Android Porting

移植要領與策略分析

by Jollen Chen（陳俊宏）

email: jollen@jollen.org
blog: jollen.org/blog
plurk: www.plurk.com/jollenchen

活動地點：台大集思會議中心洛克廳-台北市羅斯福路四段85號
活動時間：2009 年 6 月 24 日(三) 10:00~16:30 共4.5小時

主辦單位：

CTimes

www.jollen.tw

www.jolle.tw

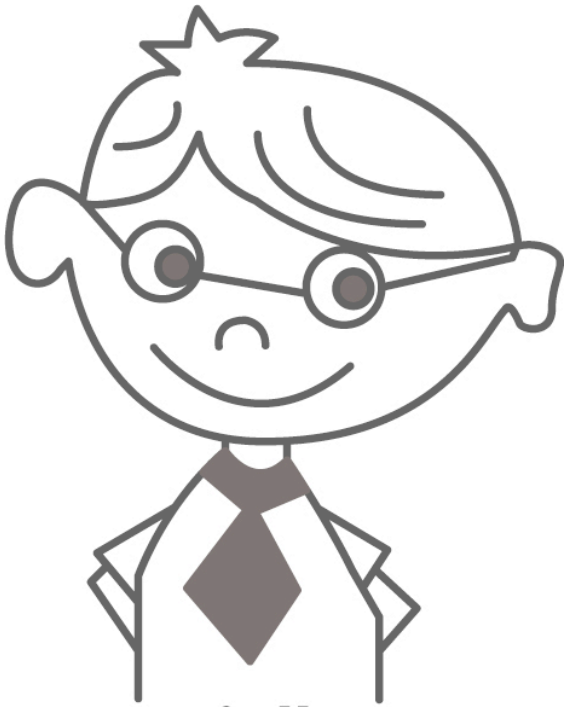
- Section 1. 移植策略、框架、Android Kernel

android market

open handset alliance

Android Porting

- ☑ Android porting, 把 Android 放到你的 hardware
- ☑ 移植三層面
  - 建立 product 分支、維護自有版本
  - 架構端的移植
  - 驅動程式

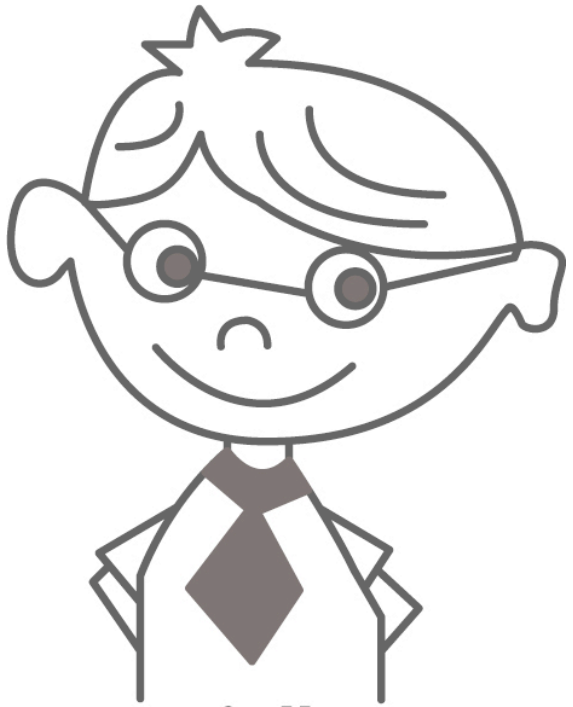


Android Porting 技術層面

- ☑ Application (API)
- ☑ Product
- ☑ Framework / Library
- ☑ Android kernel

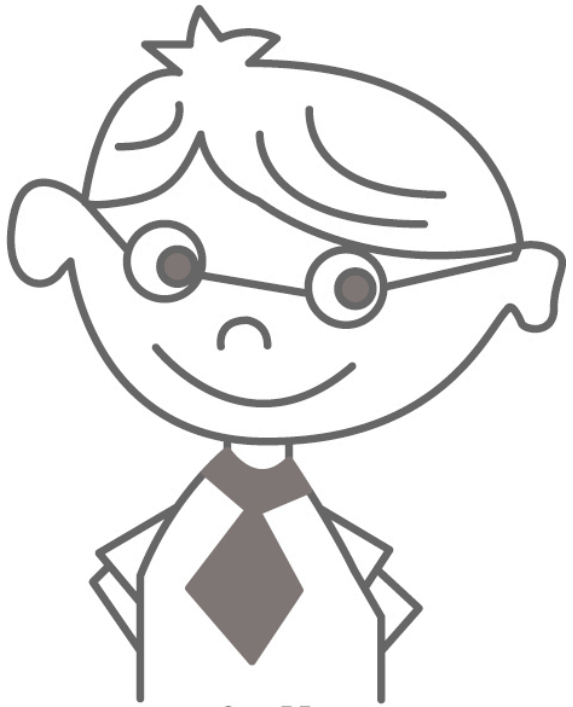
Android Porting: 不用動刀的做法

- ☑ 建立 new product、編譯 Android
  - cupcake 支援的架構 armv4/armv5/x86
- ☑ API Level
  - 注意 Android 版本
- ☑ Vanilla Kernel Configs
  - Android framework 驅動程式
- ☑ Non-Vanilla Kernel
  - Merge Android framework 驅動程式



Android Porting: 動手術

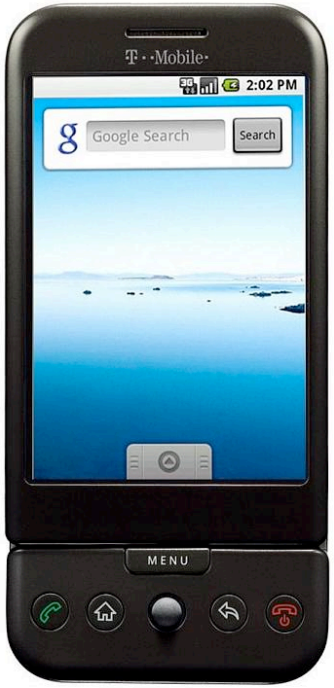
- ☑ 加入週邊驅動程式
  - 修改 Android framework、加入 library
- ☑ 有關 Android framework 驅動程式
  - ASHMEM、wakelock 等
- ☑ 加入 shared library
  - 修改 Android framework
- ☑ 架構的移植
  - case study: FreeRunner



API Level

System Image	API Level	Release
Android 1.5	3	2009.4.27
Android 1.1	2	2009.2.10
Android 1.0	1	2008.9.23

- ☑ 2007.11.5: Android 首度現身
- ☑ 2007.11.12: Adnroid SDK （early look version）開放下載
- ☑ 2008.1.3: Adnroid Developer Challenge 比賽開始
- ☑ 2008.2.13: Android SDK m5-rc15 釋出
- ☑ 2008.5.12: Top 50 Android Application 公佈



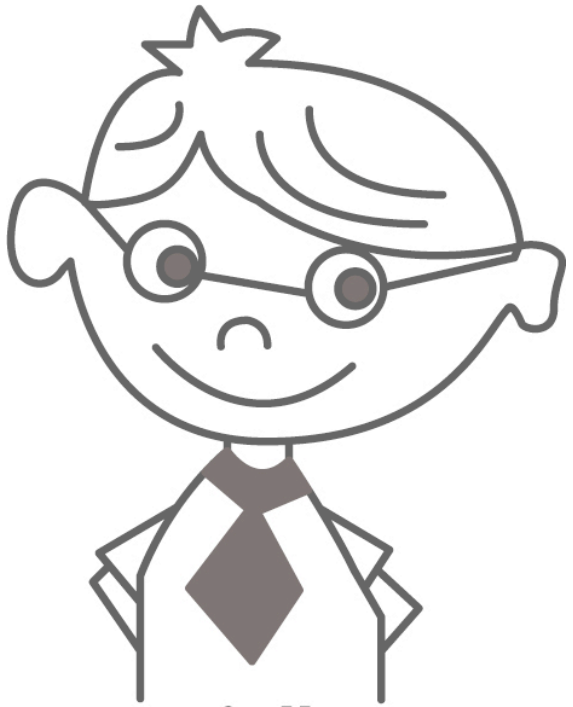
Phone
ARMv4/v5

Netbook
x86

Devices
misc...

Cupcake 重要的新功能

- ☑ MMS夾檔
- ☑ WebView 開始支援 Touch events
- ☑ WebView 支援 SquirrelFish (JavaScript engine)
- ☑ 加入了 IME
- ☑ Basic x86 support
- ☑ SIM Application Toolkit 1.0
- ☑ ...



Google groups

android-porting

android-porting 討論

Home

New since last time: 193 messages

Android Porting
Porting Android to various hardware.

Welcome to the Android community! We're glad you're here and invite you to participate in these discussions. Before posting, please read the Groups Charter that covers the community guidelines.

To get the most out of this group, please do the following before you post:

- **Read the FAQs**. The most common questions are addressed in this frequently updated list.
- **Type in keywords of your question in the main Android site's search bar**. This search encompasses all previous discussions, across all groups, as well as the full contents of the site, documentation, and blogs. Chances are good that

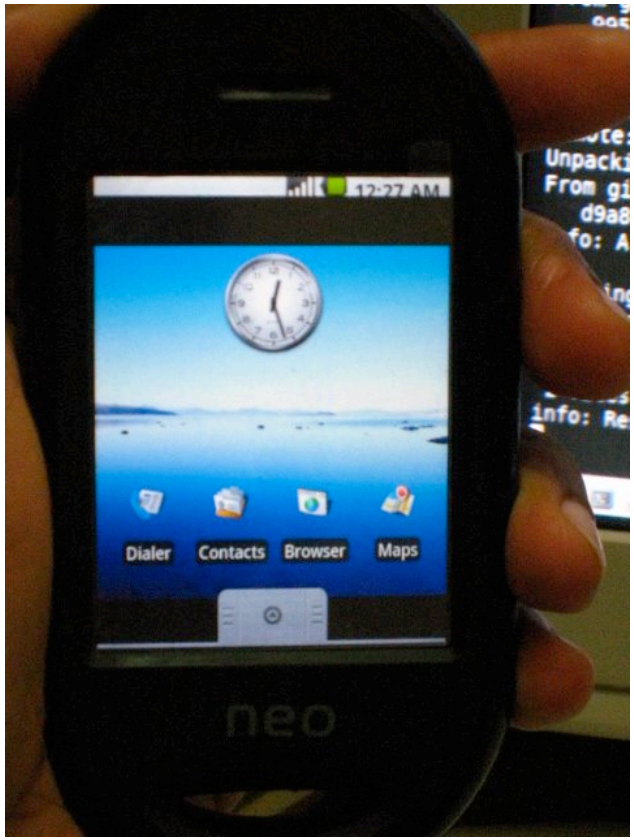
Android Source Code

android.git.kernel.org

☑ 1. Cupcake 正式加入 x86 ports

☑ 2. 於./.repo/manifest.xml 加入 eee_701 platform 取得完整 x86 ports

Android / armv4 的移植

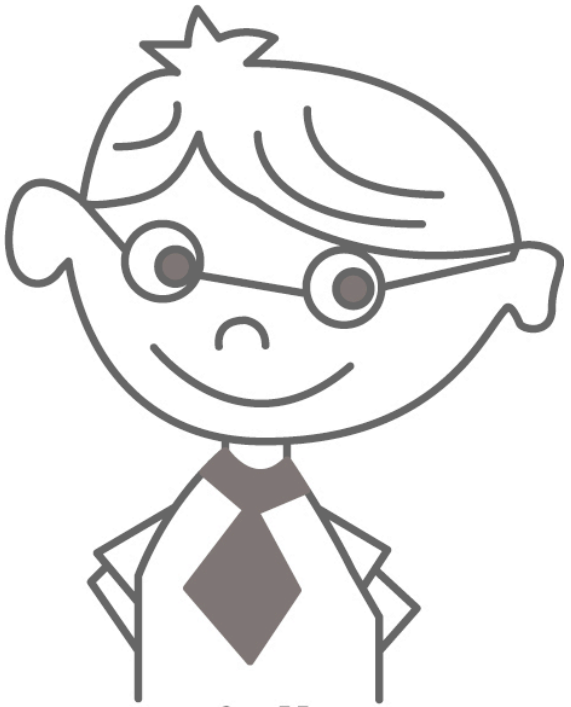


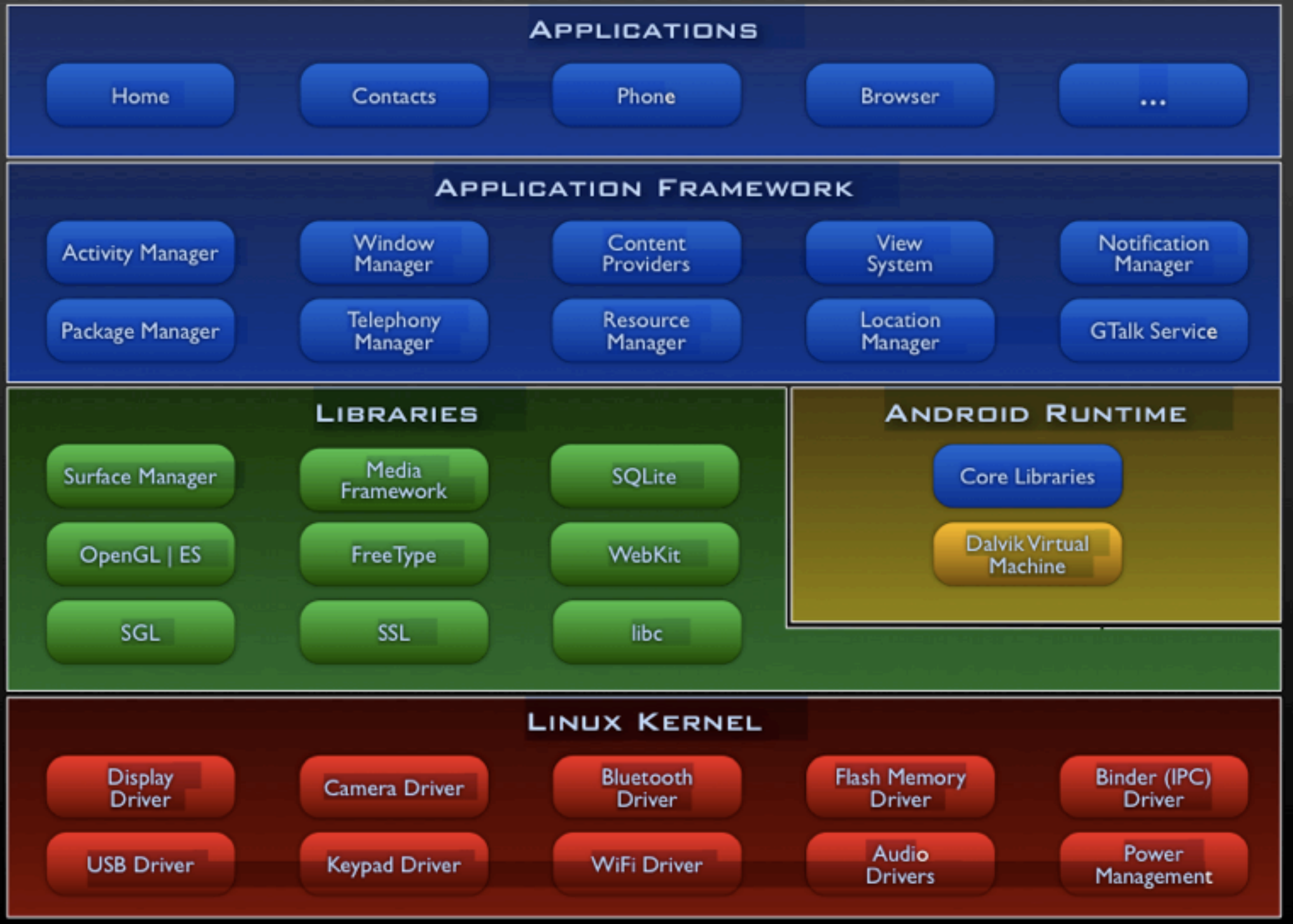
git.koolu.org

- ☑ 主要的 armv4 移植可由 Koolu 取得
- ☑ 支援 s3c2410 / s3c244x
- ☑ 相容於 Neo FreeRunner

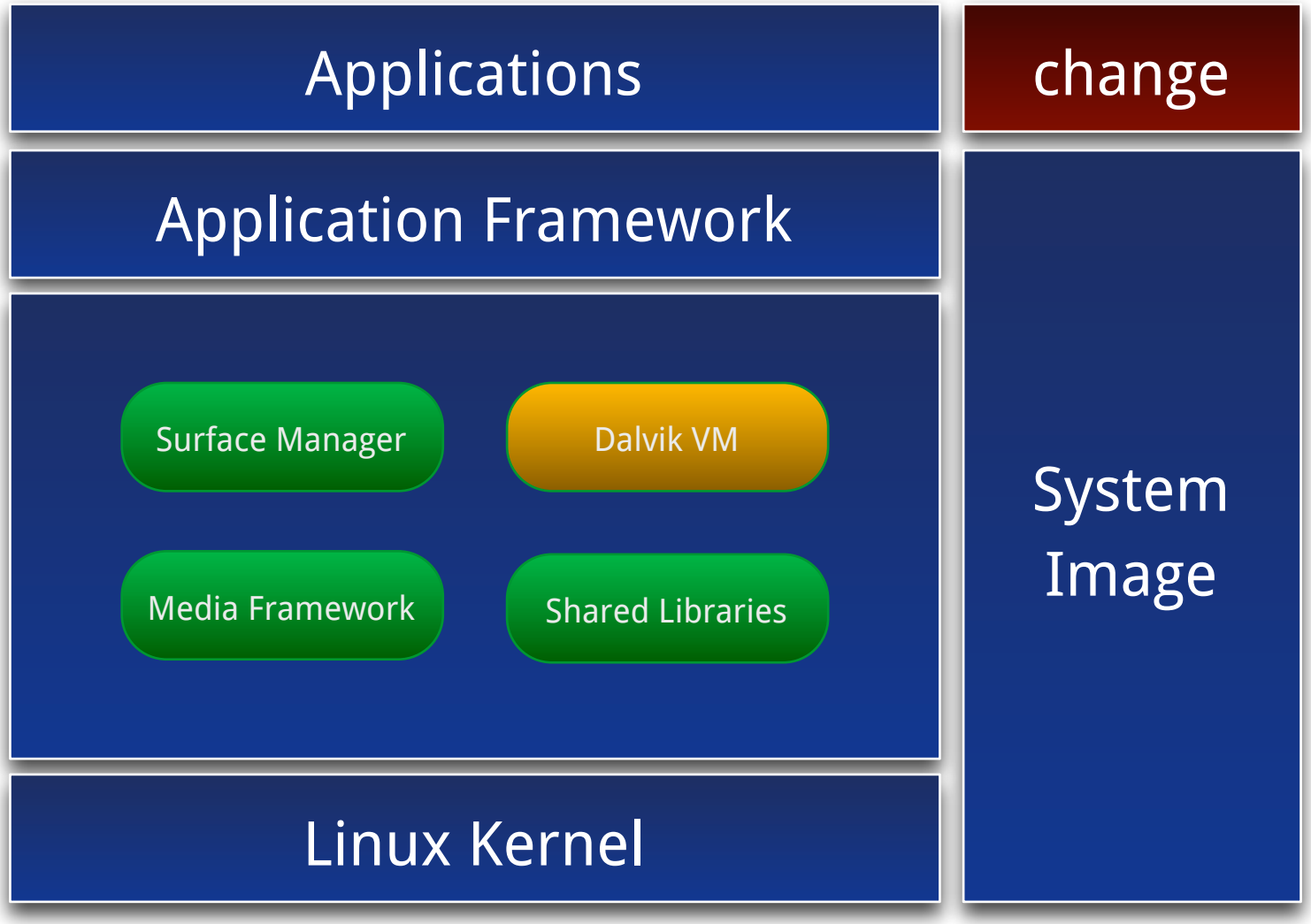
下載 Android / armv4 移植

```
$ mkdir koolu-android
$ cd koolu-android
$ repo init -u git://git.koolu.org/freerunner/platform/manifest.git -b koolu-1.0
$ repo sync
```

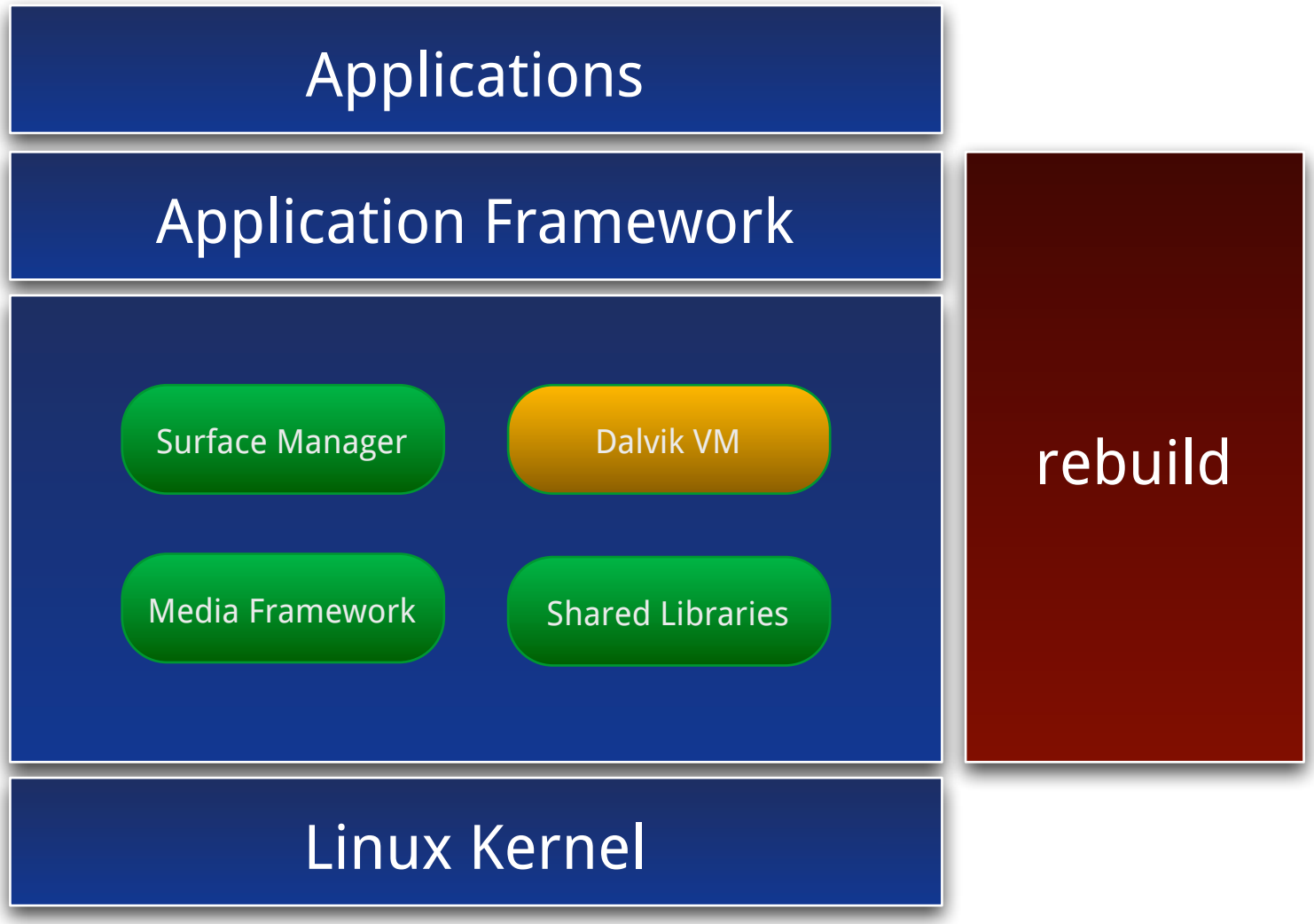



Applications
Home
Contacts
Phone
Browser
...
Application Framework
Activity Manager
Window Manager
Content Providers
View System
Notification Manager
Package Manager
Telephony Manager
Resource Manager
Location Manager
GTalk Service
Libraries
Surface Manager
Media Framework
SQLite
OpenGL | ES
FreeType
WebKit
SGL
SSL
libc
Android Runtime
Core Libraries
Dalvik Virtual Machine
Linux Kernel
Display Driver
Camera Driver
Bluetooth Driver
Flash Memory Driver
Binder (IPC) Driver
USB Driver
Keypad Driver
WiFi Driver
Audio Drivers
Power Management

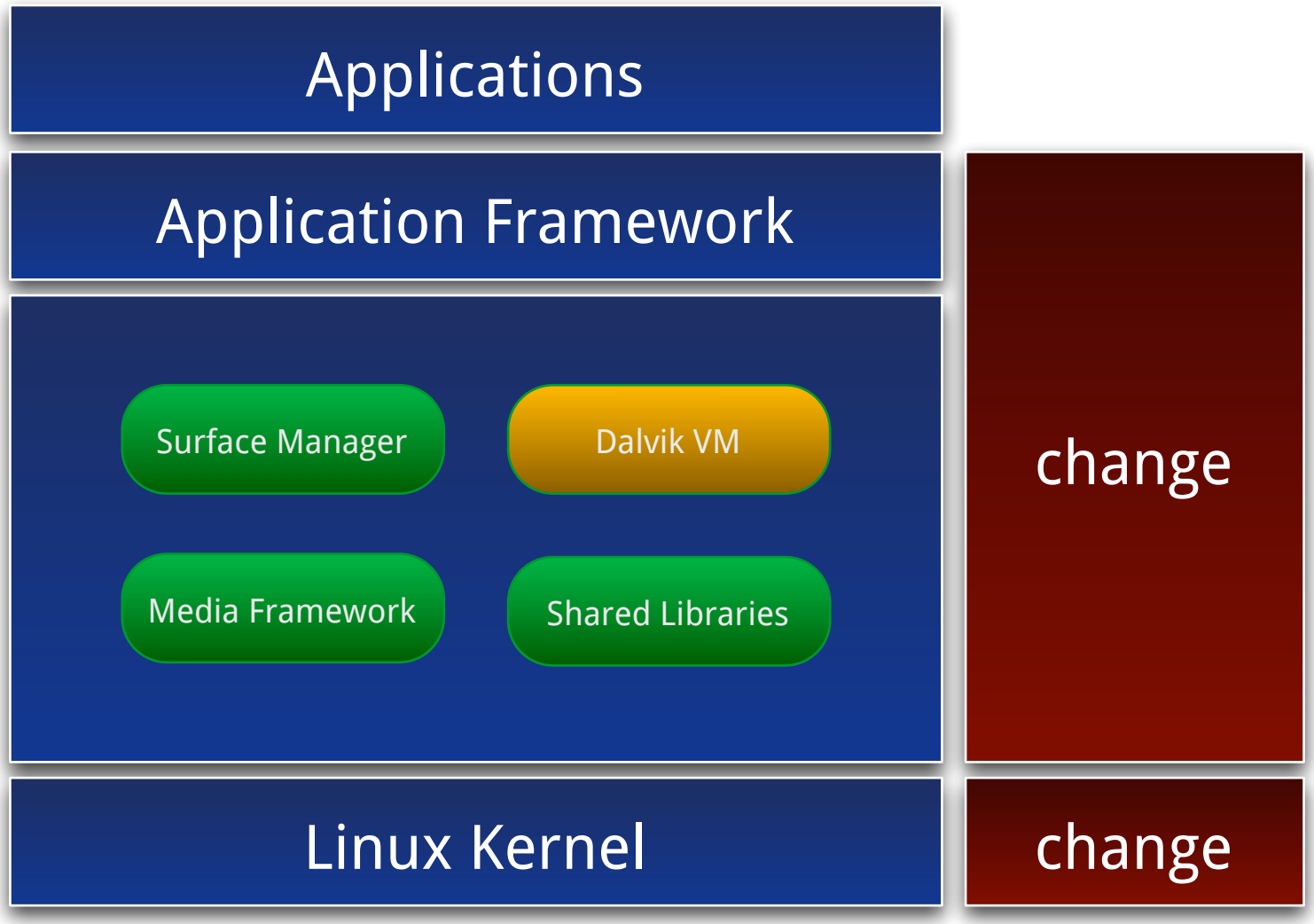
Application Developers



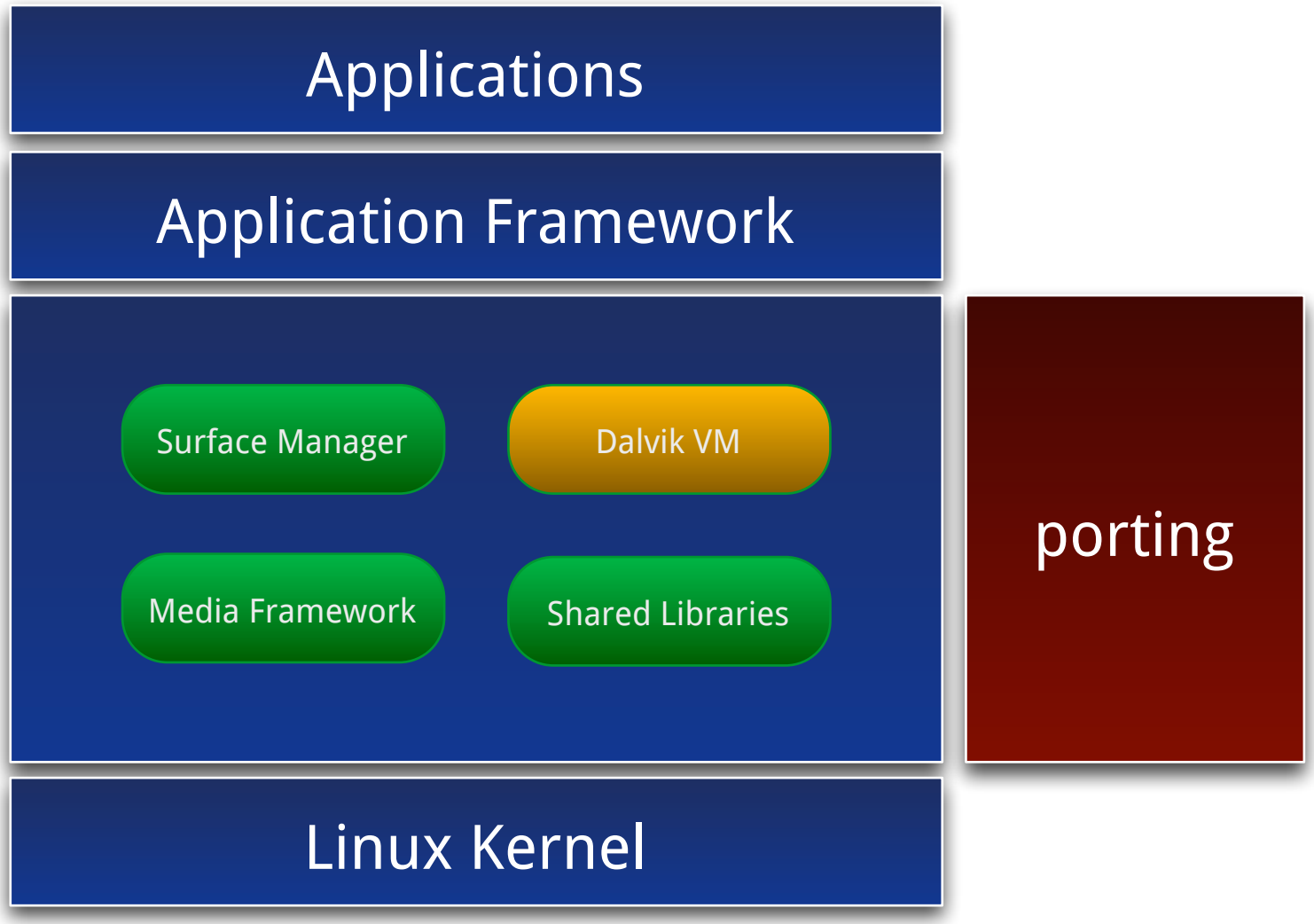
Product Branch Maintainer



Hardware Developing



Architecture Porting



Android Toolchain

- ☑ android-toolchain-20081019
  - http://android.git.kernel.org/pub/
- ☑ gcc 4.2.1 / binutils 2.17 / gdb 6.6
- ☑ ARMV5te+
- ☑ ABI: EABI, AAPCS
- ☑ --with-float=soft --with=fpu-vfp
- ☑ --enable-threads (single)

Android Kernel

- ☑ Binder
- ☑ Ashmem (Android shared memory)
- ☑ PMEM (Processor memory allocator)
- ☑ logcat (Android Logger)
- ☑ wakelock (Android Power Management)
- ☑ Alarm

	Key Features	2.6.23	2.6.25	2.6.27	2.6.29
1	Alarm Driver	O	O	O	O
2	Android Logger	O	O	O	O
3	Low Memory Killer	O	O	O	O
4	Power Management	O	O	O	O
5	USB Gadget	O	O	O	O
6	ASHMEM	X	O	O	O
7	PMEM	X	X	O	O
8	x86 Support	X	X	O	O
9	./drivers/staging/Android/	X	X	X	O

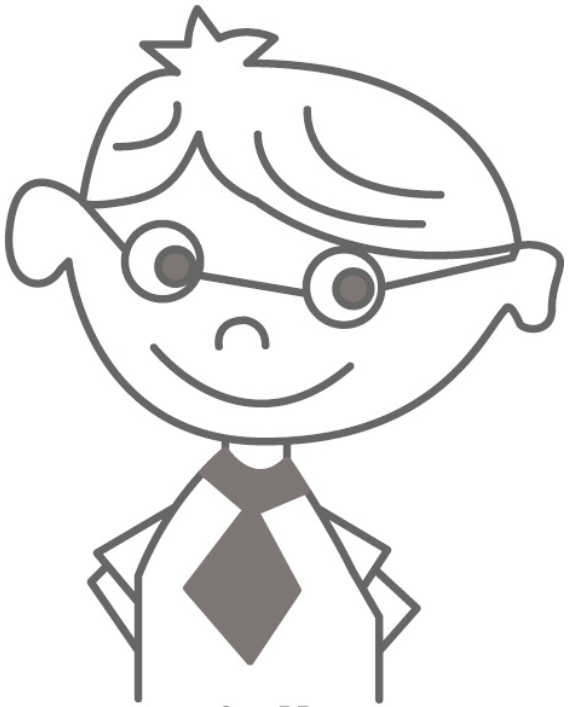
/pub/scm / linux/kernel/git/torvalds/linux-2.6.git / tree

summary | shortlog | log | commit | commitdiff | tree
snapshot

Merge branch 'for-linus' of git://git.kernel.org/pub/scm/linux/kernel/git/drzeus/mmc

[linux/kernel/git/torvalds/linux-2.6.git] /
[linux/kernel/git/torvalds/linux-2.6.git] / drivers / staging / android /

-rw-r--r--	.gitignore	blob | history | raw
-rw-r--r--	.mailmap	blob | history | raw
-rw-r--r--	COPYING	blob | history | raw
-rw-r--r--	CREDITS	blob | history | raw
drwxr-xr-x	Documentation	tree | history
-rw-r--r--	Kbuild	blob | history | raw
-rw-r--r--	MAINTAINERS	blob | history | raw
-rw-r--r--	Makefile	blob | history | raw
-rw-r--r--	README	blob | history | raw
-rw-r--r--	REPORTING-BUGS	blob | history | raw
drwxr-xr-x	arch	tree | history
drwxr-xr-x	block	tree | history
drwxr-xr-x	crypto	tree | history
drwxr-xr-x	drivers	tree | history



binder

- ☑ Android 的 IPC
- ☑ /proc/binder
  - state
  - stats
  - transactions
  - transation_log
  - failed_transation_log

Ashmem

- ☑ kernel/mm/ashmem.c
- ☑ /dev/ashmem

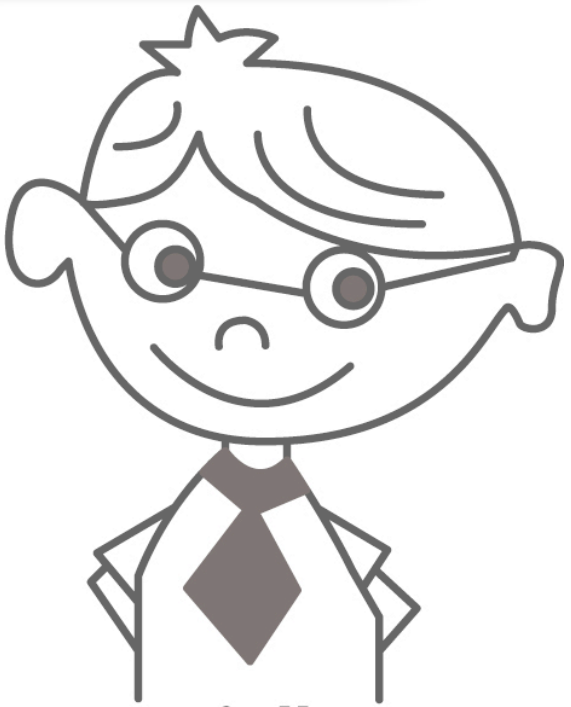
PMEM

- ☑ drivers/misc/pmem.c
- ☑ 分配實體連續空間給 userspace driver

編譯測試程式

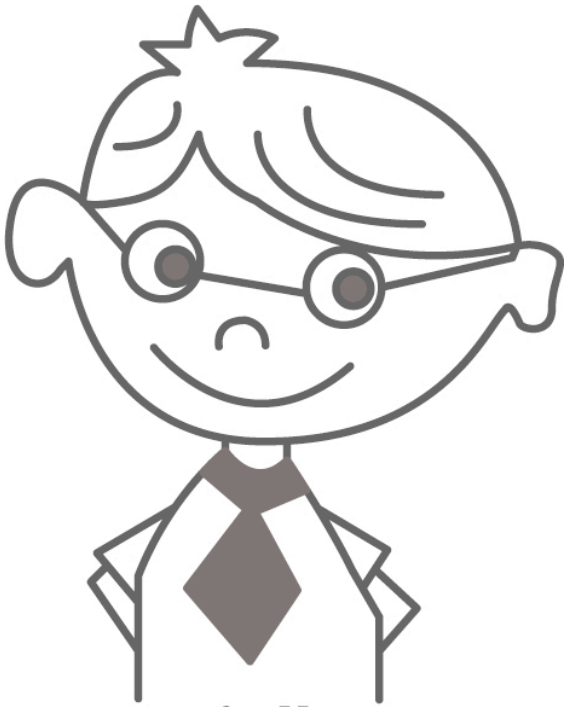
```
$ arm-eabi-gcc -o hello hello.c   -Wl,-rpath-link=./cupcake/out/
target/product/generic/obj/lib   -L./cupcake/out/target/product/
generic/obj/lib   -nostdlib  ./cupcake/out/target/product/generic/
obj/lib/crtbegin_dynamic.o   -lc
```

☑ 編譯 Android OS 的 native program

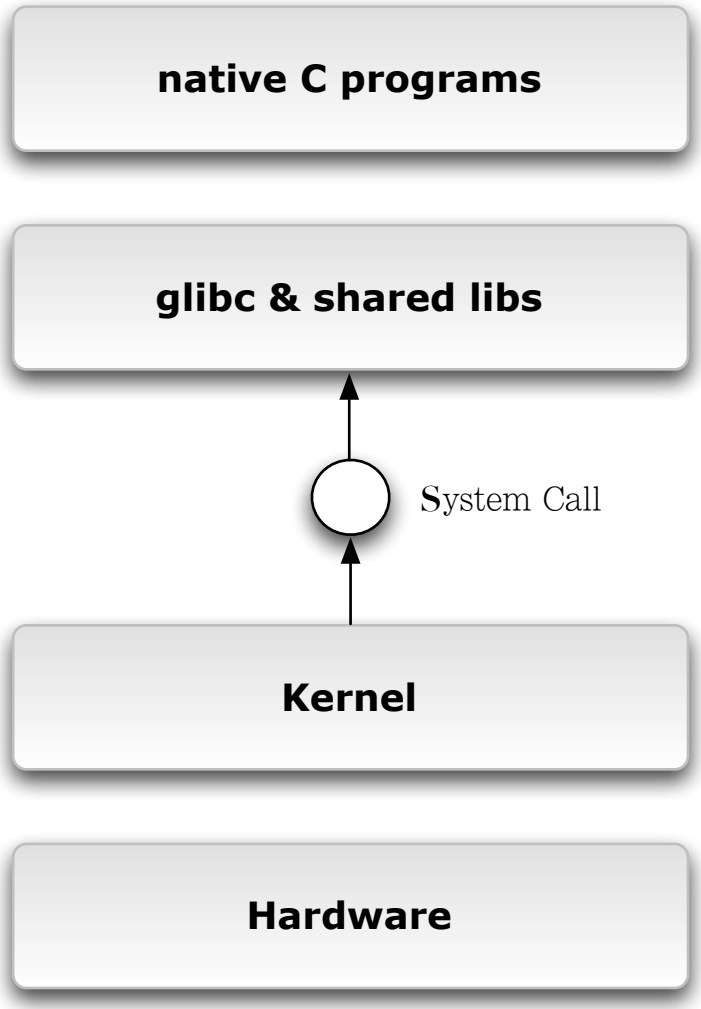


操作示範與討論

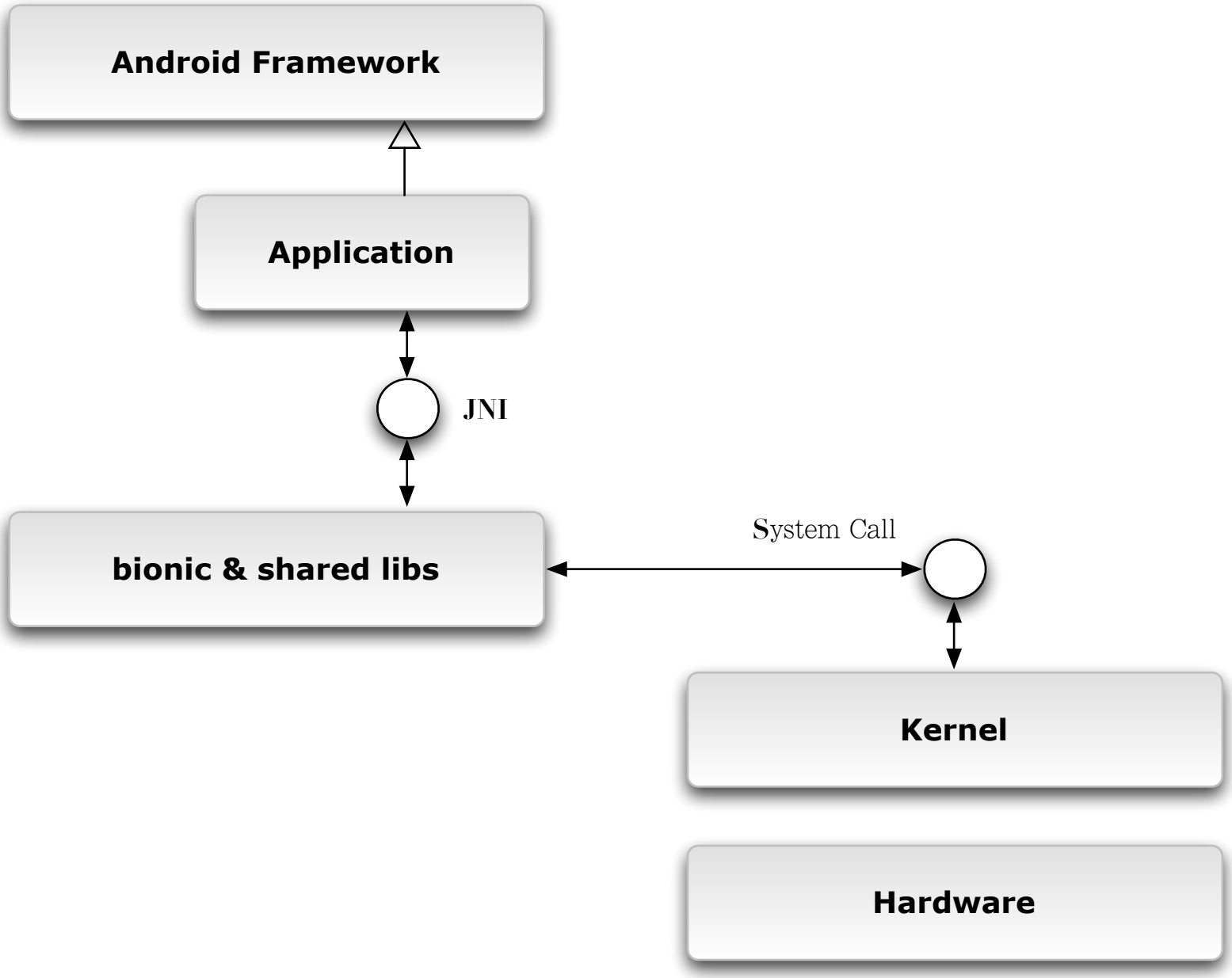
- ☑ 建立 Android 編譯環境
- ☑ 下載 Android source code (Cupcake)
- ☑ Toolchain 的使用
- ☑ 取得 EeePC 的移植



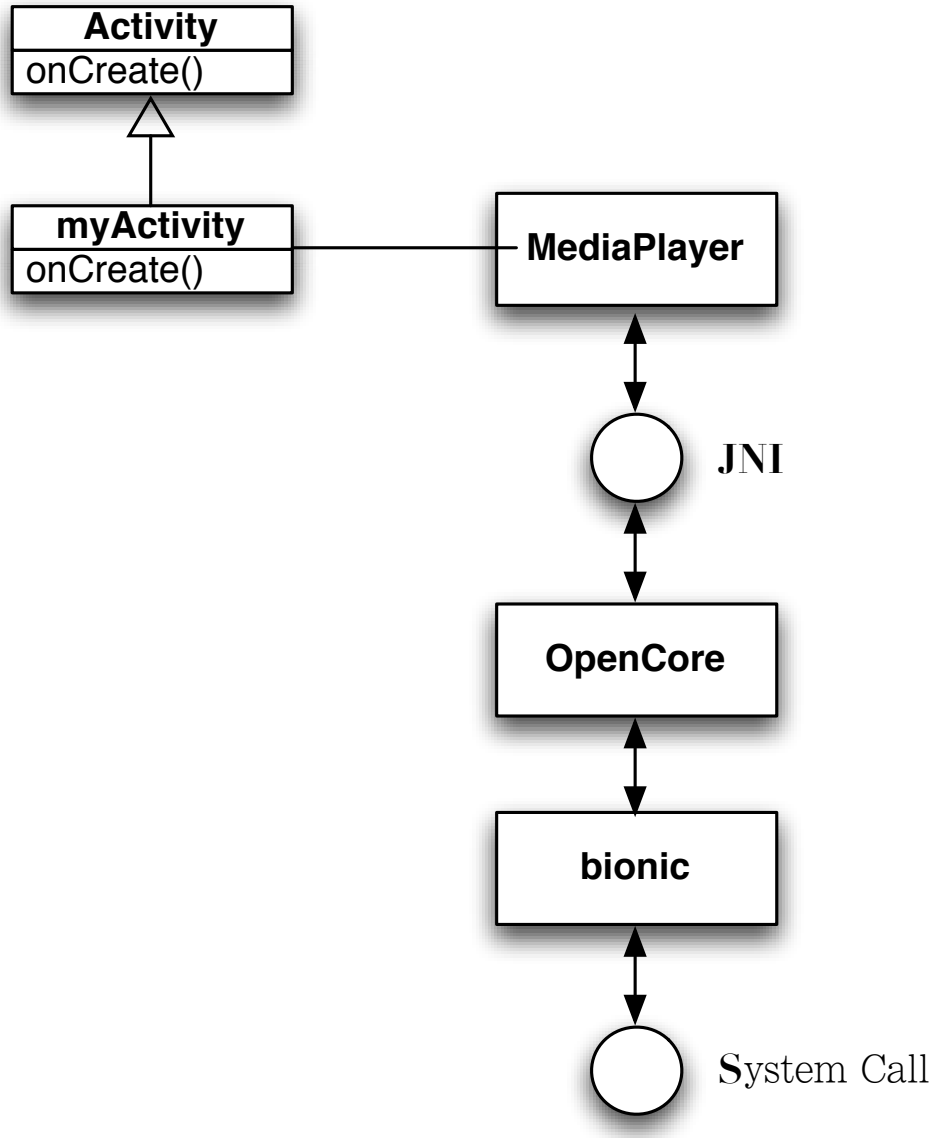
軟硬整合：Linux 模式

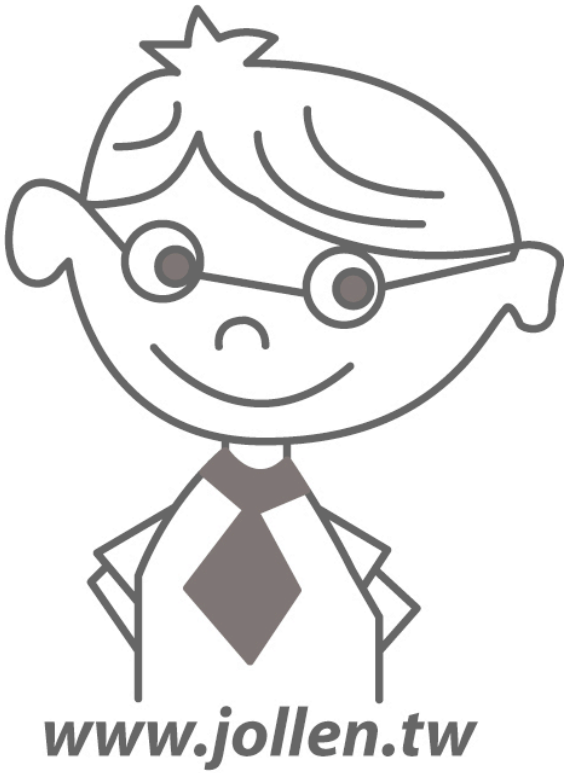


軟硬整合：Android 模式



Android 軟硬整合實作





www.jolle.tw

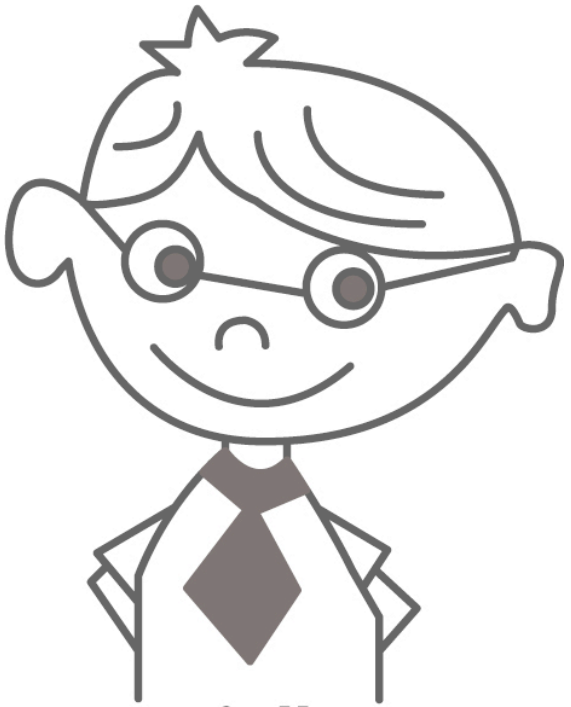
- **Section 2. Build Android、底層、實作**

android market

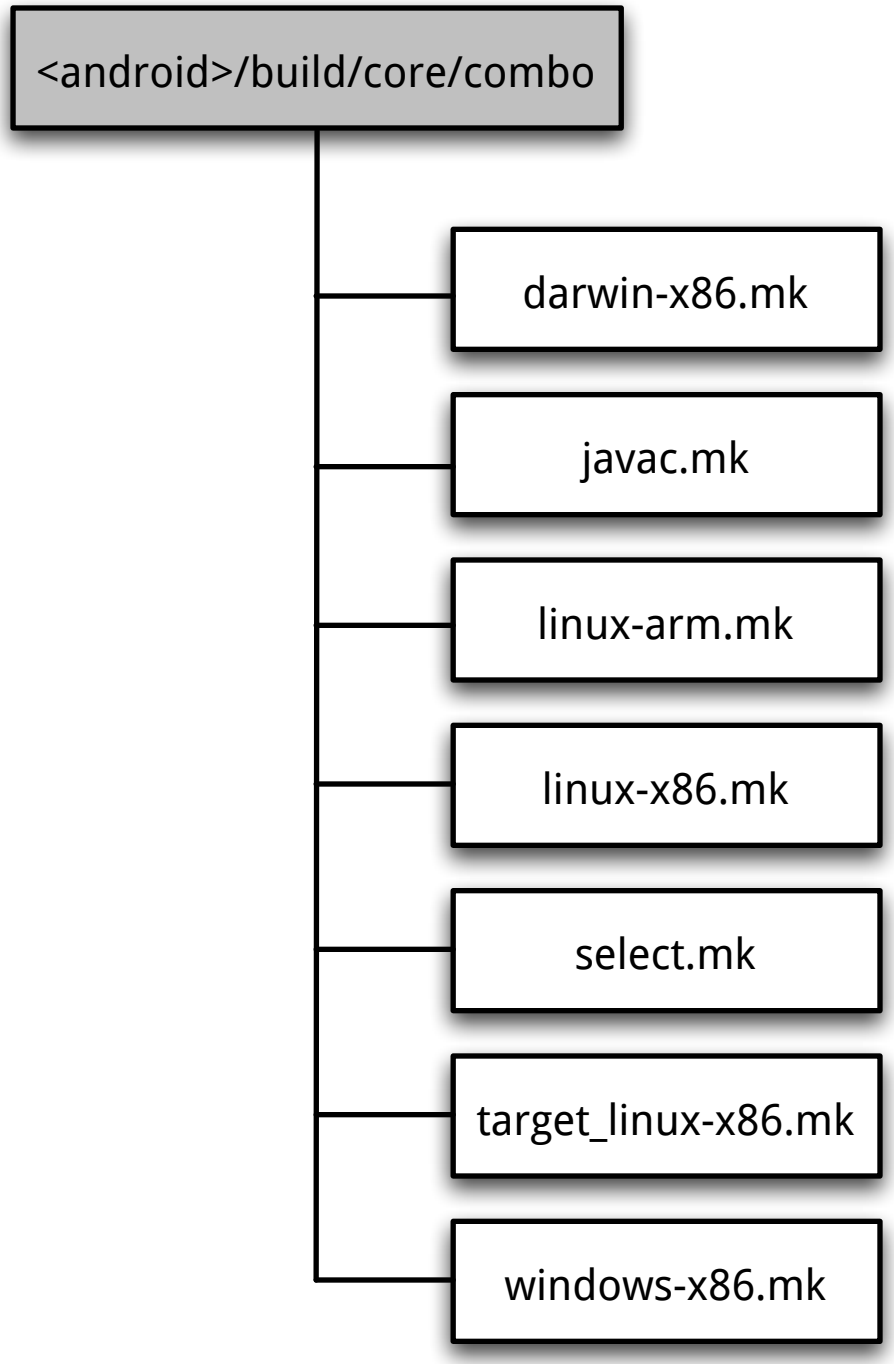
open handset alliance

Android 底層技術

- ☑ Build system (toolchain).
- ☑ Dalvik::JNI
- ☑ Dalvik::Interpreter
- ☑ Bionic
- ☑ System call



Makefile macros



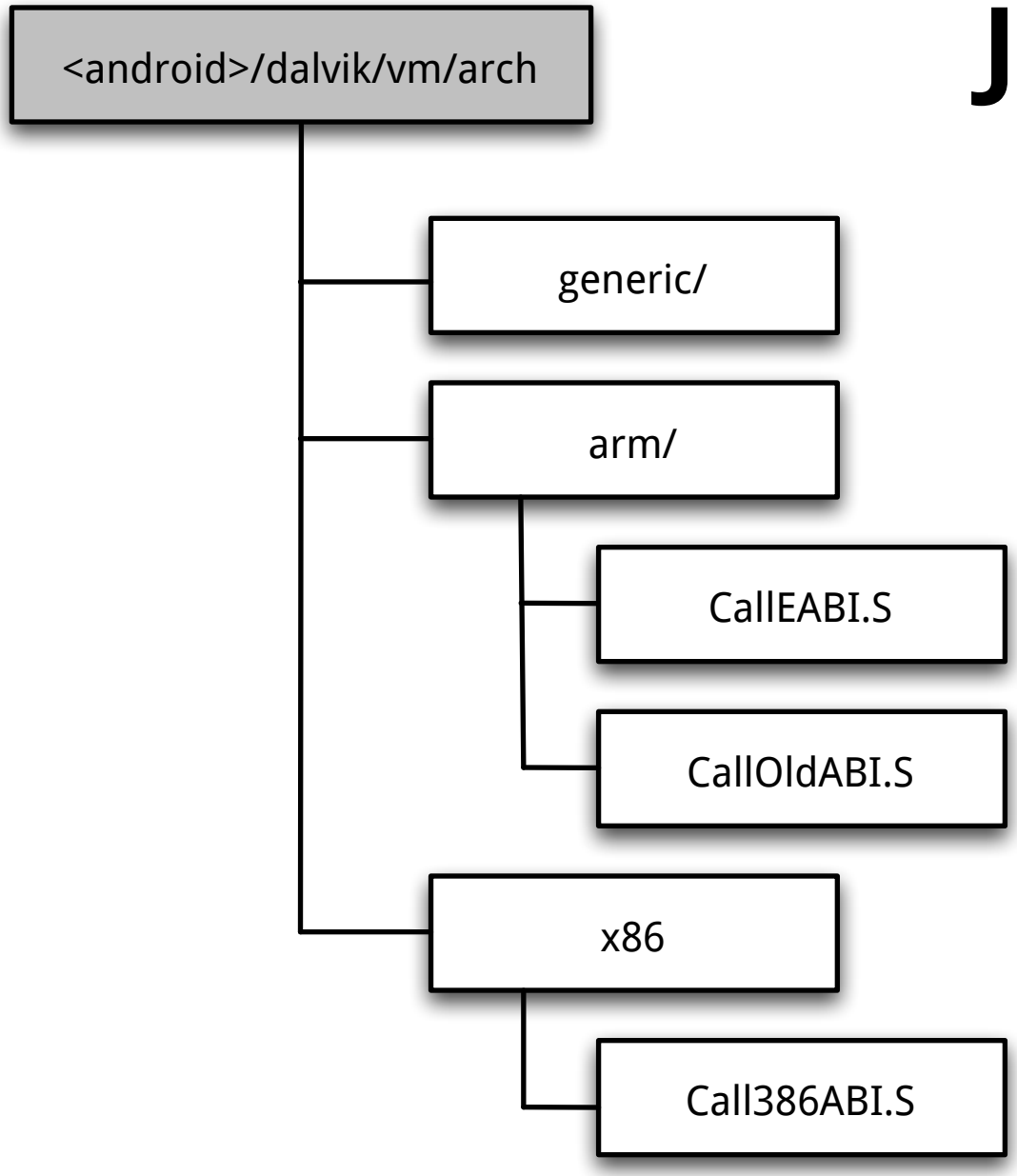
GLOBAL CFLAGS

```
$(combo_target)GLOBAL_CFLAGS += \
                -march=armv5te -mtune=xscale \
                -msoft-float -fpic \
                -mthumb-interwork \
                -ffunction-sections \
                -funwind-tables \
                -fstack-protector \
                -fno-short-enums \
                -D__ARM_ARCH_5__ -D__ARM_ARCH_5T__ \
                -D__ARM_ARCH_5E__ -D__ARM_ARCH_5TE__ \
                -include $(call select-android-config-h,linux-arm)
```

GLOBAL CFLAGS for ARMv4

```
$(combo_target)GLOBAL_CFLAGS += \
                -march=armv4t -mcpu=arm920t -mtune=xscale \
                -msoft-float -fpic \
                -mthumb-interwork \
                -ffunction-sections \
                -funwind-tables \
                -fstack-protector \
                -fno-short-enums \
                -D__ARM_ARCH_4__ -D__ARM_ARCH_4T__ \
                -D__ARM_ARCH_5E__ -D__ARM_ARCH_5TE__ \
                -include $(call select-android-config-h,linux-arm)
```

JNI Porting


<android>/dalvik/vm/arch
generic/
arm/
CallEABI.S
CallOldABI.S
x86
Call386ABI.S


www.jollen.tw

ARM EABI

r0-r3 hold first 4 args to a method
r9 is given special treatment in some situations, but not for us
r10 (sl) seems to be generally available
r11 (fp) is used by gcc (unless -fomit-frame-pointer is set)
r12 (ip) is scratch -- not preserved across method calls
r13 (sp) should be managed carefully in case a signal arrives
r14 (lr) must be preserved
r15 (pc) can be tinkered with directly

JNI Entry

r0 JNIEnv (can be left alone)
r1 clazz (NULL for virtual method calls, non-NULL for static)
r2 arg info *
r3 argc (number of 32-bit values in argv)
[sp] argv
[sp,#4] short signature
[sp,#8] func
[sp,#12] pReturn

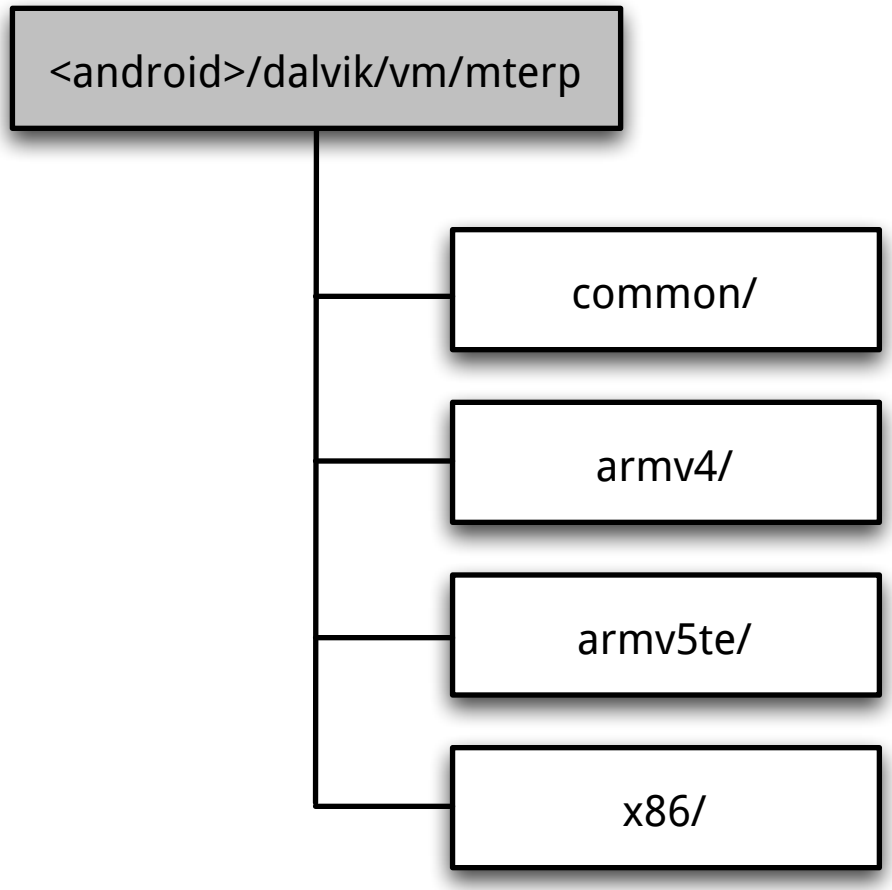
Assembly Code #1: armv4/armv5

```
.Lcopy_done:
    @ call the method
    ldr     ip, [r4, #8]              @ func
#ifndef __ARM_ARCH_4__
    blx     ip
#else
    mov     lr, pc
    bx      ip
#endif
```

Assembly Code #2: armv4/armv5

```
#ifndef __ARM_ARCH_4__
   ldmdb   r4, {r4, r5, r6, r7, r8, r9, sp, pc}
#else
   ldmdb   r4, {r4, r5, r6, r7, r8, r9, sp, lr}
   bx      lr
#endif
```

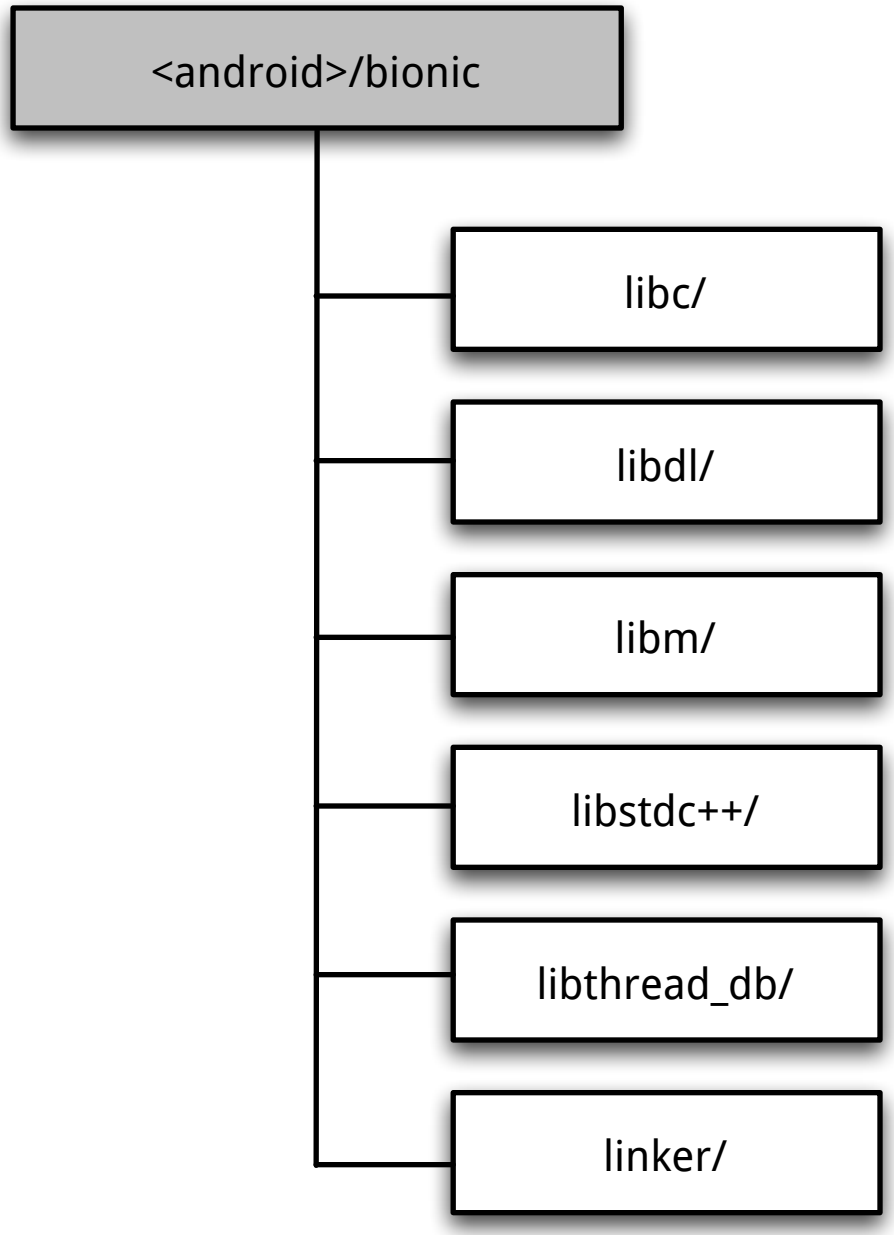
Dalvik Interpreter



Bionic

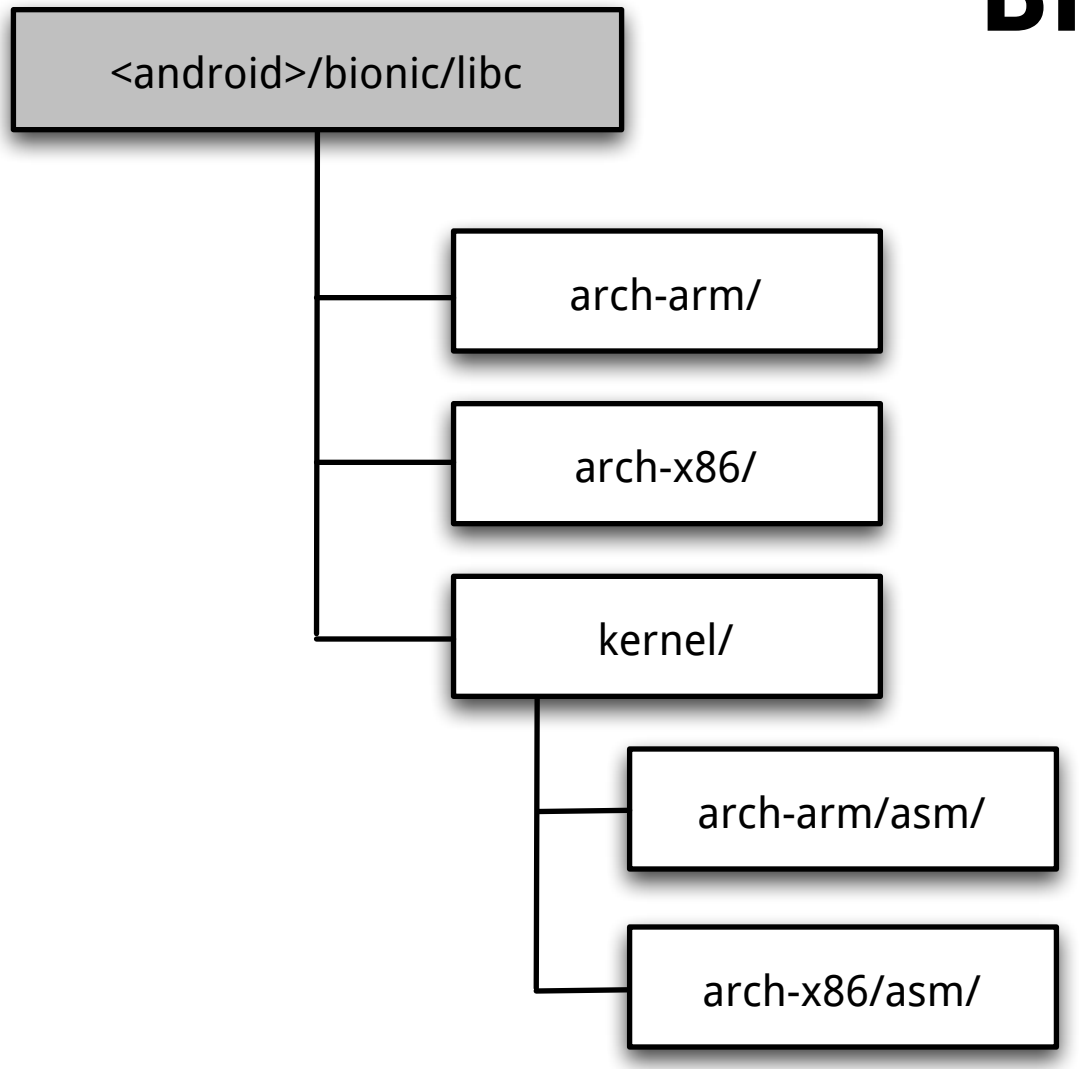
- ☑ Small and custom C library for the Android platform.
- ☑ A mainly port of BSD C library.
- ☑ Its own small implementation of pthreads based on Linux futexes.
- ☑ Support for x86, ARM and ARM thumb.

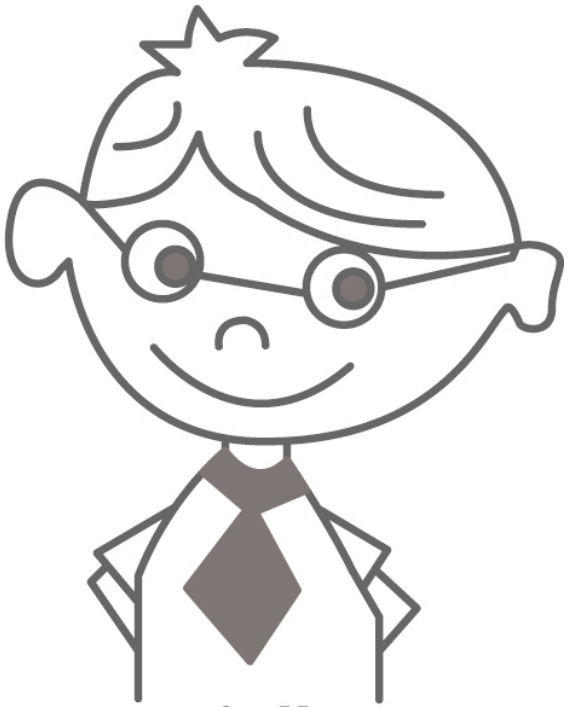
Bionic


<android>/bionic
libc/
libdl/
libm/
libstdc++/
libthread_db/
linker/


www.jollen.tw

Bionic::libc


<android>/bionic/libc
arch-arm/
arch-x86/
kernel/
arch-arm/asm/
arch-x86/asm/


www.jollen.tw

Android Generated Kernel Header

- ☑ bionic/libc/kernel/ 並非標準的 kernel header files
- ☑ Android 的 kernel header 是利用工具由 Linux kernel header 所產生的
- ☑ 目的是只保留使用到的常數、資料結構與巨集
- ☑ 給 userspace 使用

External Library

- ☑ Android 所採用的 library（shared library）
- ☑ 有些 library 的實作在特定平臺上以低階實作、效能優勢
- ☑ 例如 opencore 的實即為一例

```
~/google-android/external/opencore$ find ./ -name "*.s" -print
./codecs_v2/audio/mp3/dec/src/asm/pvmp3_polyphase_filter_window_gcc.s
./codecs_v2/audio/mp3/dec/src/asm/pvmp3_mdct_18.s
./codecs_v2/audio/mp3/dec/src/asm/pvmp3_dct_9.s
./codecs_v2/audio/mp3/dec/src/asm/pvmp3_dct_9_gcc.s
./codecs_v2/audio/mp3/dec/src/asm/pvmp3_mdct_18_gcc.s
./codecs_v2/audio/mp3/dec/src/asm/pvmp3_polyphase_filter_window.s
./codecs_v2/audio/mp3/dec/src/asm/pvmp3_dct_16_gcc.s
```

sonivox & OpenSSL

./sonivox/arm-hybrid-22k/lib_src/ARM-E_filter_gnu.s
./sonivox/arm-hybrid-22k/lib_src/ARM-E_mastergain_gnu.s
./sonivox/arm-hybrid-22k/lib_src/ARM-E_voice_gain_gnu.s
./sonivox/arm-hybrid-22k/lib_src/ARM-E_interpolate_noloop_gnu.s
./sonivox/arm-hybrid-22k/lib_src/ARM-E_interpolate_loop_gnu.s
./sonivox/arm-wt-22k/lib_src/ARM-E_filter_gnu.s
./sonivox/arm-wt-22k/lib_src/ARM-E_mastergain_gnu.s
./sonivox/arm-wt-22k/lib_src/ARM-E_voice_gain_gnu.s
./sonivox/arm-wt-22k/lib_src/ARM-E_interpolate_loop_gnu.s
./openssl/crypto/bn/asm/pa-risc2W.s
./openssl/crypto/bn/asm/pa-risc2.s
./openssl/crypto/bn/asm/mips3.s
./openssl/crypto/0.9.9-dev/aes/aes-armv4.s
./openssl/crypto/0.9.9-dev/bn/armv4-mont.s
./openssl/crypto/0.9.9-dev/sha/sha256-armv4.s
./openssl/crypto/0.9.9-dev/sha/sha512-armv4.s
./openssl/crypto/0.9.9-dev/sha/sha1-armv4-large.s

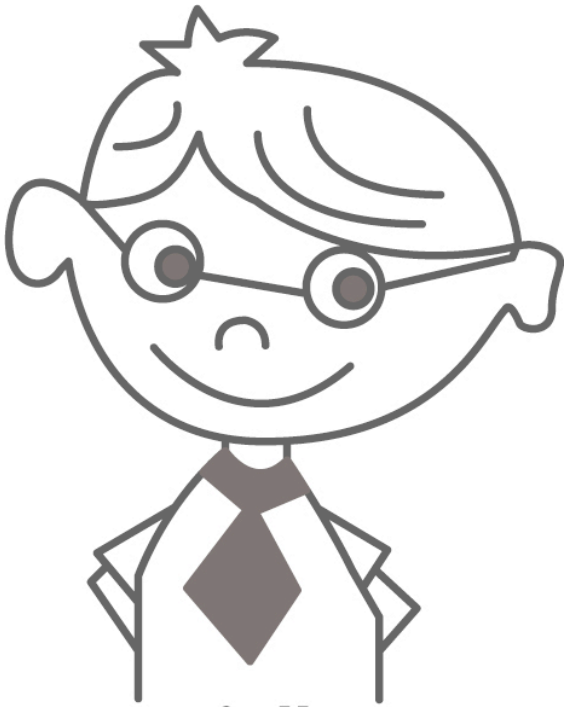
Android & Kernel 的考量項目

☑SurfaceHolder 的 type 為 SURFACE_TYPE_GPU 時、須考量GPU（Graphics Processing Unit）的支援

☑SurfaceHolder 的 type 為 SURFACE_TYPE_HARDWARE 時、須考量 DMA 與硬體加速的支援

☑目的是只保留使用到的常數、資料結構與巨集

☑給 userspace 使用



Android 多媒體支援的考量

☑Surface Manager 與 Media Framework

☑SurfaceHolder 的 type 為 SURFACE_TYPE_GPU 時、須考量GPU（Graphics Processing Unit）的支援

☑SurfaceHolder 的 type 為 SURFACE_TYPE_HARDWARE 時、須考量 DMA 與硬體加速的支援

☑MediaPlayer 使用 OpenCore 程式庫

Surface Manager

```
private SurfaceView mPreview;
private SurfaceHolder holder;

public void onCreate(Bundle icicle) {
    super.onCreate(icicle);
    setContentView(R.layout.mediaplayer_2);
    mPreview = (SurfaceView) findViewById(R.id.surface);
    holder = mPreview.getHolder();
    holder.addCallback(this);
    holder.setType(SurfaceHolder.SURFACE_TYPE_PUSH_BUFFERS);
}
```

☑ Surface Manager 部份要考量底層硬體

Media Framework

```
private MediaPlayer mMediaPlayer;

 public void surfaceCreated(SurfaceHolder holder) {
        mMediaPlayer = new MediaPlayer();
        mMediaPlayer.setDataSource(path);
        mMediaPlayer.setDisplay(holder);
        mMediaPlayer.prepare();
        mMediaPlayer.setOnBufferingUpdateListener(this);
        mMediaPlayer.setOnCompletionListener(this);
        mMediaPlayer.setOnPreparedListener(this);
        mMediaPlayer.setAudioStreamType(AudioManager.STREAM_MUSIC);
}
```

☑ Media Framework 考量 OpenCore 的移植工作

New Product File Tree

<company_name>
    <board_name>
        + Android.mk
        + product_config.mk
        + system.prop
    products
        + AndroidProducts.mk
        + <first_product_name>.mk
        + <second_product_name>.mk

Android x86 port (target product = EeePC 701)

1. Get Google Android

$ repo init -u git://android.git.kernel.org/platform/manifest.git -b cupcake
$ repo sync

2. Manifest file.

<manifest>
...
<project name="platform/vendor/asus/eee_701" \ path="vendor/asus/eee_701"/>
...
</manifest>

3. Get EeePC platform.

$ repo sync

4. Build Android image.

$ TARGET_ARCH=x86 TARGET_PRODUCT=eee_701
DISABLE_DEXPREOPT=true make -j2 installer_img

5. Create USB boot stick.

Use make-live script. https://review.source.android.com/Gerrit#change,6475

Build EeePC 701 Product Tips # Google API issue

$(call inherit-product, $(SRC_TARGET_DIR)/product/**generic.mk**)

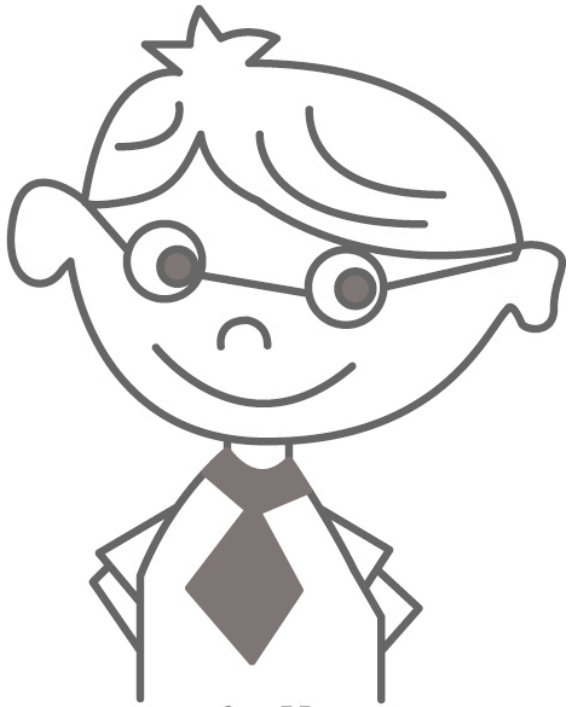
```
PRODUCT_NAME := eee_701
PRODUCT_DEVICE := eee_701
PRODUCT_POLICY := android.policy_mid
PRODUCT_PROPERTY_OVERRIDES += \
        ro.com.android.dataroaming=true
```

☑ Cupcake 將 Google APIs 放到 add-ons

Build EeePC 701 Product Tips # e2fsprogs issue

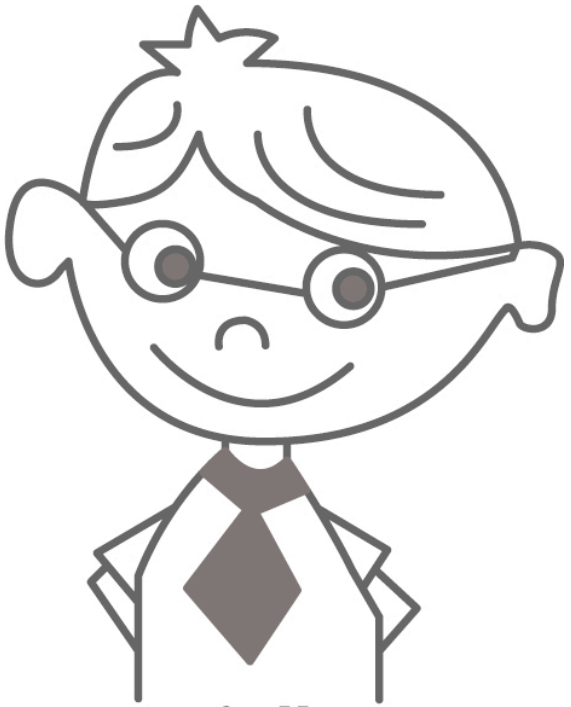
external/e2fsprogs/Android.mk:

```
--- a/Android.mk
+++ b/Android.mk
@@ -1,3 +1,3 @@
ifneq ($(TARGET_SIMULATOR),true)
-# include $(call all-subdir-makefiles)
+ include $(call all-subdir-makefiles)
endif
```



Android Image Files

- ☑ boot.img
- ☑ installer.img
- ☑ ramdisk.img
- ☑ system.img
- ☑ userdata.img



建立 Android 開機隨身碟

```
$ make-live
```

android
open source project

Change 6475: Adding a preliminary script that converts an USB installer.img into a "live" usb stick that can be b

Description

Owner	Chris Elford
Project	platform/vendor/asus/eee_701
Branch	master
Uploaded	Dec 30, 2008 11:24 AM
Updated	Jan 13, 2009 7:21 AM
Status	Review in Progress

Permalink

Adding a preliminary script that converts an USB installer.img into a "live" usb stick that can be booted directly

Dependencies

Approvals

Reviewer	*Verified*	*Code Review*
Alexey Tarasov		

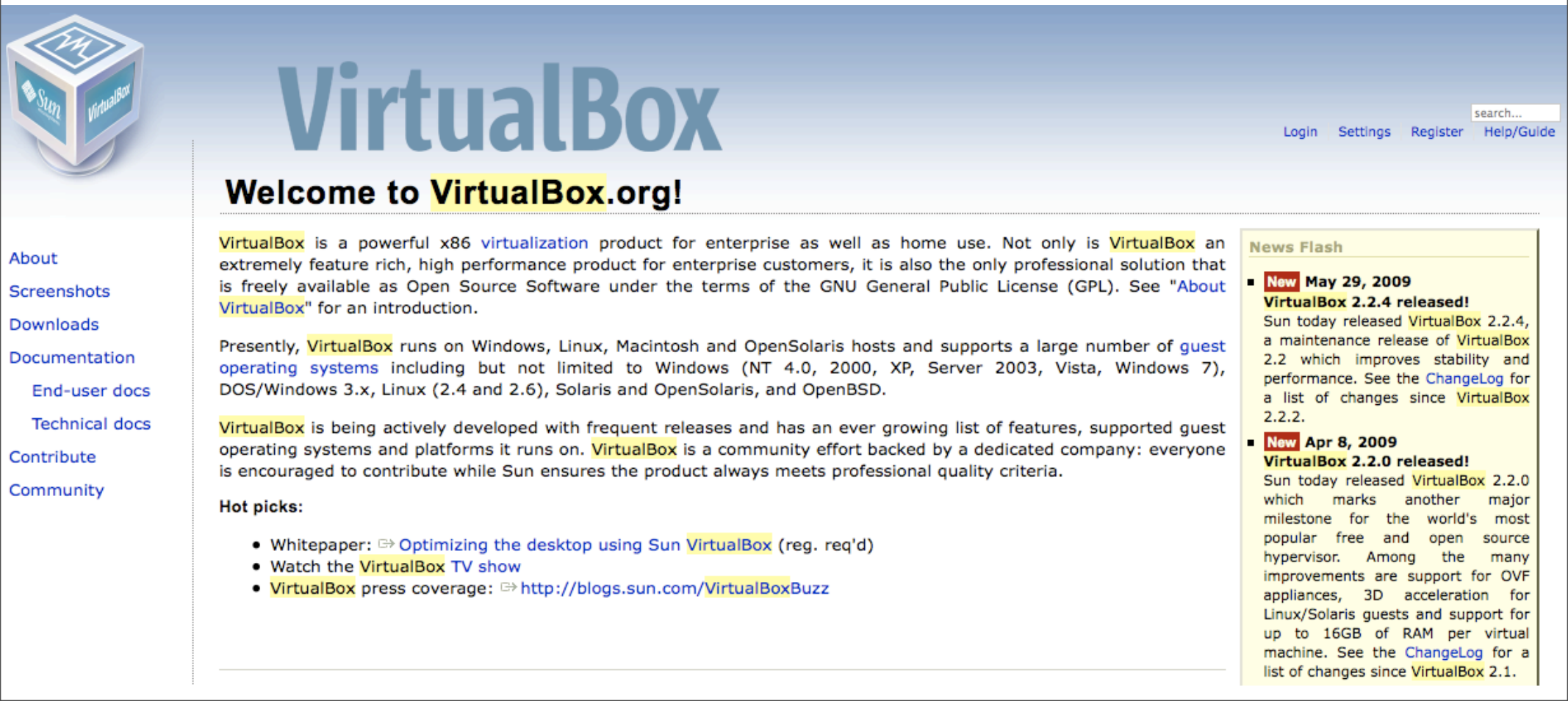
Need Verified
Need Code Review

使用 VirtualBox

- ☑ AMD PCnet32 PCI support
- ☑ VESA VGA graphics support
- ☑ VGA 8x8 font
- ☑ VGA 8x16 font

轉換 installer.img 為 VDI 格式

```
$ VBoxManage convertromraw -format VDI ./installer.img ./android.vdi
```



Android Kernerl Configs

```
#
# Android
#
# CONFIG_ANDROID_GADGET is not set
# CONFIG_ANDROID_RAM_CONSOLE is not set
CONFIG_ANDROID_POWER=y
CONFIG_ANDROID_POWER_STAT=y
CONFIG_ANDROID_LOGGER=y
# CONFIG_ANDROID_TIMED_GPIO is not set
CONFIG_ANDROID_BINDER_IPC=y
CONFIG_ANDROID_ASHMEM=y
```

Android Init Process

- ☑ device/system/init
- ☑ device/system/init/init.c
- ☑ /etc/init.rc
- ☑ 自動 mount file system (不需要 /etc/fstab)

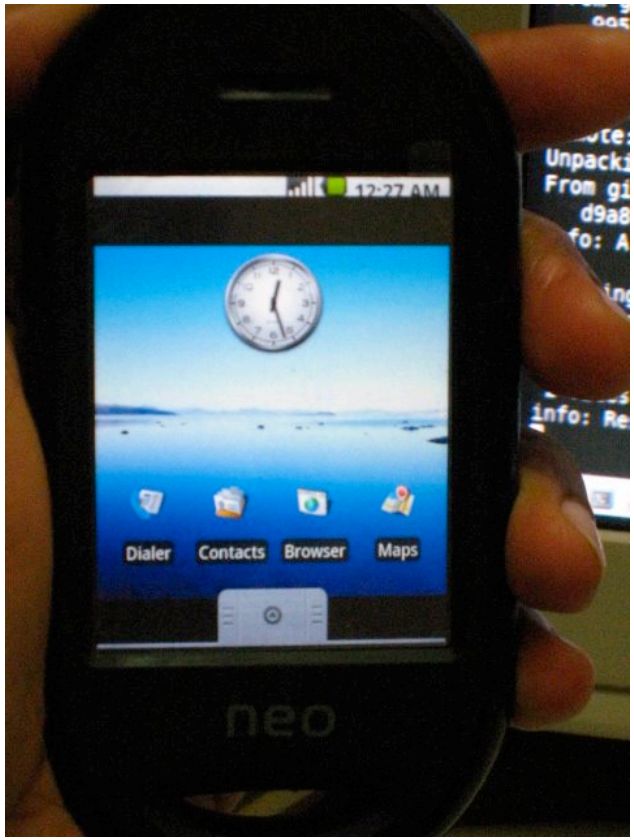
Running Applications

- ☑ /system/bin/logd
- ☑ /sbin/adbd
- ☑ /system/bin/usbd
- ☑ /system/bin/debuggerd
- ☑ /system/bin/rild
- ☑ /system/bin/app_process
- ☑ /system/bin/runtime
- ☑ /system/bin/dbus-daemon
- ☑ system_server

Zygote Process Startup

```
zygote {
    exec /system/bin/app_process
    args {
        0 -Xzygote
        1 /system/bin
        2 --zygote
    }
    autostart 1
}
runtime {
    exec /system/bin/runtime
    autostart 1
}
```

Android FreeRunner



- ☑ 支援 s3c2410 / s3c244x
- ☑ 相容於 Neo FreeRunner
- ☑ 即將支援 Mokofly (coming...)

android-way.com

首頁 Android 專欄 Android 課程 報名方式 關於...

打造與眾不同的 UI

Android 產品,觀點,解決方案 更多說明

Android-way 最新文章與動態

什麼是對話盒 (Dialog)？如何建立對話盒？

自從有了圖形化應用程式之後，對話盒（dialog）一直是元老級的元件（widget）；智慧型手機開始流行後，對話盒仍然是手機介面的重要圖形元件 More...

快顯訊息 android.widget.Toast

這是一個很好用的類別，特別是在初步建立Android應用程式的控制或行為時，可以輔助我們進行初步的測試工作 More...

如何建立選單 Menu

android-way.com
www.jollen.tw

FAQ

陳俊宏 Jollen Chen <jollen@jollen.org>

- 資深Embedded Linux顧問與講師，在Embedded Linux以及Linux驅動程式方面有豐富的經驗；同時也負責Openmoko在大中華區的行銷與推廣，在過去二年，帶領Openmoko深耕教育市場，對開放手機平臺的教育與推廣不遺餘力。目前則是專注於Android OS底層的技術研究，以及提供Android專案設計服務

- Jollen的部落格 - www.jollen.org/blog

- Jollen的噗浪 - www.plurk.com/jollenchen