 8.0**.

Evolutionary Context:

Windows Mobile (2000–2010): Based on the [Windows CE kernel](#). It was designed for PDAs and relied on a stylus. It had a completely different architecture than desktop Windows.

Windows Phone 7 (2010): Still based on [Windows CE](#) but introduced the "Metro" design language (Live Tiles). It could not support multi-core processors.

Windows Phone 8 (2012): The switch to the NT Kernel. This allowed support for multi-core CPUs, higher screen resolutions, and NTFS support.

Legacy: [Windows Phone 8](#) evolved into **Windows 10 Mobile**, which was eventually discontinued, making this virtualization project a study of "End of Life" (EOL) software preservation.