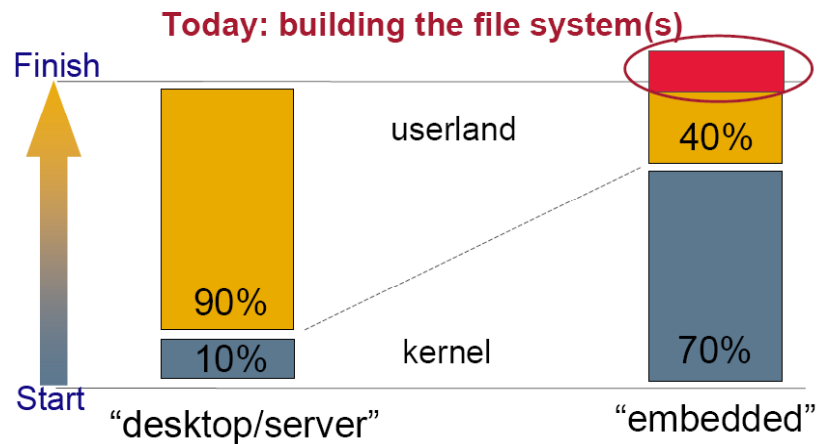


# 임베디드 리눅스의 사용자 공간과 안드로이드

연세대학교 컴퓨터정보통신공학부  
윤상균

- BusyBox
- 웹서버
- GUI
  
- Android

## Kernel Space vs. User Space



## User Space

- 응용 프로그램을 위한 POSIX 환경
- 주요 컴포넌트
  - Program – init, shell
  - Libraries – libc
  - Configuration files – in /etc
  - Device nodes – in /dev
  - User data – in /home

## Root file system

- kernel boot 동안 mount 됨
- 다음 장치에서 root file system이 mount 됨
  - RAM disk (initramfs)
  - storage device – flash memory, hard disk
  - network – NFS
- Root file system 통합 빌드 도구
  - Buildroot – <http://buildroot.uclibc.org>
  - OpenEmbedded – <http://www.openembedded.org>

## BusyBox

"The Swiss Army Knife of Embedded Linux"

- Combines tiny versions of many common UNIX utilities into a single small executable
- Fewer options than their full-featured GNU cousins
- Size-optimization and limited resources
- Provides a fairly complete environment for any small or embedded system
  
- Extremely modular and easy to customize
- <http://www.busybox.net>
- now BusyBox 1.21.0

## BusyBox or Full tools ?

### Busybox

479 kB

Contains 177 commands:

```
addgroup adduser ash cat chgrp chmod chown cp cpio date dd
delgroup deluser df dmesg echo egrep false fgrep grep gunzip gzip
hostname ip ipcalc kill ln login ls mkdir mknod mktemp more mount
mt mv netstat nice pidof ping ping6 ps pwd rm rmdir run-parts sed sh
sleep stty su sync tar touch true umount uname usleep vi watch zcat
linuxrcdevfsd fdisk getty halt hdparm hwclock ifconfig ifdown ifup init
insmod klogd loadkmap losetup lsmod makedevs mkswap modprobe
nameif pivot_root poweroff reboot rmdir route start-stop-daemon
sulogin swapoff swapon syslogd vconfig [ [ arping awk basename
bunzip2 bzip2 chvt clear cmp crontab cut dc deallocvt dimame
dos2unix du env expr find fold free ftpget ftpput head hexdump
hostid id install killall last length logger logname md5sum msg
mkfifo nc nslookup od openvt passwd patch printf readlink realpath
renice reset rpm2cpio rx seq sha1sum sort strings tail tee telnet test
tftp time top tr traceroute tty uniq unix2dos unzip uptime uuencode
uuencode vlock wc wget which who whoami xargs yes chroot crond
fbset httd inetd rdate telnetd
```

sh	312k (tcsh), 656k (bash), 86k (dash)
cp	55k
grep	105k
login	34k
mkdosfs	24k
mkfifo	17k
mkfs.ext3	39k
mount	78k
mv	63k
nice	18k
rm	38k
setserial	20k
sleep	18k
stty	42k
getty	15k (agetty), 93k (mgetty)
ifconfig	61k
vi	352k (nvi), 1003k (vim)
-----	
TOTAL	1291 kB

```
[root@WENDERS bin]# busybox
BusyBox v1.01 (2007.08.10-05:47:0000) multi-call binary
```

```
Usage: busybox [function] [arguments]...
or: [function] [arguments]...
```

BusyBox is a multi-call binary that combines many common Unix utilities into a single executable. Most people will create a link to busybox for each function they wish to use and BusyBox will act like whatever it was invoked as!

Currently defined functions:

```
[, addgroup, adduser, adjtimex, ar, arping, ash, awk, basename,
bunzip2, busybox, bzip2, cal, cat, chgrp, chmod, chown, chroot,
chvt, clear, cmp, cp, cpio, crond, crontab, cut, date, dc, dd,
deallocvt, delgroup, deluser, df, dirname, dmesg, dos2unix, du,
dumkmap, echo, egrep, env, expr, false, fbset, fdisk, fgrep,
find, fold, free, ftpget, ftpput, getopt, grep, gunzip, gzip,
halt, head, hexdump, hostid, hostname, hwclock, id, ifconfig,
ifdown, ifup, inetd, init, insmod, install, kill, killall, klogd,
last, length, ln, loadfont, loadkmap, logger, login, logname,
logread, losetup, ls, lsmod, makedevs, md5sum, msg, mkdir, mkfifo,
mknod, mkswap, mktemp, modprobe, more, mount, mt, mv, netstat,
nslookup, od, openvt, passwd, patch, pidof, ping, poweroff, printf,
ps, pwd, readlink, realpath, reboot, renice, reset, rm, rmdir,
rmdir, route, rpm, rpm2cpio, run-parts, rx, sed, setkeycodes,
sh, sha1sum, sleep, sort, start-stop-daemon, strings, stty, su,
sulogin, swapoff, swapon, sync, syslogd, tail, tar, tee, telnet,
telnetd, test, tftp, time, top, touch, tr, traceroute, true, tty,
udhcpc, umount, uname, uncompress, uniq, unix2dos, unzip, uptime,
usleep, uuencode, uuencode, vi, watch, wc, wget, which, who, whoami,
xargs, yes, zcat
```

## BusyBox의 실행

- BusyBox의 인수에 명령어 이름을 제공

```
[root@WENDERS /]# busybox ls
bin      home     lost+found  root      srv      usr
dev      info     media       sbin     sys      var
etc      lib      proc        share    tmp
```

- 실행시키려는 명령어의 이름을 busybox와 심볼릭 링크하여 사용
  - 설치할 때에 자동적으로 수행됨
  - /bin, /sbin, /usr/bin, /usr/sbin 에 위치한 상당수의 명령어들이 /bin/busybox와 심볼릭 링크되어 있음

```
[root@WENDERS bin]# ls -l
lrwxrwxrwx 1 root root 9 Jul 11 2008 addgroup -> tinylogin
lrwxrwxrwx 1 root root 9 Jul 11 2008 adduser -> tinylogin
-r-xr-xr-x 1 root root 5496 Mar 25 2008 alsalisp
lrwxrwxrwx 1 root root 7 Jul 11 2008 ash -> busybox
-r-xr-xr-x 1 root root 803540 Mar 25 2008 bash
-r-xr-xr-x 1 root root 1491064 Mar 25 2008 busybox
lrwxrwxrwx 1 root root 7 Jul 11 2008 cat -> busybox
-r-xr-xr-x 1 root root 8476 Mar 25 2008 chattr
lrwxrwxrwx 1 root root 7 Jul 11 2008 chgrp -> busybox
lrwxrwxrwx 1 root root 7 Jul 11 2008 chmod -> busybox
lrwxrwxrwx 1 root root 7 Jul 11 2008 chown -> busybox
lrwxrwxrwx 1 root root 7 Jul 11 2008 cp -> busybox
lrwxrwxrwx 1 root root 7 Jul 11 2008 cpio -> busybox
lrwxrwxrwx 1 root root 7 Jul 11 2008 date -> busybox
lrwxrwxrwx 1 root root 7 Jul 11 2008 dd -> busybox
lrwxrwxrwx 1 root root 9 Jul 11 2008 delgroup -> tinylogin
lrwxrwxrwx 1 root root 9 Jul 11 2008 deluser -> tinylogin
lrwxrwxrwx 1 root root 7 Jul 11 2008 df -> busybox
lrwxrwxrwx 1 root root 7 Jul 11 2008 dmesg -> busybox
lrwxrwxrwx 1 root root 7 Jul 11 2008 dumpkmap -> busybox
lrwxrwxrwx 1 root root 7 Jul 11 2008 echo -> busybox
lrwxrwxrwx 1 root root 7 Jul 11 2008 egrep -> busybox
lrwxrwxrwx 1 root root 7 Jul 11 2008 false -> busybox
lrwxrwxrwx 1 root root 7 Jul 11 2008 fgrep -> busybox
lrwxrwxrwx 1 root root 7 Jul 11 2008 getopt -> busybox
lrwxrwxrwx 1 root root 7 Jul 11 2008 grep -> busybox
```

```
[root@WENDERS etc]# gzip --help
BusyBox v1.01 (2007.08.10-05:47+0000) multi-call binary

Usage: gzip [OPTION]... [FILE]...

Compress FILE(s) with maximum compression.
When FILE is '-' or unspecified, reads standard input.  Implies -c.

Options:
  -c      Write output to standard output instead of FILE.gz
  -d      Decompress
  -f      Force write when destination is a terminal
```

## BusyBox Startup Mechanism

- BusyBox는 init로도 사용할 수 있다. (/sbin/init)
- System V approach
  - /etc/init.d/rcX.d/\*
  - /etc/inittab
    - runlevels – 0: halt, 1: single user, 2-5: multi user, 6: reboot
    - many options like wait, respawn, powerwait/powerfail
    - scripts usually require sed, grep, awk present
    - many fork/exec
- BusyBox approach
  - /etc/inittab (optional)
    - Much simpler, no concept of runlevels, will start console
  - 1 simple /etc/rcS file
  - can run other files if needed

## C library

- There are many C libraries to choose from, and some obvious criteria.
  - GNU glibc
  - Eglibc, uClibc, Newlib, Diet libc, BSD libc
- Impact is tremendous:
  - quality of C++ support
  - completeness
  - stability
  - size / configurability
  - availability for architecture
  - compiler modifications (uClibc!)
  - do not underestimate community !
    - Security / bugfixes / future

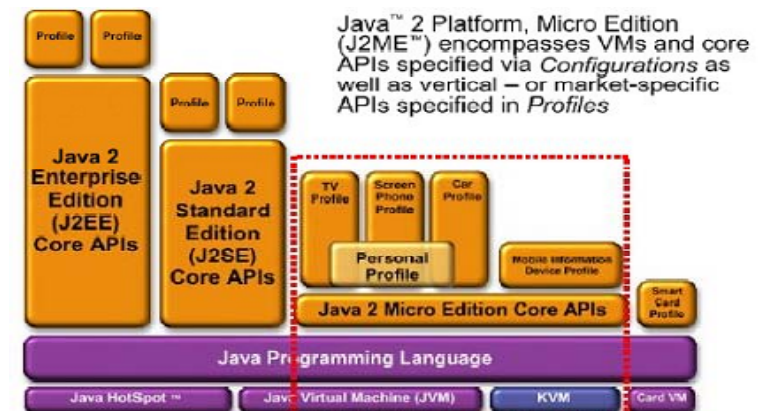
## 웹서버

- 임베디드 시스템에는 low overhead를 가진 웹서버를 주로 사용함
  - Boa
  - lighthttpd
  - Cherokee, qhttpd, nginx, Hiawatha
  - GoAhead
- cgi 프로그램을 사용하여 외부에서 임베디드 시스템의 제어를 쉽게 할 수 있도록 할 수 있다.
- GoAhead 웹서버
  - (별첨 자료 참조)

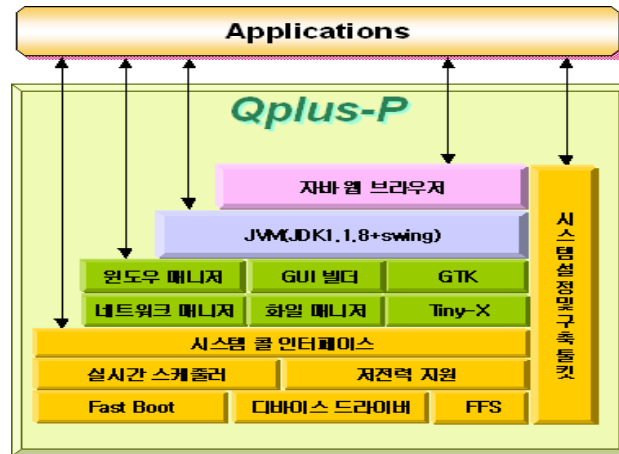
## GUI 개발

- GUI (Graphic User Interface)
  - Qt/Embedded
    - Smart and easy GUI
    - Open source, once commercialized -> royalty (별첨자료 참조)
  - GTK+
    - Open source, GPL
  - MicroWindow (now Nano-X)
  - Tiny X : down-sized X Window system (now smallX)
  - PicoGUI (now halt)
  - Nano-Qplus (ETRI)

## Java ME



## Qplus/Esto



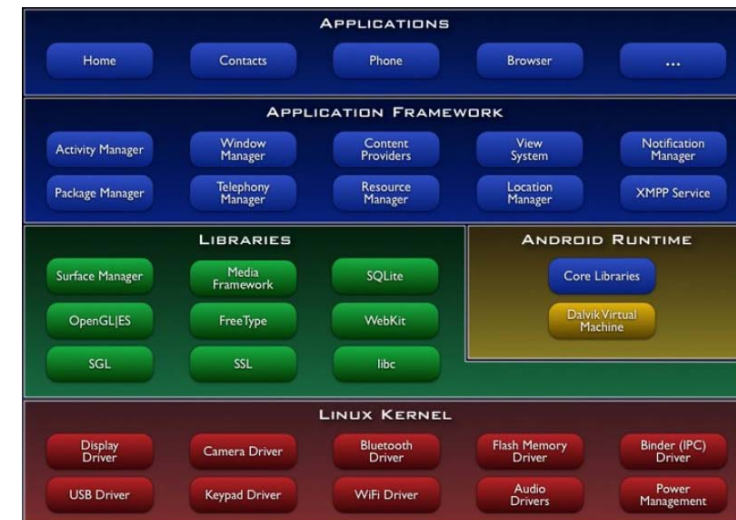
## Android

- Android is a software stack for mobile devices that includes an operating system, middleware and key applications
- Based on Linux OS (kernel 2.6.x or later)
- Using Java Programming
- Open Source

## Android Features

- Application framework enabling reuse and replacement of components
- **Dalvik virtual machine** optimized for mobile devices
- Integrated browser based on the open source **WebKit** engine
- **Optimized graphics** powered by a custom 2D graphics library; 3D graphics based on the OpenGL ES 1.0 specification (hardware acceleration optional)
- **SQLite** for structured data storage
- **Media support** for common audio, video, and still image formats (MPEG4,H.264, MP3, AAC, AMR, JPG, PNG, GIF)
- GSM Telephony (hardware dependent)
- Bluetooth, EDGE, 3G, and WiFi (hardware dependent)
- Camera, GPS, compass, and accelerometer (hardware dependent)
- Rich development environment

## Android Architecture



## Application Framework

- Developers have full access to the same framework APIs used by the core applications
- Underlying all applications is a set of services and systems, including
  - **Rich and extensible set of Views** that can be used to build an application, including lists, grids, text boxes, buttons, and even an embeddable web browser
  - **Content Providers** that enable applications to access data from other applications (such as Contacts), or to share their own data
  - **Resource Manager**, providing access to non-code resources such as localized strings, graphics, and layout files
  - **Notification Manager** that enables all applications to display custom alerts in the status bar
  - **Activity Manager** that manages the life cycle of applications and provides a common navigation backstack

## Libraries

- Android includes a set of C/C++ libraries
- Some of the core libraries are
  - System C library - a BSD-derived implementation of the standard C system library (libc), tuned for embedded Linux-based devices
  - Media Libraries - based on PacketVideo's OpenCORE
  - Surface Manager - manages access to the display subsystem and seamlessly composites 2D and 3D graphic layers from multiple applications
  - LibWebCore – web browser
  - SGL - the underlying 2D graphics engine
  - 3D libraries - an implementation based on OpenGL ES 1.0 APIs
  - FreeType - bitmap and vector font rendering
  - SQLite - a powerful and lightweight relational database engine

## Linux Kernel

- Linux version
  - 1.5 (cupcake) – 2.6.27
  - 1.6 (donut) – 2.6.29
  - 2.0/2.1 (eclair) – 2.6.29
  - 2.2 (froyo) – 2.6.32
  - 2.3 (Gingerbread) – 2.6.33 or 34
  - 3.x (Honeycomb)
  - 4.0 (Ice Cream Sandwich)
  - 4.1/4.2 (Jelly Bean)
- core system services
  - Security
  - memory management, process management
  - network stack, driver model.

## Android Runtime

- Every Android application runs in its own process
- instance of the Dalvik virtual machine
- Dalvik VM executes files in the Dalvik Executable (.dex) format

---

## Develop Android Applications

- Java Programming Background
  - Java SE
- Eclipse IDE
- Android SDK
- Android Development Tool Plugin (Option)