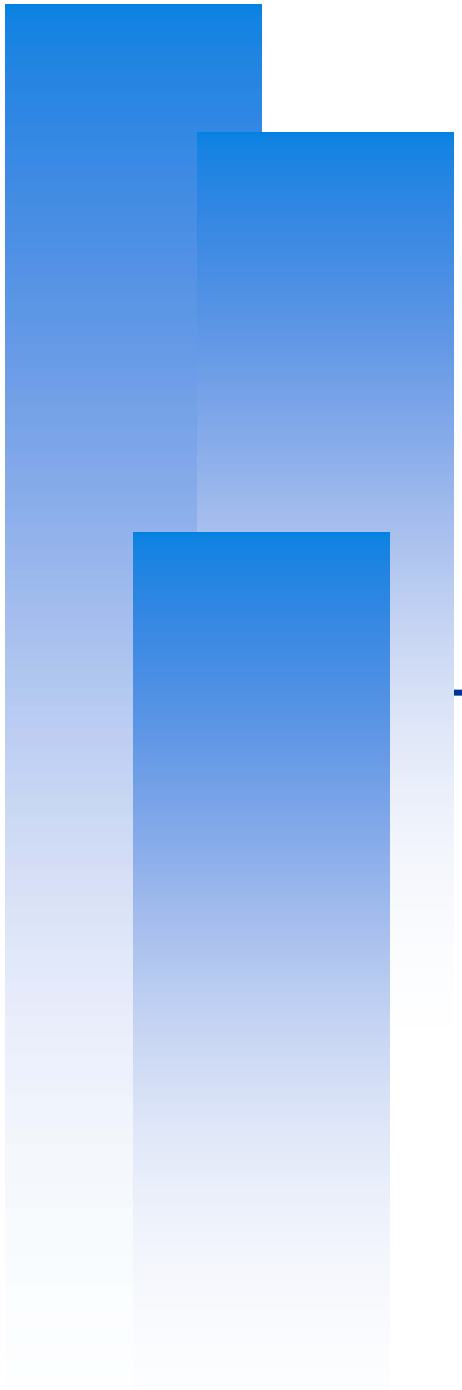
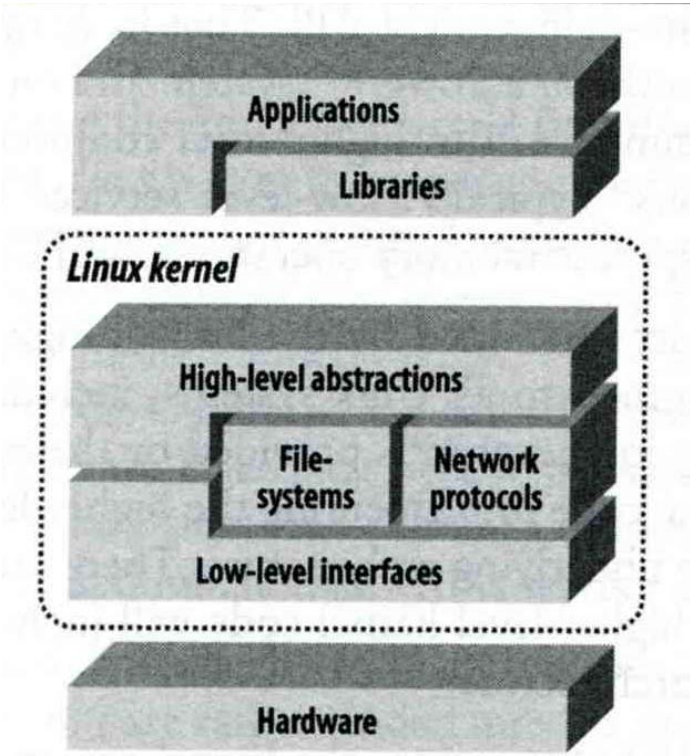


Chapter 12
Drivers and the Kernel

Roles of Kernel

- Components of a UNIX System
  - User-level programs
  - Kernel
  - Hardware
- Two roles of kernel
  - High-level abstractions
    - Process managements
    - File system management
    - Memory management
    - I/O management
  - Low-level interface
    - drivers



Kernel Types

- Two extreme types
  - **Micro kernel**
    - **Provide only necessarily, compact and small functionalities**
    - **Other functions is added via well-defined interface**
  - **Monolithic kernel (龐大的)**
    - **Whole functionalities in one kernel**
- Modern OS
  - Solaris
    - **Completely modular kernel**
    - **Load necessarily module when it is needed**
  - BSD-derived system
    - **Explicitly specify the devices on kernel compile process**
  - Linux
    - **Between BSD and Solaris System**

Kernel related directory

- Build directory and location

System	Build Directory	Kernel file
FreeBSD	/usr/src/sys	/kernel (< 4.x) /boot/kernel (> 5.x)
Red Hat	/usr/src/linux	/vmlinuz or /boot/vmlinuz
Solaris	-	/kernel/unix
SunOS	/usr/kvm/sys	/vmunix

Why configure the kernel?

- The native kernel is often big and common
- Tailoring kernel to match site situation
  - Purge unnecessary kernel devices and options
  - Add functionalities that you want
- OS patch
  - Remedy security hole of kernel implementation
- Fine-tune system performance
  - Such as adjusting important system parameters
- Adding device drivers

Building a FreeBSD Kernel

- Kernel source
  - /usr/src/sys
- Kernel configuration file
  - /usr/src/sys/i386/conf
    - GENERIC, LINT (< 4.X)
    - GENERIC, “make LINT” under this dir (> 5.x)
- Steps to build a new kernel
  - Edit /usr/src/sys/i386/conf/SABSD
  - % cd /usr/src ;
  - % make KERNCONF=SABSD buildkernel
  - % make KERNCONF=SABSD installkernel

Building a FreeBSD Kernel – Configuration file

- Each line is a control phrase

  [Ref] http://www.freebsd.org/doc/en_US.ISO8859-1/books/handbook/kernelconfig-config.html

  - Keyword + arguments

Keyword	Function	Example
machine	Sets the machine type	i386 or amd64
cpu	Sets the CPU type	I586_CPU or HAMMER
ident	Sets the name of the kernel	SABSD
maxusers	Sets the kernel's table sizes	0
options	Sets various comiple-time options	INET or INET6
device	Declares devices	fxp
Pseudo-device	Declares pseudo-devices	loop

FreeBSD 5.x~ has no Pseudo-device keyword anymore

Tuning the FreeBSD Kernel

- sysctl command
  - Dynamically set or get kernel parameters
  - All changes made by sysctl will be lost across reboot
  - Use sysctl to tune the kernel and test it, then recompile the kernel

  - Format:

    % sysctl [options] name[=value] …

  Ex:

    % sysctl –a list all kernel variables

    % sysctl –d kern.maxfiles print the description of the variable

    % sysctl kern.maxfiles print the value of the variable

    % sudo sysctl kern.maxfiles=2048