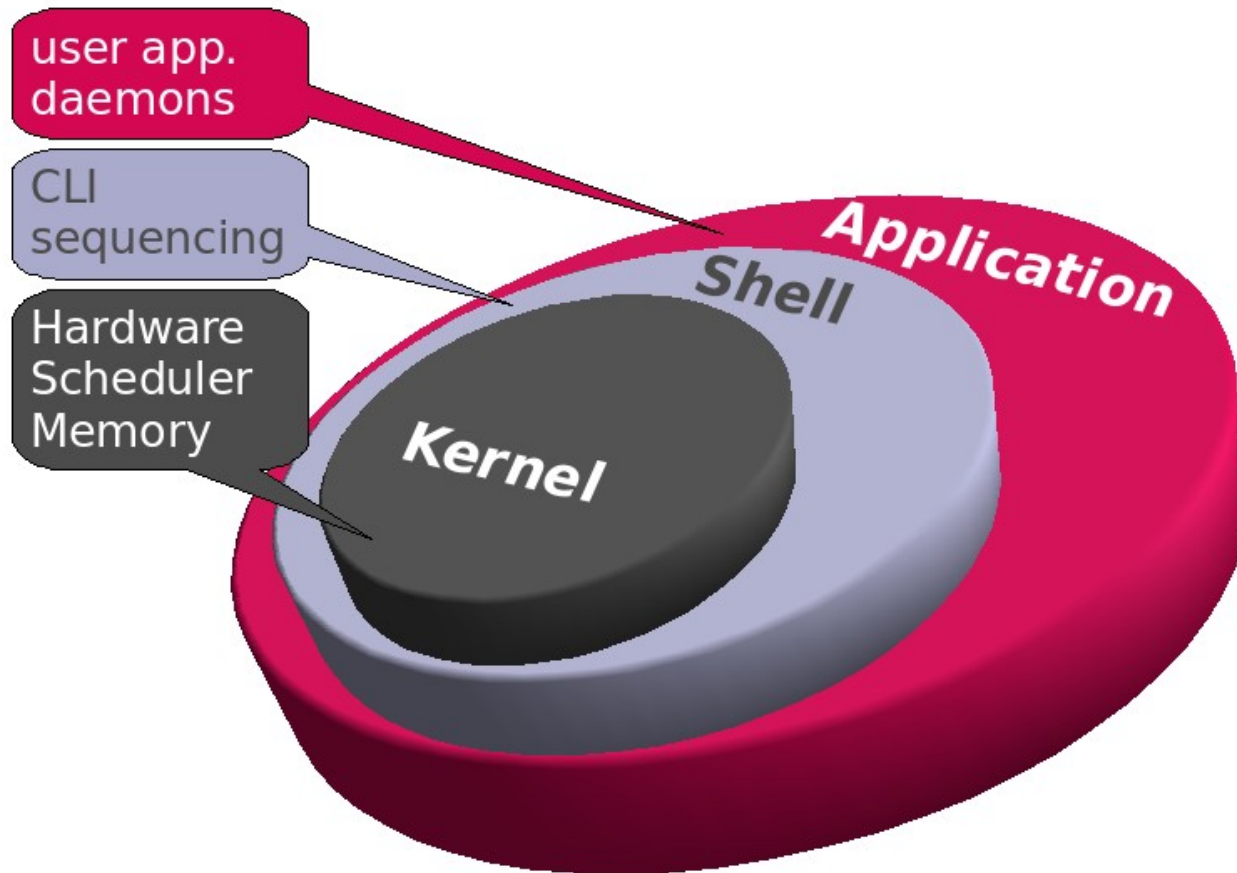


Linux Kernel & Modul

Henry Saptono, S.Si, M.Kom



Apa itu Kernel ?

- ◆ Dalam komputasi, kernel adalah sebuah program komputer yang mengelola dan mengatur permintaan Input dan Output dari perangkat lunak (software) dan menerjemahkannya kedalam instruksi instruksi pemrosesan data untuk CPU dan komponen komponen elektronik sebuah komputer.
- ◆ Kernel adalah sebuah bagian yang mendasar (bagian utama) dari sebuah sistem operasi yang pertama kali dimuat kedalam memori saat boot dan tetap dalam memori selama sistem berjalan
- ◆ Kernel bertanggung jawab untuk melakukan manajemen memori, manajemen proses dan tugas, dan manajemen disk



Kernel dan Sistem Operasi ?

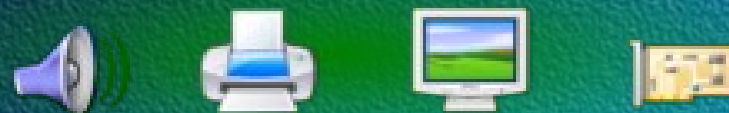
- ★ Kernel adalah bagian dari sistem operasi yang menjembatani akses ke sumber daya sistem. Kernel bertanggung jawab untuk memungkinkan beberapa perangkat lunak untuk secara efektif berbagi perangkat keras dengan mengendalikan akses ke CPU, memori, disk I/O, dan jaringan.
- ★ Sistem operasi adalah kernel yang dilengkapi dengan perangkat lunak (aplikasi) yang memungkinkan pengguna untuk dapat melakukan sesuatu (seperti shell, compiler, editor teks, window manager, dll)



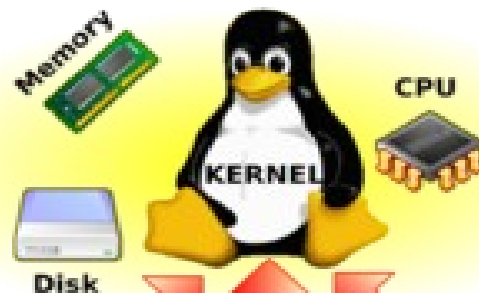
Monolithic dan micro kernel

- Kernel dapat diklasifikasikan menjadi dua kategori utama:
 - Monolithic kernel
 - Micro kernel

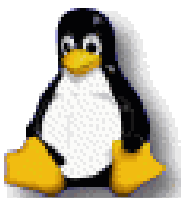




DEVICES



APPLICATIONS

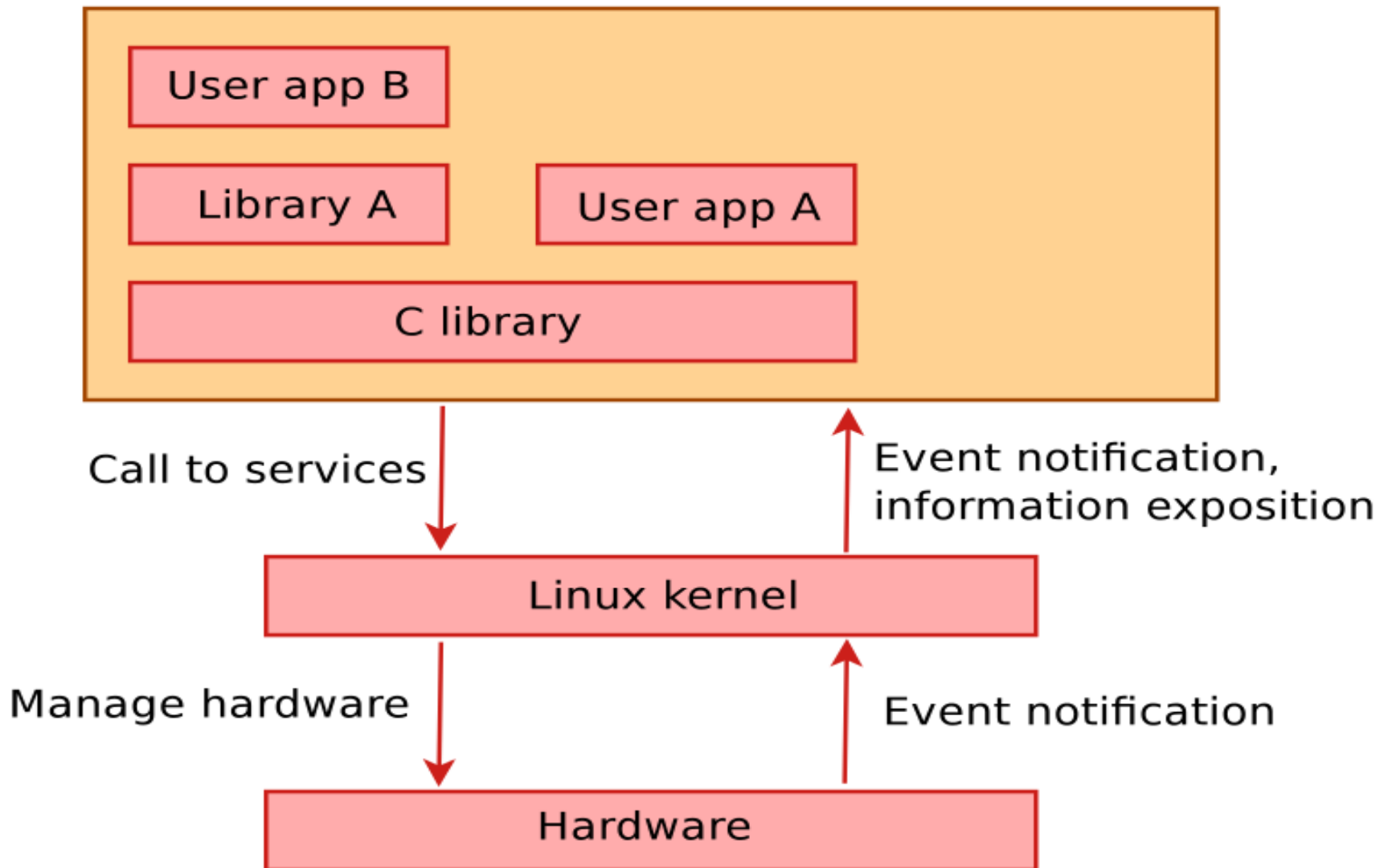


Sejarah Kernel Linux

- Kernel Linux merupakan salah satu komponen dari suatu sistem operasi, yang juga membutuhkan pustaka dan aplikasi untuk menyediakan fitur-fitur untuk pengguna akhir.
- Kernel linux diciptakan sebagai proyek hobi pada tahun 1991 oleh seorang mahasiswa Finlandia, bernama Linus Torvalds. Linux dengan cepat digunakan sebagai kernel untuk sistem operasi perangkat lunak bebas
- Linus Torvalds telah mampu membuat perkembangan para pengembang dan komunitas linux menjadi lebih besar dan dinamis. Saat ini, ratusan orang berkontribusi terhadap setiap rilis kernel linux, individu atau perusahaan besar maupun kecil.



Kernel linux dalam sistem



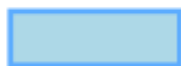
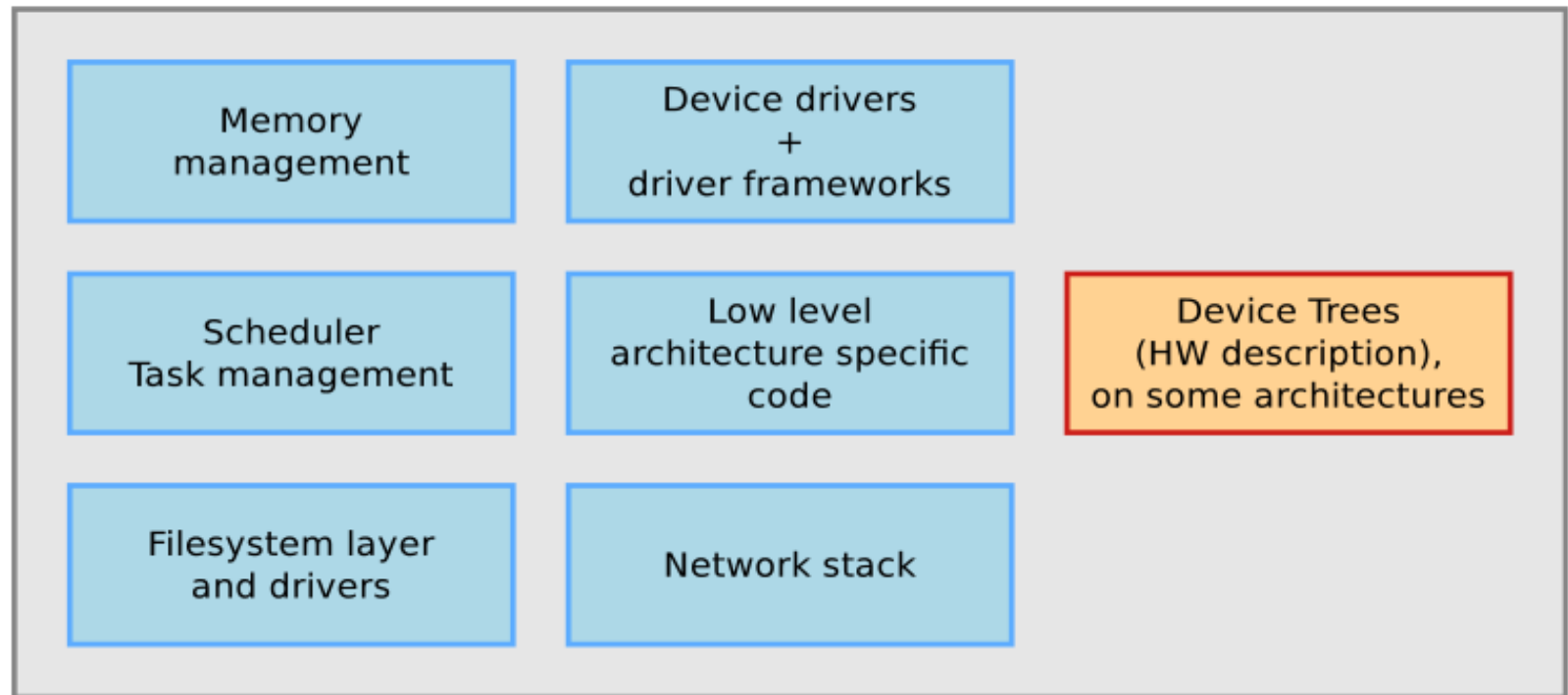
Peran utama kernel linux

- ◆ Mengelola seluruh sumber daya perangkat keras: cpu, memory, I/O
- ◆ Menyediakan sebuah set portabel, arsitektur dan API yang independen terhadap hardware untuk memungkinkan aplikasi user-space dan pustaka untuk menggunakan sumber daya perangkat keras.
- ◆ Menangani akses bersamaan dan penggunaan sumber daya perangkat keras dari aplikasi yang berbeda.
 - ◆ Contoh: sebuah antarmuka jaringan yang digunakan oleh beberapa aplikasi user-space melalui berbagai koneksi jaringan. Kernel bertanggung jawab untuk melakukan "multiplex" sumber daya perangkat keras.



Didalam kernel linux

Linux Kernel



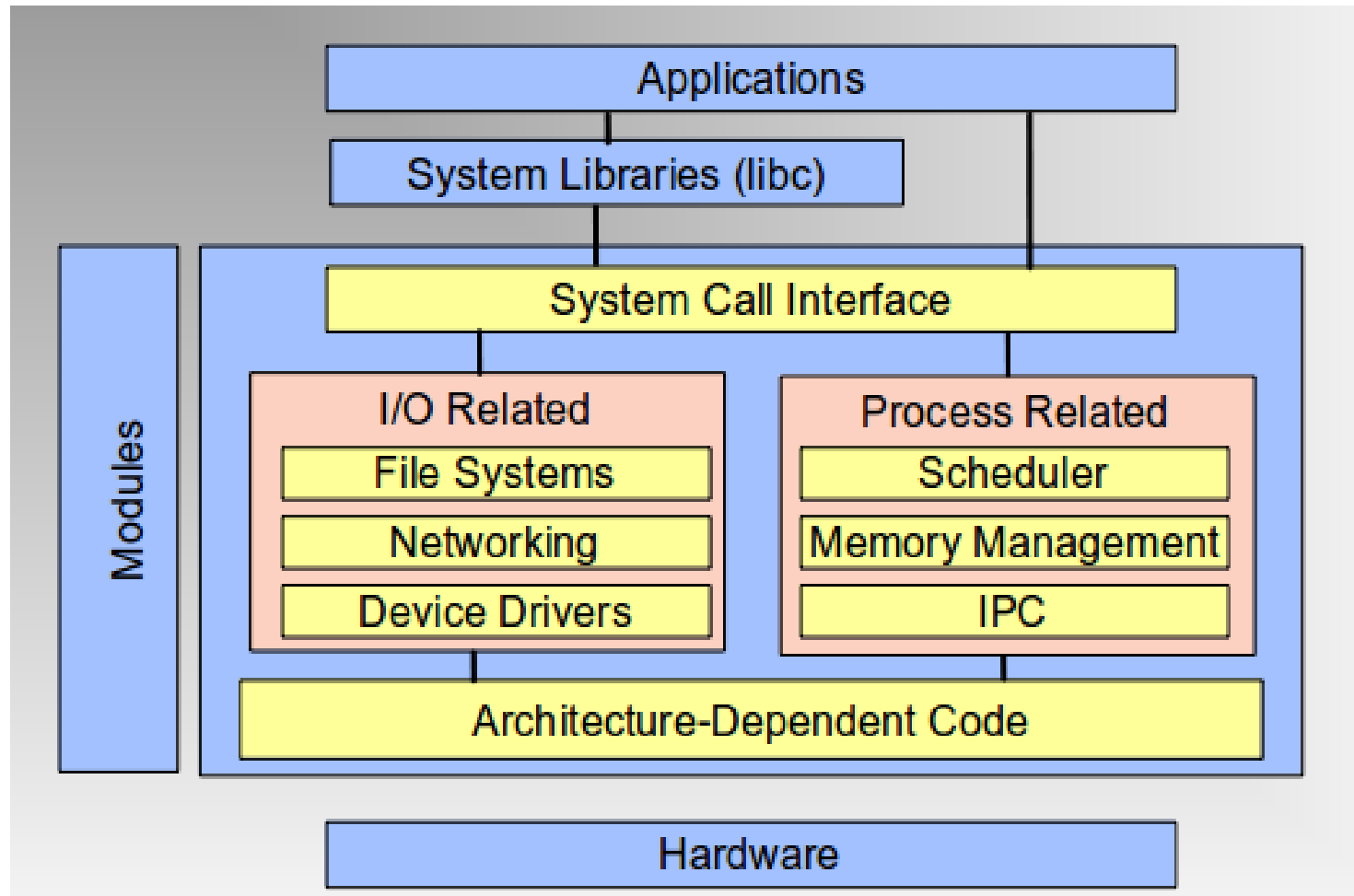
Implemented mainly in C,
a little bit of assembly.



Written in a Device Tree
specific language.



Arsitektur kernel linux



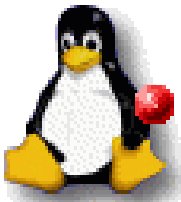
Lisensi linux

- Seluruh kode sumber kernel Linux adalah Free Software yang dirilis dibawah GNU General Public License version 2 (GPL v2).
- Lisensi ini pada dasarnya berarti bahwa:
 - Bila Anda menerima atau membeli perangkat dengan Linux di atasnya, Anda seharusnya menerima kode sumber Linux, dengan hak untuk mempelajari, memodifikasi dan mendistribusikannya.
 - Bila Anda memproduksi perangkat berbasis Linux, Anda harus melepaskan kode sumber kepada penerima, dengan hak yang sama, tanpa pembatasan



Fitur- fitur kunci kernel linux

- Portabilitas dan dukungan hardware. Berjalan pada kebanyakan arsitektur
- Skalabilitas. Dapat dijalankan pada komputer super serta pada perangkat kecil (4 MB RAM sudah cukup)
- Kepatuhan terhadap standar dan interoperabilitas.
- Dukungan jaringan lengkap
- Keamanan. Ini artinya tidak bisa menyembunyikan kesalahan yang ada. Kode ditinjau oleh banyak ahli.
- Stabilitas dan keandalan
- Modularitas. Bisa hanya mencakup apa yang sistem perlukan bahkan pada saat dijalankan.
- Mudah diprogram. Anda dapat belajar dari kode yang ada. Banyak sumber daya yang berguna di internet.



Arsitektur hardware yang didukung

- Lihat direktori **arch/** dalam kode sumber linux
- Minimum: 32 bit processors, dengan atau tanpa MMU, dan dukungan gcc
- 32 bit architectures (subdirektori arch/)
Contoh: arm, avr32, blackfin, m68k, microblaze, mips, score, sparc, um
- 64 bit architectures:
Contoh: alpha, arm64, ia64, sparc64, tile
- 32/64 bit architectures
Contoh: powerpc, x86, sh
- Temukan detilnya dalam kode sumber linux:
arch/<arch>/Kconfig, **arch/<arch>/README**, atau **Documentation/<arch>/**



Penomoran versi kernel linux



Mengidentifikasi versi kernel linux

- Untuk mengidentifikasi versi kernel linux dari sistem operasi linux yang Anda gunakan dapat dilakukan dengan cara mengetikkan perintah :

uname -r

- Atau dengan perintah:

cat /proc/version

- Dapat juga dengan Melihat isi direktori /boot, temukan file dengan nama vmlinuz-x.x.x.x



System calls

- ➔ Antarmuka utama antara kernel dan userspace adalah ***system call***
- ➔ Sekitar 300 system call yang menyediakan layanan-layanan utama kernel
 - ➔ Operasi File dan device, operasi jaringan, komunikasi antar proses (inter-process communication), manajemen proses, manajemen memori, memory mapping, timers, threads, synchronization, dll.
- ➔ Antarmuka system call ini dibungkus oleh pustaka C, dan aplikasi userspace biasanya tidak pernah membuat system call langsung melainkan menggunakan fungsi pustaka C yang sesuai



Virtual filesystems

- Linux membuat informasi sistem dan kernel tersedia dalam userspace melalui virtual filesystem
- Virtual filesystem memungkinkan aplikasi untuk melihat direktori dan file yang tidak ada pada setiap penyimpanan nyata: mereka diciptakan ***on the fly*** oleh kernel
- Dua virtual filesystem yang penting adalah:
 - **proc**, biasanya di mount ke /proc. Berisi informasi informasi berkaitan dg sistem operasi (seperti: parameter parameter manajemen memory, proses-proses, dll)
 - **sys**, biasanya di mount ke /sys. Merepresentasikan sistem sebagai sekumpulan device dan bus. Mengandung informasi tentang device dan bus.

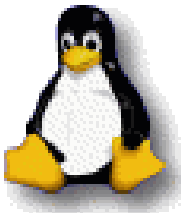
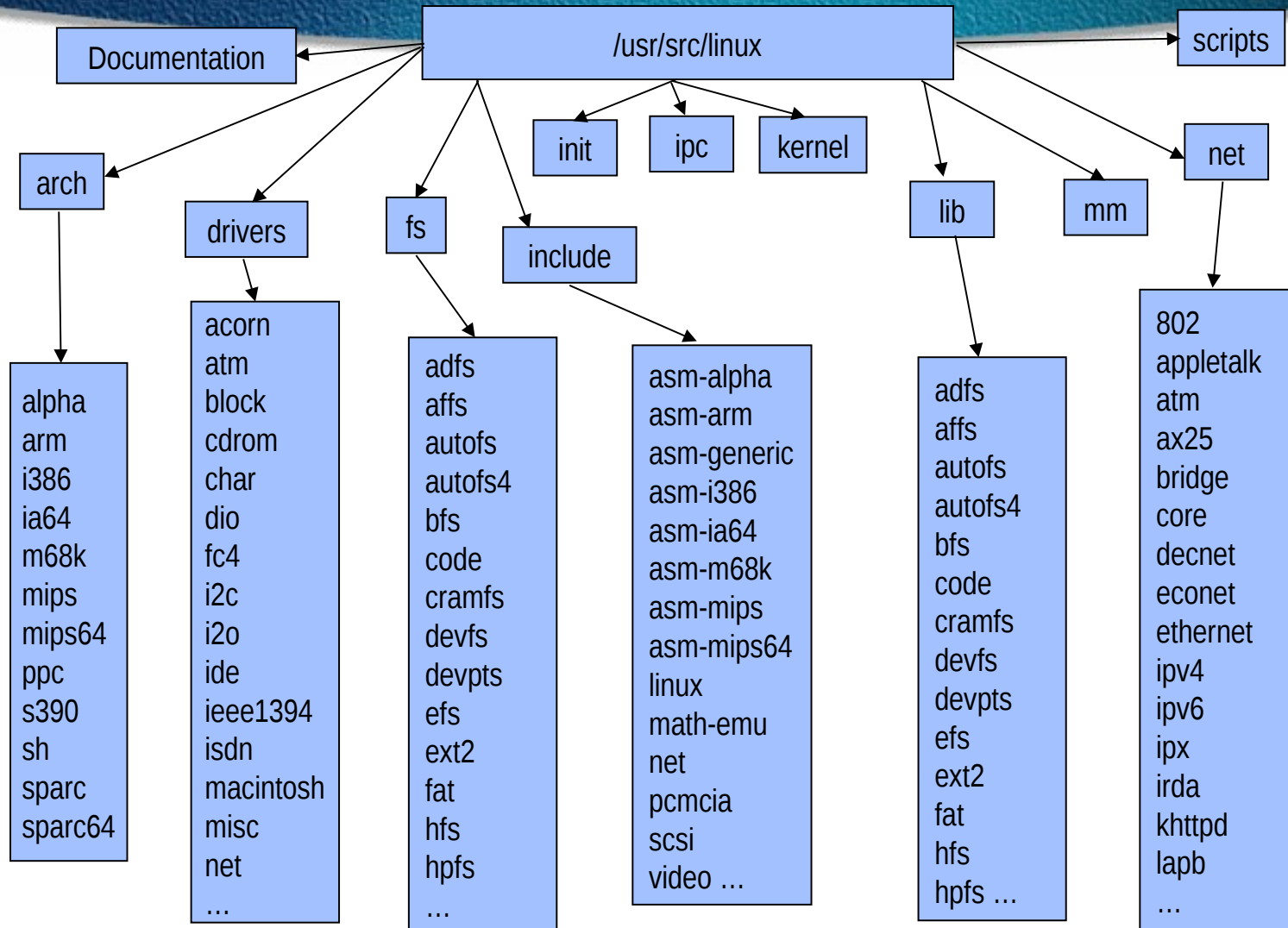


Kode sumber kernel

- Versi official kernel linux di rilis oleh Linus Torvalds tersedia di <http://www.kernel.org>
- Kode sumber kernel Linux versi 3.10:
 - Raw size: 573 MB (43,000 files, approx 15,800,000 lines)
 - gzip compressed tar archive: 105 MB
 - bzip2 compressed tar archive: 83 MB (better)
 - xz compressed tar archive: 69 MB (best)
- Mengapa ukuran kode sumber kernel linux begitu besar ?
 - Karena menyertakan banyak sekali driver driver perangkat keras (ribuan), mendukung banyak protokol jaringan, mendukung banyak arsitektur dan filesystem



Linux Source Tree Layout



Modul kernel

Meskipun "monolitik," dalam arti bahwa seluruh kernel berjalan dalam satu ruang alamat, kernel Linux adalah modular, mendukung penyisipan (insertion) dan penghapusan (removal) kode secara dinamis dari dirinya sendiri pada saat runtime. Subrutin terkait, data, entry dan exit point dikelompokkan bersama kedalam sebuah image biner tunggal, sebuah obyek kernel loadable, yang disebut modul.

Dukungan untuk modul-modul memungkinkan sistem untuk hanya memiliki image kernel dasar yang minimal, dengan fitur-fitur opsional dan driver-driver yang disertakan melalui loadable, sebagai obyek yang terpisah.

Modul-modul juga memungkinkan penghapusan dan pemuatan ulang kode kernel, memfasilitasi debugging, dan memungkinkan untuk permintaan pemuatan driver-driver baru dalam menangani hot plugging



Manfaat modul

- ★ Module mempermudah pengembangan driver-driver tanpa reboot: load, test, unload, rebuild, load...
- ★ Berguna untuk menjaga agar ukuran image kernel minimum.
- ★ Berguna mengurangi waktu boot: kita tidak perlu menghabiskan waktu untuk inisialisai device dan fitur fitur kernel yang hanya kita butuhkan di waktu akan datang



Dependensi modul

- ★ Beberapa modul kernel dapat bergantung pada modul modul lainnya, yang butuh dimuat (load) lebih dahulu.
- ★ Contoh: modul **usb-storage** bergantung pada modul modul **scsi_mod**, **libusual** dan **usbcore** .
- ★ Dependensi digambarkan dalam file **/lib/modules/<kernel-version>/modules.dep**

File ini dihasilkan ketika kita menjalankan perintah **make modules_install** setelah membangun (build) kernel.



Log kernel

- Ketika modul baru dimuat (load), informasi terkait tersedia dalam log kernel
 - Kernel menjaga pesan-pesanya dalam sebuah circular buffer (agar tidak banyak mengkonsumsi memori dengan banyaknya pesan)
 - Pesan pesan log kernel tersedia melalui perintah **dmesg** (**diagnostic message**)
 - Pesan pesan log kernel juga ditampilkan ke sistem konsol
 - Perhatikan, kita juga dapat menulis pesan ke log kernel melalui user-space:

```
# echo "Debug info" > /dev/kmsg
```



Utilitas Modul

- Melihat modul modul yang sedang dimuat:

lsmod

- Mendapatkan informasi tentang modul:

modinfo <module_name> atau ***modinfo <module_path>.ko***

- Memuat modul :

insmod <module_path>.ko

- Melepas modul:

rmmod <module_name>

- Memuat dan melepas modul dengan modprobe

modprobe <modul_name> atau ***modprobe -r <modul_name>***

