Design and Implementation of Tizen Emulator

Yeongkyoon Lee and Hyun-goo Kang
S-Core Co., Ltd.
Contents

• Introduction
• Key Design Concepts
• Tizen Virtual Board
• Advanced Functionalities
• Conclusion
Introduction
Tizen Development Runtime

- **Web simulator**
  - For web applications
  - Support oneshot web runtime
- **Emulator**
  - For web/native applications and platforms
  - Support full system runtime
- **Physical phone**
  - For web/native applications and platforms including BSP (Board Support Package)
  - Support full system runtime
Tizen Emulator Tools (1/2)

• Tizen emulator tools
  – Provide development environment for Tizen platform/app w/o real physical devices
  – Consist of QEMU based **Emulator**, **Emulator manager** and **Event injector**
  – Provide interoperation with **Tizen IDE** using **SDB** (Smart Development Bridge)
  – Currently, support x86 guest arch only
Tizen Emulator Tools (2/2)

IDE

Emulator

Event Injector

Launch/debug app using SDB

Generate virtual events

Emulator Manager

Launch emulator

Host OS
# Features Summary

- Tizen emulator features compared to physical targets or other emulators

<table>
<thead>
<tr>
<th>Category</th>
<th>Pros</th>
<th>Cons</th>
</tr>
</thead>
<tbody>
<tr>
<td>Usability</td>
<td>- Easy to get (just download SDK)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Configurable devices and skin</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Various virtual device input support</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Multi-instance support for emulator</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Host directory sharing</td>
<td></td>
</tr>
<tr>
<td>Performance</td>
<td>- Capability of HW VT acceleration</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Codec and GLES acceleration</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- virtio</td>
<td></td>
</tr>
<tr>
<td>Compatibility</td>
<td></td>
<td>- Not all devices are fully supported</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- e.g. WiFi, BT, radio, etc.</td>
</tr>
</tbody>
</table>
Key Design Concepts
Emulation vs. Simulation

• Emulation
  – Strict implementation of device spec
  – No guest modification
  – Conventional approach of QEMU
  – Poor performance and flexibility

• Simulation
  – API-centric implementation
  – Guest modification needed
  – e.g. virtio
Hybrid Emulation

- “HW emulation + simulation” for better performance and flexibility

**Physical Target (ARM)**
- App
- FW
- Kernel
- Real device

**Emulator (x86)**
- App
- FW
- Kernel
- CPU (x86)
- HW Emulation
- Simulation
- TCG
- KVM (or HAX)

**Virtual board (x86)**

- VCPU: Virtual CPU
- KVM: Kernel Virtual Machine
- HAX: Hardware Accelerated eXecution
- TCG: Tiny Code Generator
Tizen Virtual Board
Virtual Board Overview

• Tizen virtual board provides virtual HW devices

• Board constructions
  – QEMU PC board
    • Intel Pentium II chipset + devices
  – virtio devices from QEMU
    • virtio-disk, virtio-net and virtio-9p
  – New devices from Tizen
    • virtio-opengl
    • Overlay, codec, camera, multi-touch, etc.
Execution with HW VT Acceleration

- High performance w/ HW VT support
  - Intel VT-x or AMD-V
  - Special kernel driver needed: KVM in Linux and HAXM in Windows

- Additional performance considerations
  - Removing core affinity from Windows
  - Optimization for guest memory access
  - Separate display thread

```
User mode (host PC)
  User Process
  Tizen emulator (User Process)
  Virtual Board (Devices)
  HW VT Interface

Kernel mode (host PC)
  Kernel
  KVM (or HAXM) Driver

Guest mode (Tizen)
  Tizen VM
    User
    Kernel
    VM Entry/Exit
```
Skin/Display

- Emulator skin with high portability/usability written in Java
- Display devices based VGA
  - VGA device (1 framebuffer) + Overlay device (2 overlay buffers)
  - Sharing SDL display buffer between QEMU and skin processes
GLES Acceleration

- Support OpenGLES/EGL APIs with high performance
- GLES/EGL calls are performed in offscreen rendering by host GPU if it exists
  - gl command passing to QEMU via virtio
  - gl command buffering to avoid copy-back overhead
Codec Acceleration

- Support video playing even w/o HW VT support
  - Typically, QEMU TCG is not fast enough to run guest video codec
- AV Codec API delegation to host
Advanced Functionalities
Event Injector

- Rich event injection for ease of test
  - Sensors
    - Accelerometer
    - Gyroscope
    - Geomagnetic
    - Proximity
    - Light
    - Motion
  - Location
    - Manual/Map/Log file
  - Telephony
    - Call/SMS (from/to event injector)
  - NFC
    - NDEF message
    - NFC Tag
    - P2P
  - Device
    - Battery level
    - Earjack
    - USB
    - RSSI
Host Directory Sharing

• Directory sharing between host PC and Tizen guest
  – Useful for large size of resource files (e.g. multimedia files)
  – No need to upload files to guest via ssh
  – No worries about guest disk size

• Implementation via network file sharing
  – For Linux host
    • virtio 9p protocol
  – For Windows host
    • Samba protocol (Windows7 needs ID/PW according to security policy)
Emulator Manager

- Provides interfaces to experience various emulator targets and to test portability
  - VM management including file system images and HW configurations
  - VM (Virtual Machine): a set of configuration for Tizen guest

- Supports configurable virtual HWs
  - Display resolution: HVGA / WVGA / WSVGA / HD
  - Display density (DPI)
  - RAM size
  - Front key type

- Saves disk spaces for Tizen guests using QCOW2
  - QCOW2 (QEMU Copy-On-Write 2) image format
    - “Read-only base image” / “Read-writable image” pair
    - Multiple images can share a same base image
Conclusion

• Hybrid emulation is effective for mobile emulator
  – Flexible to support various mobile devices with high performance
  – OpenGL / Codec performance (Ubuntu 11.04, Intel i7 3.4GHz, 4GB RAM)
    • WebGL fpstest (fps): 0 (guest mesa w/o VT) $\rightarrow$ 6 (w/o VT), 20 (w/ VT)
    • H.264 decoder (fps): 0 (guest codec w/o VT) $\rightarrow$ 3.7 (w/o VT), 22.4 (w/ VT)

• x86 guest boosts up emulator performance with HW VT support
  – Execution performance (Windows7, Intel i7 2.93GHz, 4GB RAM)
    • Booting time (sec): 57 (w/o VT) $\rightarrow$ 17 (w/ VT)
    • CoreMark (iteration/sec): 910 (w/o VT) $\rightarrow$ 8450 (w/ VT)
    • SunSpider (msec): 6540.5 (w/o VT) $\rightarrow$ 575.2 (w/ VT)

• Event injector is user-friendly enough to test various virtual events

• Emulator manager provides interfaces to experience various emulator targets and to test portability