



# Running Windows 8 on top of Android with KVM

21 October 2013

Zhi Wang, Jun Nakajima, Jack Ren

# Legal Disclaimer

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL® PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER, AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL® PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT. INTEL PRODUCTS ARE NOT INTENDED FOR USE IN MEDICAL, LIFE SAVING, OR LIFE SUSTAINING APPLICATIONS.

Intel may make changes to specifications and product descriptions at any time, without notice.

All products, dates, and figures specified are preliminary based on current expectations, and are subject to change without notice.

Intel, processors, chipsets, and desktop boards may contain design defects or errors known as errata, which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Intel and the Intel logo are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

\*Other names and brands may be claimed as the property of others.

Copyright © 2013 Intel Corporation.



# Agenda

- Motivation
- Prototype
- Demo
- Next Steps

# Motivation

- Hardware-based virtualization features have been enabled even on mobile devices
  - EPT, VPID are available on latest Intel® Atom™ processors
  - Unmodified operating systems can be run in virtualization
- Android\* is very popular mobile OS
  - Linux-based, therefore KVM module should be available
  - Userland is very different from typical Linux distros
- What does it takes to enable KVM on Android?
  - **Exercise: Run Windows 8 on x86 Android**

# Modes of Windows 8 Guests

## 1. Legacy PC mode

- Legacy PC devices (e.g. mouse, keyboard, network, IDE, etc.)
- Legacy PC BIOS
- OSPM can be disabled

## 2. Tablet mode – much more complex

- Modern tablet devices (e.g. touch panel, sensors, secure boot, etc.)
- UEFI
- Advanced OSPM, Connected Standby

Begin with “Legacy PC Mode”

# Scope of Prototype

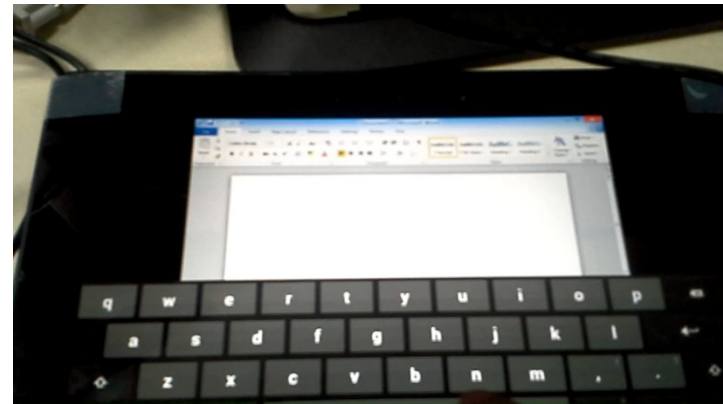
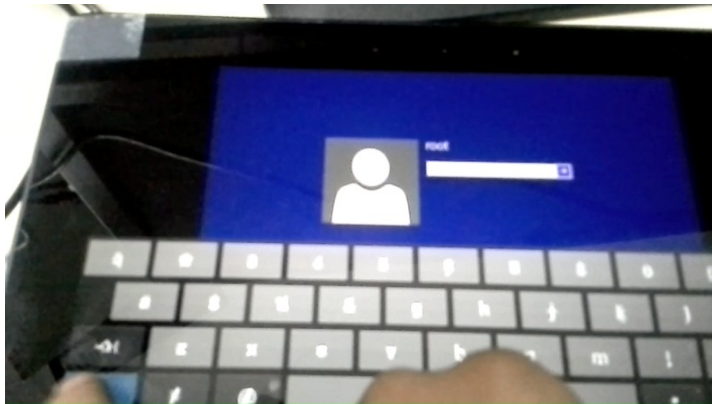
- Platforms
- Limbo-android
- Modifications
- Current Status

# Android\* Platforms

- Core-based Tablet
  - Intel Ivybridge Core i5 3337U with 4G memory and 128G SSD
- Intel® Atom™-based Tablet
  - Intel® Bay Trail Atom™ with 2G memory and 128G eMMC

# Qemu/KVM for Android\* - Limbo

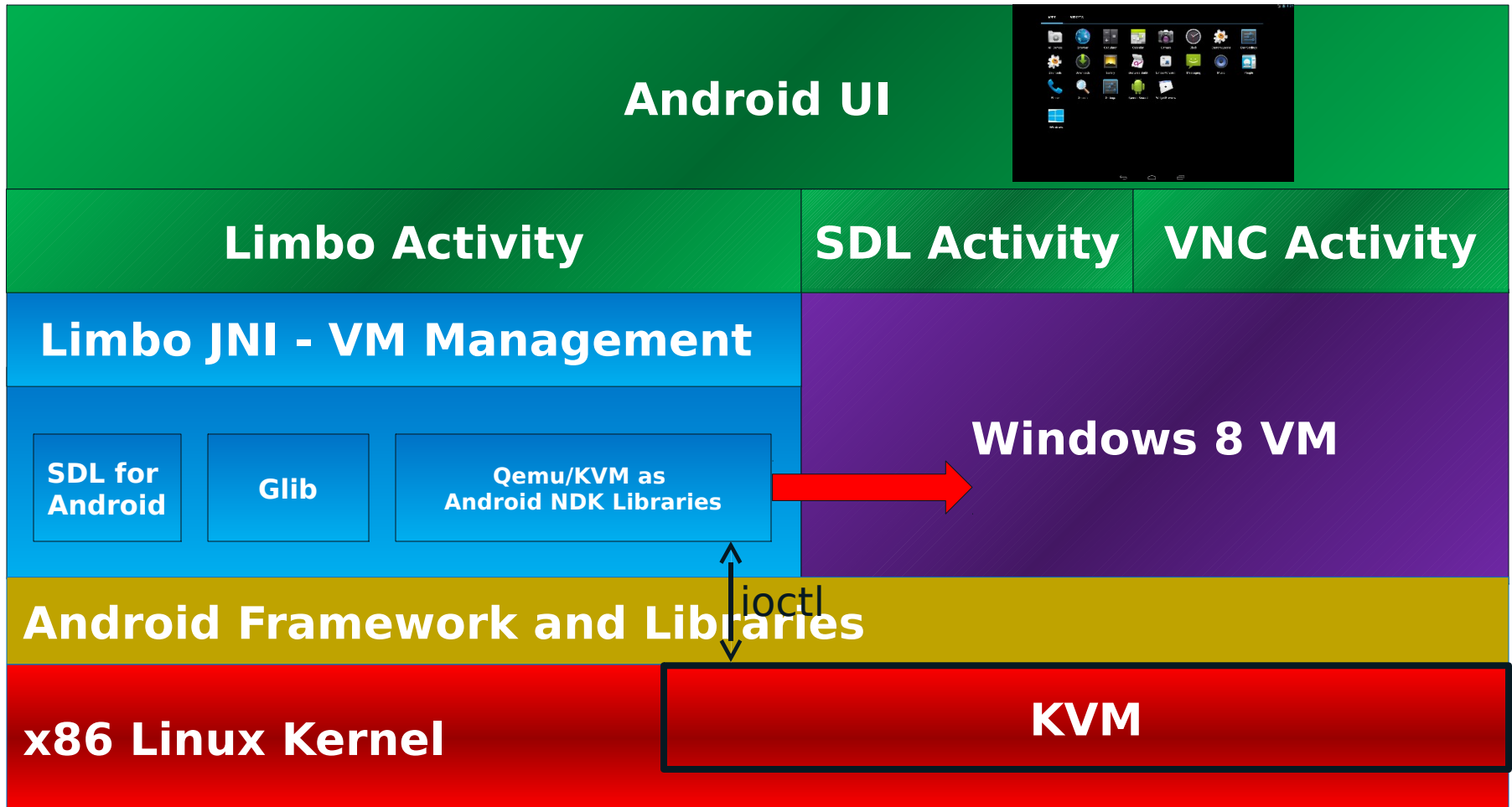
- What is Limbo?
  - Limbo is an open source project based on Qemu which can manage virtual machines under ARM\* Android.
  - <https://code.google.com/p/limbo-android/>



Port limbo-android to  
x86/KVM



# Overview of Limbo-Android – Qemu/KVM for Android



# Qemu/KVM under Android with SDL

- Display
  - Qemu allocates and draws surface via SDL API
  - SDL Android libraries manipulates the surface from SDL Activity via Android EGL API and NativeWindow
- Input
  - SDL Activity forwards user activities from Android WindowManager to SDL Android Libraries
  - SDL Android Libraries forward them to Qemu via SDL event loop
- Sound
  - SDL Activity contains a AudioThread and use AudioTrack from AudioFlinger to playing music forwarded from Qemu

# Modifications Required

- Linux Kernel
  - No modifications: just enable KVM in the kernel configuration
- Bionic (C library for Android)
  - Add missing system calls: `pread64()/pwrite64()/lseek64()`.
  - Add missing POSIX function: `sigtimedwait()`
    - Required for KVM
- limbo-android
  - Resolved x86 build issues
  - Pass KVM-specific arguments to Qemu in Limbo JNI.
    - “-enable-kvm”, virtio configurations
  - Cleanups for 64-bit support (to handle large images)

# Current Status

32-bit SMP Windows 8 works well: 2GB (on Core) and <1GB (on Atom)

- Virt IO: Network and storage
- Display: SDL/VNC
- Sound: SDL Activity under Android AudioFlinger
- Input: Keyboard / Mouse / Touch events are passed from Android Activity to Qemu

RDP Support

- Next Page

# RDP Support

- RDP and Remote APP
  - Gestures – Scrolling / Scaling / Software Keyboard
  - Software Keyboard Auto Activation
  - Seamless UI
  - Sound Forwarding
  - Share Folder

# Demo

Load VM: demo-win8

Resume VM State: None

User Interface: SDL

Start Stop Reset Ready

VM Device Configuration

CPU: pentium3

CPUs: 1

RAM Memory Size: 768

CDROM (\*.iso): None

Floppy Drive A(\*.img): None

Floppy Drive B(\*.img): None

Hard Disk A(\*.img/\*-qcow2): /data/win8.img

Hard Disk B(\*.img/\*-qcow2): None

# Next Steps

2D/3D acceleration:

- Issues: No Vmware VGA / QXL drivers for Windows 8 (WDDM)

Tablet Mode:

- Virtual UEFI
- Sensors
  - USB HID Class Sensors emulation in Qemu
- Camera
  - USB Video Class Camera emulation in Qemu
- UX improvements
  - Touch optimizations
- Virtual Connected Standby
  - “Hardware-reduced ACPI”

