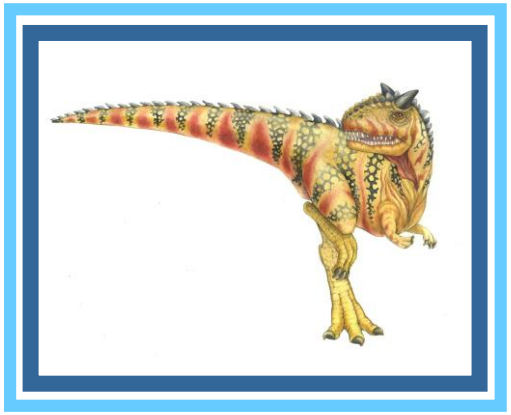
Chapter 2: System Structures



Silberschatz, Galvin and Gagne ©2013

لمحات عن النظام :2الفصل

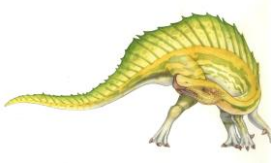
- خدمات النظام التشغيل
- المستخدم واجهة نظام التشغيل
- يدعو النظام
- أنواع المكالمات النظام
- برامج النظام
- تعمل تصميم النظام وتنفيذ
- هيكل نظام التشغيل
- نظام التشغيل التصحيح
- الجيل نظام التشغيل
- التمهيد نظام

أهداف

- لوصف الخدمات التي يقدمها نظام التشغيل للمستخدمين، والعمليات، وغيرها من النظم
- لمناقشة الطرق المختلفة لهيكلة نظام التشغيل
- لشرح كيفية تثبيت أنظمة التشغيل وتخصيص وكيف التمهيد

خدمات النظام التشغيل

- توفر أنظمة التشغيل بيئة لتنفيذ البرامج والخدمات لبرامج والمستخدمين
- مجموعة واحدة من خدمات نظام تشغيل يوفر وظائف التي هي مفيدة للمستخدم:
  - **واجهة المستخدم** - تقريبا كل أنظمة التشغيل لديها واجهة المستخدم (واجهة المستخدم).
    - 4 يتراوح بين سطر الأوامر (CLI) ، الرسومات واجهة المستخدم ، دفعة (واجهة المستخدم الرسومية)
  - **تنفيذ البرنامج** - يجب أن يكون النظام قادرا على تحميل البرنامج في الذاكرة ولتشغيل أن تنفيذ البرنامج، (تشير إلى خطأ) نهاية، إما بشكل طبيعي أو غير طبيعي
  - **عمليات الإدخال / الإخراج** - برنامج تشغيل قد تتطلب I / O، والتي قد تنطوي على ملف أو جهاز I / O
  - **نظام الملفات التلاعب** - البرامج تحتاج لقراءة وكتابة الملفات والدلائل، إن نظام الملفات ذات أهمية خاصة. وإنشاء وحذفها، تفتيشها، معلومات ملف القائمة، وإدارة إذن.



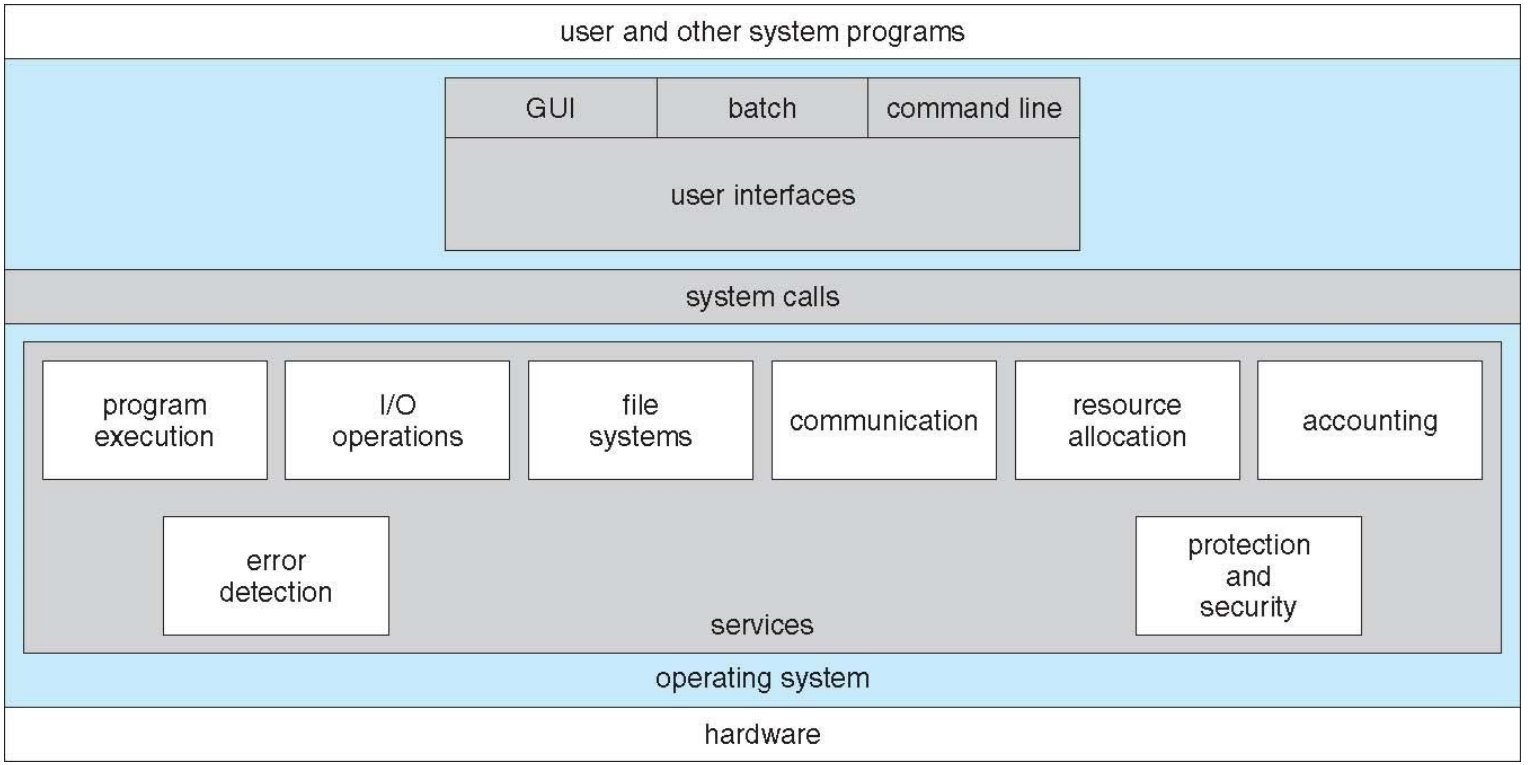
(يتبع)خدمات نظام التشغيل

- **مجال الاتصالات** - العمليات قد تبادل المعلومات، على نفس جهاز الكمبيوتر أو بين أجهزة الكمبيوتر عبر شبكة اتصال
  - الحزم انتقلت من)قد تكون الاتصالات عبر الذاكرة المشتركة أو من خلال رسالة عابرة نظام التشغيل(
- **اكتشاف الخطأ** - نظام التشغيل يجب أن يكون على علم باستمرار من الأخطاء المحتملة
  - الإخراج، في /قد تحدث في وحدة المعالجة المركزية والذاكرة الأجهزة، في أجهزة الإدخال برنامج المستخدم
  - لكل نوع من الخطأ، يجب أن نظام التشغيل تتخذ الإجراءات المناسبة لضمان الحوسبة الصحيحة ومتسقة
  - يمكن مرافق التصحيح يعزز إلى حد كبير قدرات المستخدم ومبرمج الاستعمال الفعال للنظام

(يتبع)خدمات نظام التشغيل

- مجموعة أخرى من وظائف نظام التشغيل موجود لضمان كفاءة تشغيل النظام نفسه عن طريق تقاسم الموارد
  - **-تخصيص الموارد** عندما عدة مستخدمين أو وظائف متعددة تعمل في وقت واحد، يجب تخصيص الموارد لكل منهم
    - العديد من أنواع الموارد -بعض)مثل دورات وحدة المعالجة المركزية والذاكرة الرئيسية، وتخزين قد طلب عام والافراج (I / O أجهزة)قد يكون رمز تخصيص خاص، والبعض الآخر (الملفات عن كود
  - **-محاسبة** لتتبع أي من المستخدمين استخدام كم وأنواع من موارد الكمبيوتر
  - **-الحماية والأمن** أصحاب المعلومات المخزنة في مهارات متعددة المستخدمين أو الشبكات نظام الكمبيوتر قد يرغبون في السيطرة على استخدام تلك المعلومات، يجب أن العمليات المتزامنة لا تتداخل مع بعضها البعض
    - **حماية** ينطوي على ضمان أن جميع الوصول إلى موارد النظام يتم التحكم
    - **أمن** I / O من النظام من الخارج يتطلب مصادقة المستخدم، ويمتد للدفاع عن أجهزة الخارجية من محاولات الوصول غير صالحة
    - سلسلة لا تكون .إذا كان النظام هو أن تكون محمية وآمنة، يجب أن توضع الاحتياطات طوال ذلك قوية إلا بقدر قوة أضعف حلقاتها.

وجهة نظر من خدمات نظام التشغيل



CLI -المستخدم واجهة نظام التشغيل

- CLI أو **مترجم الأوامر** يسمح بدخول القيادة المباشرة
  - تنفذ أحيانا في النواة، وأحيانا من قبل برنامج النظم
  - **قذائف** -أحيانا نكهات متعددة تنفيذها
  - جلب في المقام الأول أمر من المستخدم وتنفيذ عليه
    - أوامر أحيانا في البناء، في بعض الأحيان مجرد أسماء البرامج
      - إذا كان هذا الأخير، إضافة ميزات جديدة لا تتطلب قذيفة تعديل

بورن شل قيادة مترجم

Default

New Info Close Execute Bookmarks

Default Default

```
PBG-Mac-Pro:~ pbg$ w
15:24  up 56 mins, 2 users, load averages: 1.51 1.53 1.65
USER     TTY      FROM              LOGIN@  IDLE WHAT
pbg      console  -                 14:34      50 -
pbg      s000     -                 15:05       - w
PBG-Mac-Pro:~ pbg$ iostat 5
          disk0           disk1          disk10        cpu     load average
    KB/t tps  MB/s     KB/t tps  MB/s     KB/t tps  MB/s  us sy id   1m   5m   15m
   33.75 343 11.30    64.31  14  0.88    39.67   0  0.02  11  5 84  1.51 1.53 1.65
    5.27 320  1.65     0.00   0  0.00     0.00   0  0.00   4  2 94  1.39 1.51 1.65
    4.28 329  1.37     0.00   0  0.00     0.00   0  0.00   5  3 92  1.44 1.51 1.65
^C
PBG-Mac-Pro:~ pbg$ ls
Applications                    Music                           WebEx
Applications (Parallels)        Pando Packages                  config.log
Desktop                         Pictures                        getsmartdata.txt
Documents                       Public                          imp
Downloads                       Sites                           log
Dropbox                         Thumbs.db                       panda-dist
Library                         Virtual Machines                prob.txt
Movies                          Volumes                         scripts
PBG-Mac-Pro:~ pbg$ pwd
/Users/pbg
PBG-Mac-Pro:~ pbg$ ping 192.168.1.1
PING 192.168.1.1 (192.168.1.1): 56 data bytes
64 bytes from 192.168.1.1: icmp_seq=0 ttl=64 time=2.257 ms
64 bytes from 192.168.1.1: icmp_seq=1 ttl=64 time=1.262 ms
^C
--- 192.168.1.1 ping statistics ---
2 packets transmitted, 2 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 1.262/1.760/2.257/0.498 ms
PBG-Mac-Pro:~ pbg$
```

واجهة المستخدم الرسومية -المستخدم واجهة نظام التشغيل

- سهل الاستخدام واجهة المجاز **سطح المكتب**
  - عادة الماوس ولوحة المفاتيح، ورصد
  - **الرموز** تمثل الملفات والبرامج والإجراءات وغيرها
  - توفير المعلومات والخيارات)أزرار الماوس مختلفة على الكائنات في قضية اجهة الإجراءات المختلفة (المعروف باسم **مجلد**)وتنفيذ وظيفة، والدليل المفتوح
  - اخترع في زيروكس بارك

- واجهات واجهة المستخدم الرسومية CLIوتشمل العديد من الأنظمة الآن على حد سواء
  - "القيادة"قذيفة CLIمايكروسوفت ويندوز هو واجهة المستخدم الرسومية مع
  - واجهة المستخدم الرسومية مع يونيكس نواة تحت وقذائف المتاحة "أكوا"هو OS Xأبل ماك
  - (الرفيق، كيدي، جنوم)لينوكس ويونكس مع واجهات واجهة المستخدم الرسومية الاختيارية CLIوقد

الشاشة التي تعمل باللمس واجهات

- تتطلب أجهزة تعمل باللمس واجهات جديدة
  - الماوس غير ممكن أو غير مرغوب
  - الإجراءات واختيار على أساس الإيماءات
  - لوحة المفاتيح الافتراضية لإدخال النص

واجهة المستخدم الرسومية نظام التشغيلMac OS X

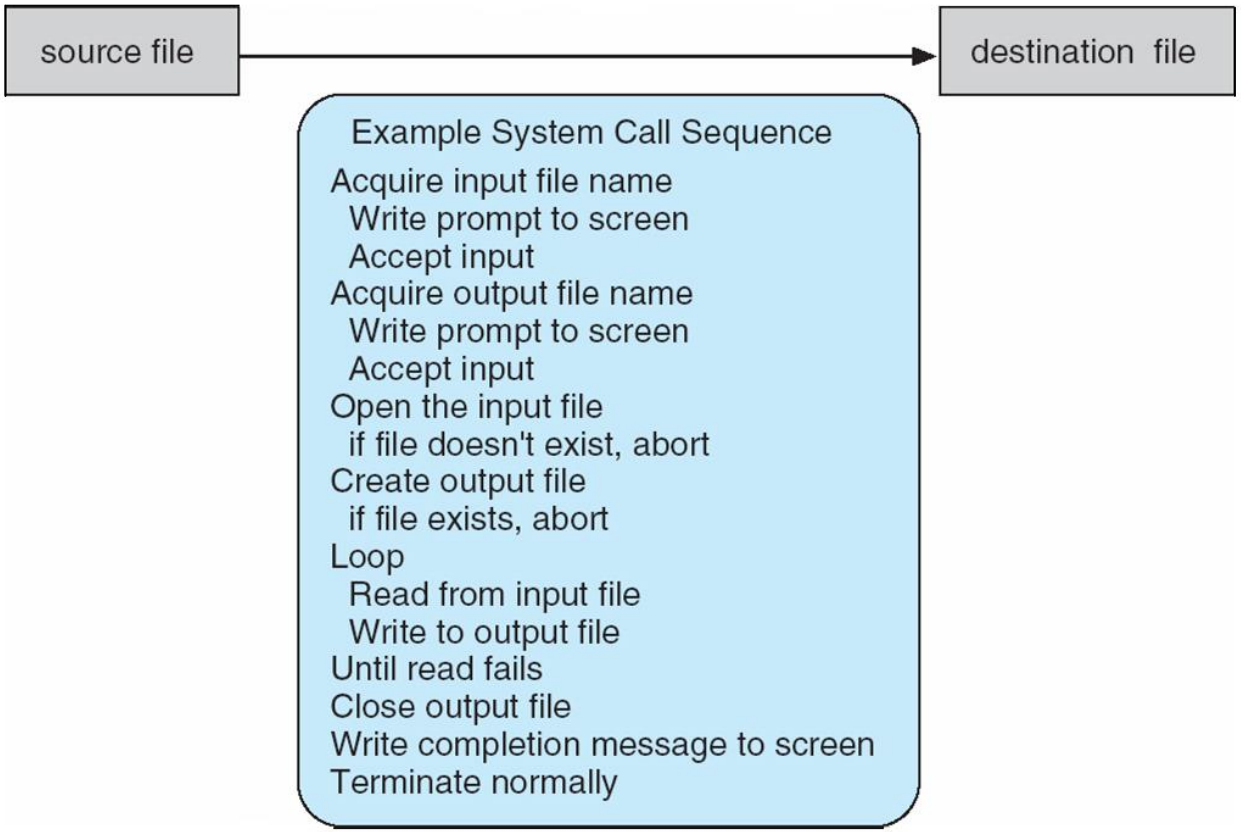
يدعو النظام

- واجهة برمجة للخدمات المقدمة من قبل نظام التشغيل
- كتب عادة في لغة عالية المستوى(C أو C ++)
- بدلا من **(API)** **واجهة برمجة التطبيقات** الوصول معظمهم من البرامج عبر رفيع المستوى استخدام استدعاء نظام مباشر
- ثلاث واجهات برمجة التطبيقات الأكثر شيوعا هي API Win32 لويندوز، والمعهد POSIX بما في ذلك جميع الإصدارات تقريبا من يونيكس، لينكس، و (POSIX للأنظمة المستندة إلى Mac OS X)، و API جافا الافتراضية (JVM جافا لآلة)
- لماذا استخدام واجهات برمجة التطبيقات بدلا من استدعاءات النظام؟

(لاحظ أن الأسماء استدعاء نظام المستخدمة في هذا النص هي عامة)

مثال على المكالمات النظام

- استدعاء نظام تسلسل لنسخ محتويات ملف واحد إلى ملف آخر



مثال API قياسي

EXAMPLE OF STANDARD API

As an example of a standard API, consider the `read()` function that is available in UNIX and Linux systems. The API for this function is obtained from the `man` page by invoking the command

```
man read
```

on the command line. A description of this API appears below:

```
#include <unistd.h>

ssize_t     read(int fd, void *buf, size_t count)
```

return value — function name — parameters

A program that uses the `read()` function must include the `unistd.h` header file, as this file defines the `ssize_t` and `size_t` data types (among other things). The parameters passed to `read()` are as follows:

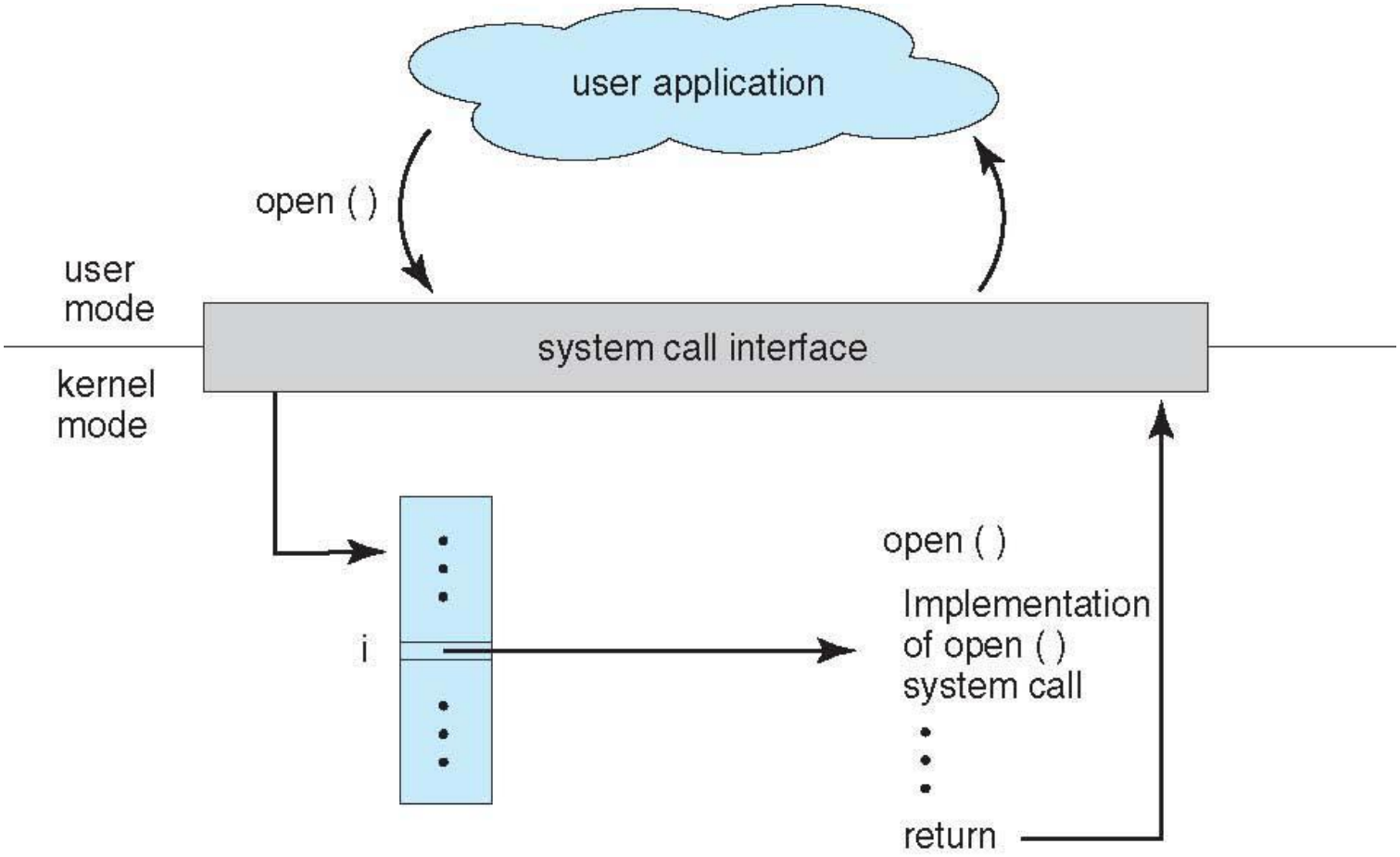
- `int fd`—the file descriptor to be read
- `void *buf`—a buffer where the data will be read into
- `size_t count`—the maximum number of bytes to be read into the buffer

On a successful read, the number of bytes read is returned. A return value of 0 indicates end of file. If an error occurs, `read()` returns −1.

نداء نظام التنفيذ

- عادة، عددا ترتبط مع كل مكالمة نظام
  - **واجهة استدعاء نظام** يحافظ على طاولة فهرستها وفقا لهذه الأرقام
- واجهة استدعاء نظام تحتج استدعاء نظام المقصودة في نواة نظام التشغيل وإرجاع حالة استدعاء نظام وأي قيم الإرجاع
- المتصل حاجة لا يعرفون شيئا عن كيفية تنفيذ استدعاء النظام
  - القيام بمثابة دعوة نتيجة OSوفهم ما سوف APIفقط يحتاج إلى طاعة
  - APIمعظم التفاصيل من واجهة نظام التشغيل مخفية عن مبرمج من قبل
    - مجموعة من الوظائف المدمجة في المكتبات)تدار من قبل مكتبة دعم وقت التشغيل (المضمنة مع مترجم

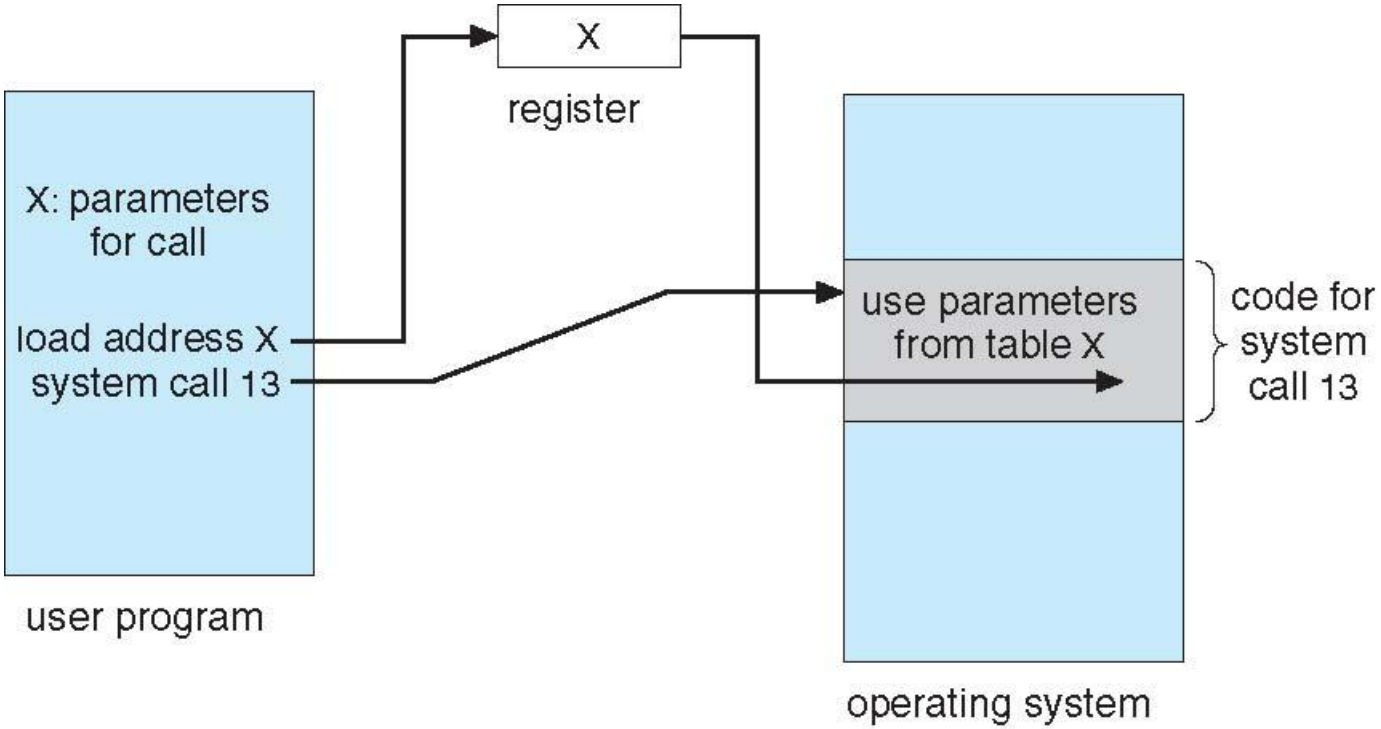
العلاقة API OS - -استدعاء النظام



نظام نداء معلمة فاة

- في كثير من الأحيان، مطلوب مزيد من المعلومات من مجرد هوية استدعاء النظام المطلوب
  - نوع الدقيق وكمية المعلومات تختلف وفقا لنظام التشغيل والدعوة
- ثلاث طرق العامة المستخدمة لتمرير المعلمات إلى نظام التشغيل
  - تمرير المعاملات في السجلات :أبسط
    - في بعض الحالات، قد يكون أكثر من المعلمات من السجلات
  - المعلمات تخزينها في كتلة، أو الجدول، في الذاكرة، وعنوان كتلة مرت كمعلمة في سجل
    - هذا النهج لينكس، وسولاريس اتخذت
  - المعلمات وضعت، أو **دفع**، على **كومة** من قبل البرنامج و **برزت** بعيدا عن كومة من قبل نظام التشغيل
  - كتلة وكومة أساليب لا تحد من عدد أو طول المعلمات يتم تمرير

المعلمة تمرير عبر الجدول



أنواع المكالمات النظام

- تحكم في عملية
  - نهاية، إجهاض
  - الحمل، وتنفيذ
  - إنشاء العملية، عملية إنهاء
  - الحصول على سمات عملية، تعيين سمات عملية
  - الانتظار لوقت
  - انتظر الحدث، الحدث إشارة
  - تخصيص وتحرير الذاكرة

  - تفريغ الذاكرة إذا كان الخطأ
  - **المصحح** لتحديد **البق، خطوة واحدة** تنفيذ
  - **أقفال** لإدارة الوصول إلى البيانات المشتركة بين العمليات

أنواع المكالمات النظام

- ادارة الملفات
  - إنشاء ملف، ملف حذف
  - فتح، إغلاق الملف
  - القراءة والكتابة وإعادة
  - الحصول على وتعيين سمات الملف
- ادارة الجهاز
  - طلب الجهاز، جهاز الإفراج
  - القراءة والكتابة وإعادة
  - الحصول على سمات الجهاز، تعيين سمات الجهاز
  - منطقيا إرفاق أو فصل الأجهزة

(يتبع)أنواع المكالمات نظام

- صيانة المعلومات
  - الحصول على الوقت أو التاريخ، ضبط الوقت أو التاريخ
  - الحصول على بيانات نظام، مجموعة البيانات النظام
  - الحصول على والعملية التي، سمات الملف، أو جهاز

- مجال الاتصالات
  - خلق، وحذف اتصال الاتصالات
  - إرسال واستقبال الرسائل إذا **نموذج تمرير رسالة** إلى **اسم المضيف** أو **اسم العملية**
    من عند **زبون** إلى **الخادم** 4
  - **نموذج الذاكرة المشتركة** إنشاء والوصول إلى مناطق الذاكرة
  - مركز المعلومات نقل
  - إرفاق و فصل الأجهزة عن بعد

(يتبع)أنواع المكالمات نظام

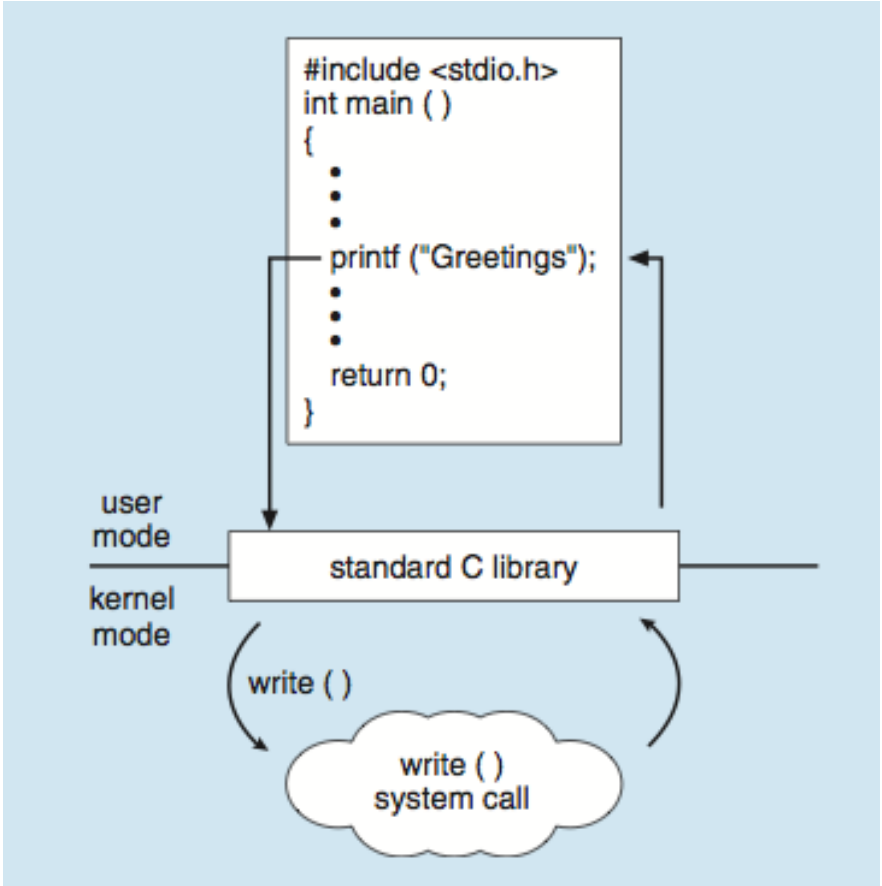
- حماية
  - التحكم في الوصول إلى الموارد
  - الحصول على وتعيين الأذونات
  - سماح ومنع وصول المستخدم

أمثلة من ويندوز و المكالمات نظام يونكس

	Windows	Unix
Process Control	CreateProcess() ExitProcess() WaitForSingleObject()	fork() exit() wait()
File Manipulation	CreateFile() ReadFile() WriteFile() CloseHandle()	open() read() write() close()
Device Manipulation	SetConsoleMode() ReadConsole() WriteConsole()	ioctl() read() write()
Information Maintenance	GetCurrentProcessID() SetTimer() Sleep()	getpid() alarm() sleep()
Communication	CreatePipe() CreateFileMapping() MapViewOfFile()	pipe() shmget() mmap()
Protection	SetFileSecurity() InitlializeSecurityDescriptor() SetSecurityDescriptorGroup()	chmod() umask() chown()

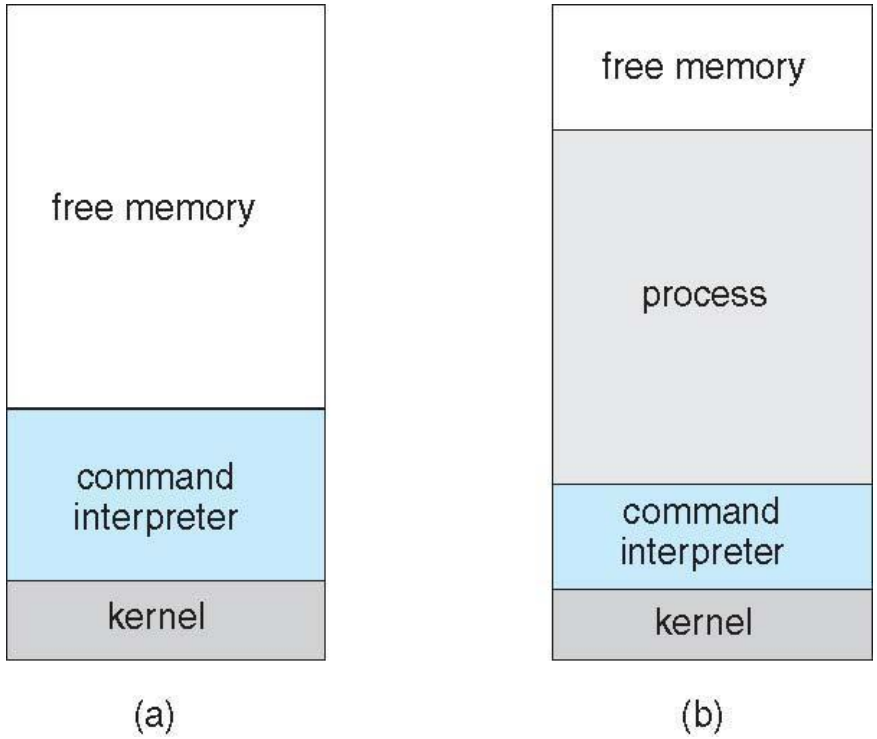
مكتبة مثال Cمستوى

- C برنامج استحضار مكتبة الدعوة printf() الذي يدعو الكتابة، (استدعاء نظام)



على سبيل المثال: MS-DOS

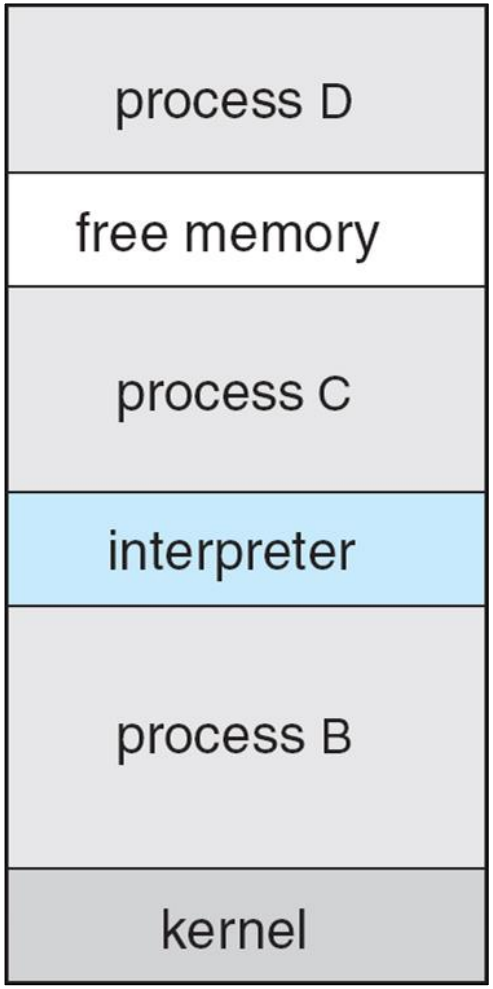
- واحد المهام
- الاحتجاج قذيفة عند تمهيد النظام
- طريقة بسيطة لتشغيل البرنامج
  - أي عملية خلق
- مساحة الذاكرة واحد
- برنامج يحمل في الذاكرة، الكتابة فوق جميع ولكن النواة
- قذيفة إعادة تحميل <-خروج البرنامج



تشغيل برنامج (ب)عند بدء تشغيل النظام (أ)

فري :على سبيل المثال

- يونيكس البديل
- تعدد المهام
- استدعاء اختيار المستخدم من <-تسجيل دخول المستخدم قذيفة
- نظام لخلق عملية)دعوة (قذيفة ينفذ شوكة
  - لتحميل البرنامج في عملية ()تعدم إكسيك
  - قذيفة ينتظر عملية لإنهاء أو تواصل مع أوامر المستخدم
- رمز - 0 <أي خطأ أو - 0إنهاء العملية مع رمز من الخطأ



برامج النظام

- ويمكن تقسيمها إلى .توفر برامج نظام بيئة ملائمة لتطوير البرامج وتنفيذها:
  - ملف التلاعب
  - مركز المعلومات المخزنة في بعض الأحيان في تعديل الملف
  - دعم لغة البرمجة
  - تحميل البرامج وتنفيذها
  - مجال الاتصالات
  - خدمات الخلفية
  - برامج التطبيقات

- ويعرف رأي معظم المستخدمين من نظام تشغيل برامج نظام، وليس استدعاءات النظام الفعلية

برامج النظام

- توفير بيئة ملائمة لتطوير البرامج وتنفيذها
  - بعضها مجرد واجهات المستخدم لدعوات النظام؛ البعض الآخر أكثر من ذلك بكثير مجمع
- إنشاء وحذف ونسخ وإعادة تسمية، والطباعة، وتفريغ القائمة، وعموما التعامل مع **-ادارة الملفات** الملفات والدلائل
- **مركز المعلومات**
  - التاريخ والوقت ومقدار الذاكرة المتاحة، مساحة **-**البعض يطلب النظام لمزيد من المعلومات القرص، وعدد من المستخدمين
  - تقدم الآخرين الأداء التفصيلية، وقطع الأشجار، وتصحيح المعلومات
  - عادة، هذه البرامج تنسيق وطباعة الإخراج إلى محطة أو أجهزة الإخراج الأخرى
  - تستخدم لتخزين واسترجاع المعلومات التكوين **-**بعض النظم تنفيذ **سجل**

(يتبع)برامج النظام

- **ملف تعديل**
  - برامج تحرير النصوص لإنشاء وتعديل ملفات
  - الأوامر الخاصة للبحث محتويات الملفات أو تنفيذ تحولات النص
- **دعم البرمجة اللغوية** - المجمعين، المجمعات، مصححات والمترجمين الفوريين تقدم أحيانا
- **تحميل البرامج وتنفيذها** - لوادر المطلقة، رافعات المنقولة المحررين الربط، وتراكب رافعات، وأنظمة التصحيح لأعلى مستوى ولغة الآلة
- **مجال الاتصالات** - توفير آلية لإنشاء اتصالات افتراضية بين العمليات والمستخدمين، وأنظمة الكمبيوتر
  - السماح للمستخدمين بإرسال رسائل إلى واحدة شاشات آخر، تصفح صفحات الويب، وإرسال رسائل البريد الإلكتروني، الدخول عن بعد، ونقل الملفات من جهاز واحد إلى آخر

(يتبع)برامج النظام

- **الخدمات الخلفية**
  - إطلاق في وقت التمهيد
    - بعض لبدء تشغيل النظام، ثم إنهاء
    - بعض من تمهيد النظام لإيقاف
  - تقديم التسهيلات مثل فحص القرص، وجدولة عملية، وقطع الأشجار الخطأ، والطباعة
  - تشغيل في سياق المستخدم ليس نواة السياق
  - المعروفة باسم **خدمات**، **فرعية**، **الشياطين**
- **برامج التطبيقات**
  - لا تتعلق النظام
  - تشغيل من قبل المستخدمين
  - لا تعتبر عادة جزءا من نظام التشغيل
  - التي أطلقها سطر الأوامر، انقر فوق الماوس، إصبع كزة

تصميم نظام التشغيل وتنفيذ

- ولكن أثبتت بعض النهج الناجح"قابلة للحل"تصميم وتنفيذ نظام التشغيل يست ،
- البنية الداخلية للنظام التشغيل المختلفة يمكن أن تختلف على نطاق واسع
- تبدأ من خلال الأهداف والمواصفات التي تحدد
- تتأثر اختيار الأجهزة، ونوع النظام
- مستخدم أهداف و نظام أهداف
  - نظام التشغيل يجب أن تكون ملائمة للاستخدام، سهلة التعلم، موثوق بها -أهداف العضو وآمنة، وسريع
  - أن نظام التشغيل يكون من السهل لتصميم وتنفيذ وصيانة، فضلا عن المرونة، -أهداف نظام يمكن الاعتماد عليها، خالية من الأخطاء، وكفاءة

تعمل تصميم النظام و (يتبع)تنفيذ

- مبدأ هام لفصل
  - **سياسة**: ***ماذا*** سينجز؟
  - **آلية**: ***كيف*** للقيام بذلك؟
- آليات تحدد كيفية القيام بشيء ما، سياسات تقرر ما سيتم القيام به
  - فصل السياسة عن آلية هو مبدأ مهم جدا، لأنها تتيح أقصى قدر من المرونة إذا القرارات السياسية هي أن تتغير في وقت لاحق
- تحديد وتصميم نظام التشغيل هو مهمة درجة عالية من الإبداع ل **هندسة البرمجيات**

التنفيذ

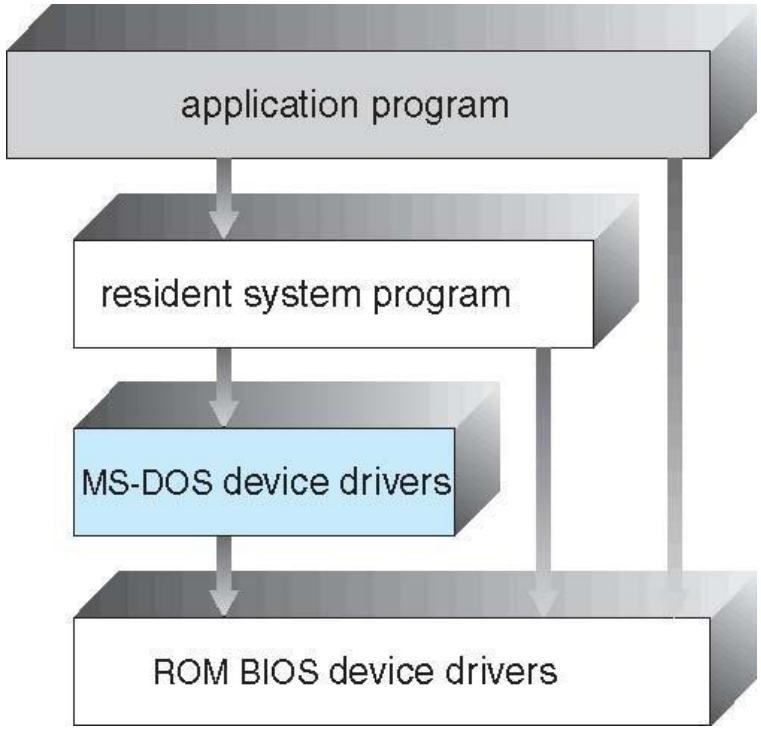
- تباين كبير
  - أنظمة تشغيل المبكرة في لغة التجميع
  - ثم لغات البرمجة نظام مثل رأس الغول، PL / 1
  - الآن C ، C ++
- في الواقع عادة مزيج من اللغات
  - أدنى مستوى في التجمع
  - الهيئة الرئيسية في C
  - ، لغات البرمجة مثل بيرل، بيثون، ومخطوطات قذيفة++ C ، C برامج نظم في
- عن مستوى عال لغة أسهل لل **ميناء** إلى الأجهزة الأخرى
  - ولكن بوتيرة أبطأ
- **محاكاة** يمكن أن تسمح لنظام التشغيل ليعمل على الأجهزة غير الناطقين بها

هيكل نظام التشغيل

- للأغراض العامة نظام التشغيل هو برنامج كبير جدا
- طرق مختلفة لمبنى واحد على النحو التالي

هيكل بسيط

- MS-DOS - أي مكتوبة لتوفير معظم الوظائف في أقل مساحة
  - لا تقسيمها إلى وحدات
  - MS-DOS لديه على الرغم من أن بعض هيكل، واجهات، ومستويات الوظائف ليست منفصلة بشكل جيد

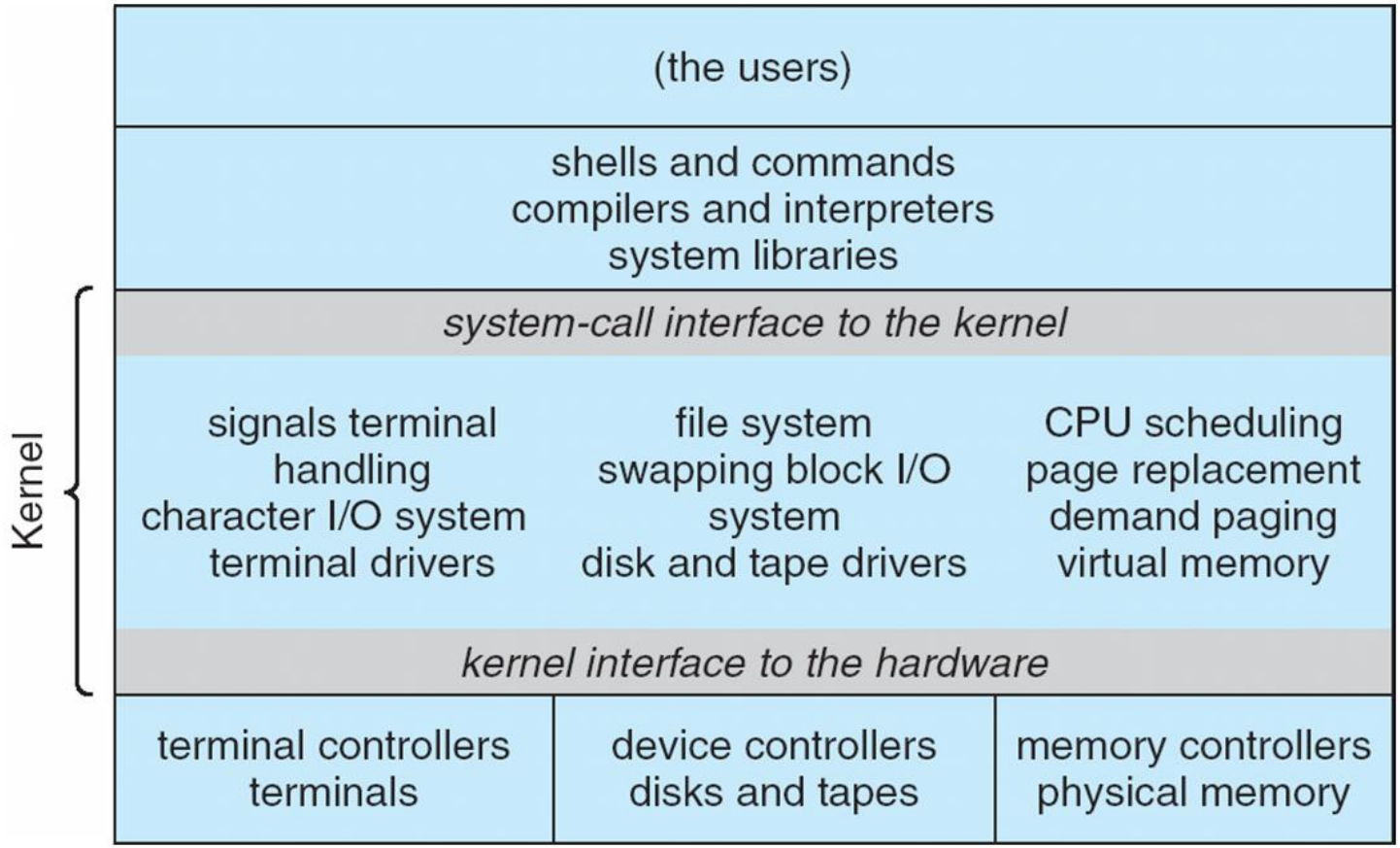


يونيكس

- يتكون .، وكان نظام التشغيل يونيكس الأصلي محدودة وظائف الأجهزة هيكلة محدودة-يونيكس نظام التشغيل يونيكس من جزئين للانفصال
  - برامج أنظمة
  - نواة
    - يتكون من كل شيء تحت واجهة استدعاء نظام وفوق الأجهزة الفعلي
    - يوفر نظام الملفات، وجدولة وحدة المعالجة المركزية، إدارة الذاكرة، وغيرها من وظائف التشغيل النظام؛ عدد كبير من الوظائف لمستوى واحد

هيكل نظام يونيكس التقليدية

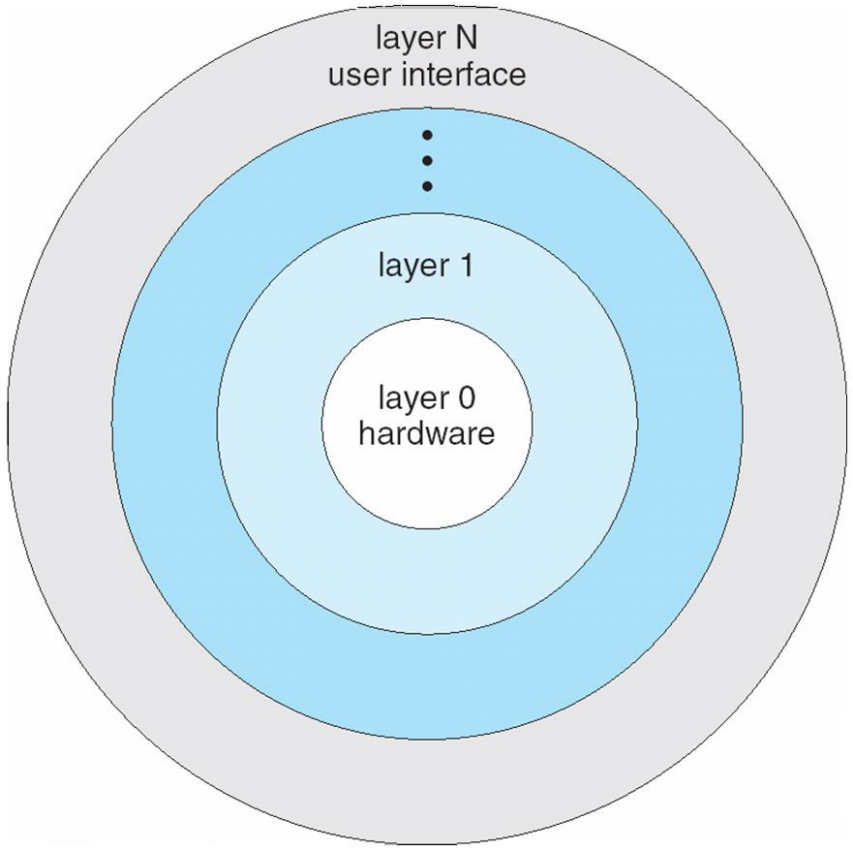
ما وراء بسيطة ولكن ليس الطبقات تماما



نهج الطبقات

- ينقسم نظام التشغيل إلى عدد من الطبقات(المستويات) التي بنيت ، على أعلى الطبقات السفلى لكل منهما. الطبقة السفلى (طبقة0)، والأجهزة. أعلى (طبقةN) هي واجهة المستخدم.

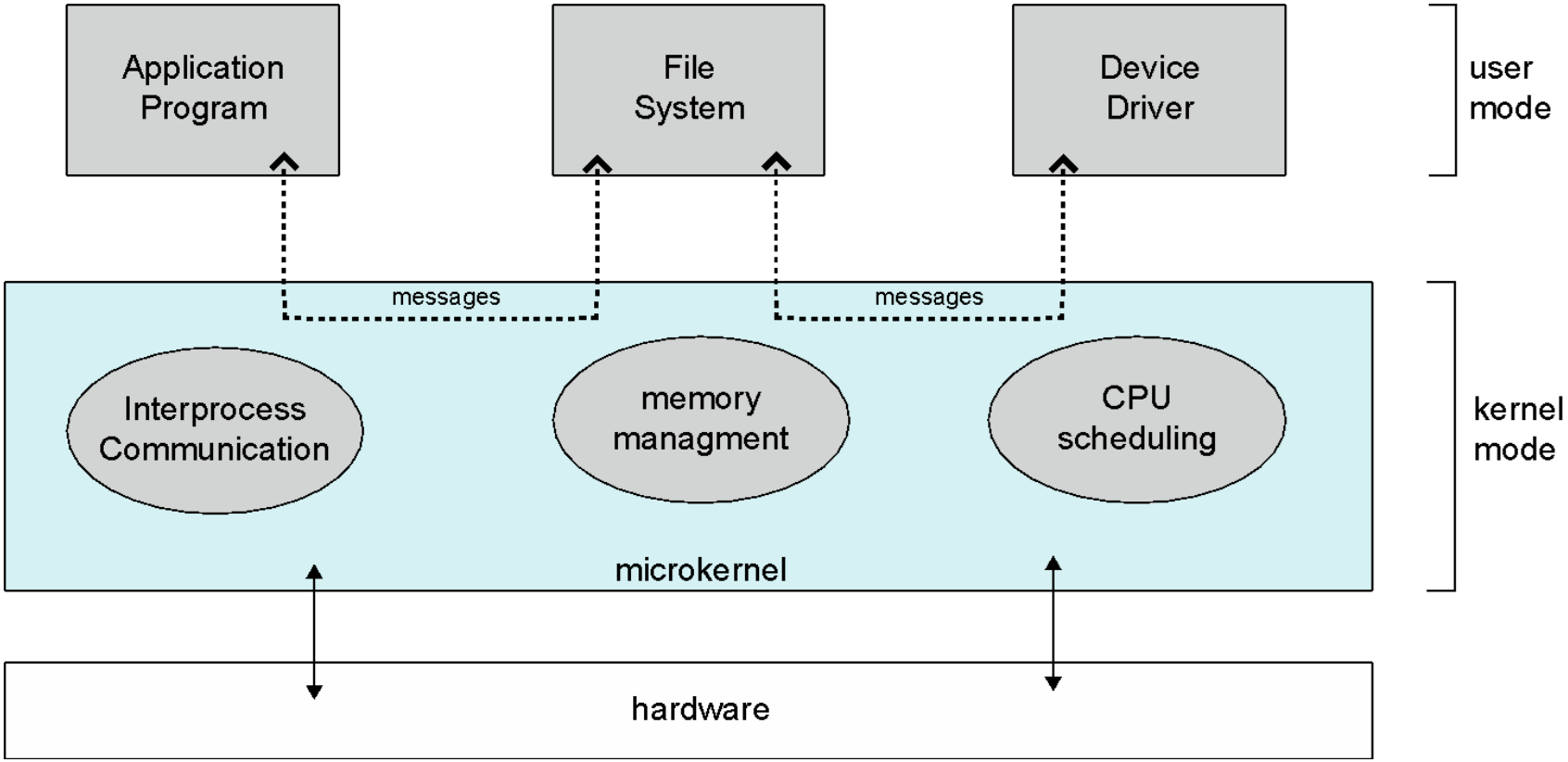
- مع نمطية، ويتم اختيار طبقات بحيث يستخدم كل وظائف (العمليات) والخدمات فقط طبقات المستوى الأدنى



هيكل النظام النوية

- يتحرك بقدر من النواة إلى الفضاء المستخدم
- **ماخ** مثال على **النوية**
  - يستند جزئيا إلى ماخ (**داروين**)نواة Mac OS X نظام التشغيل
- يتم الإتصال بينهم بين وحدات للمستخدم باستخدام **يمر رسالة**
- فوائد:
  - أسهل لتوسيع النوية
  - أسهل إلى ميناء نظام التشغيل لأبنية جديدة
  - (يتم تشغيل أقل التعليمات البرمجية في وضع النواة)أكثر موثوقية
  - أكثر أمنا
- مضارها:
  - النفقات العامة أداء الفضاء المستخدم إلى الاتصالات الفضائية النواة

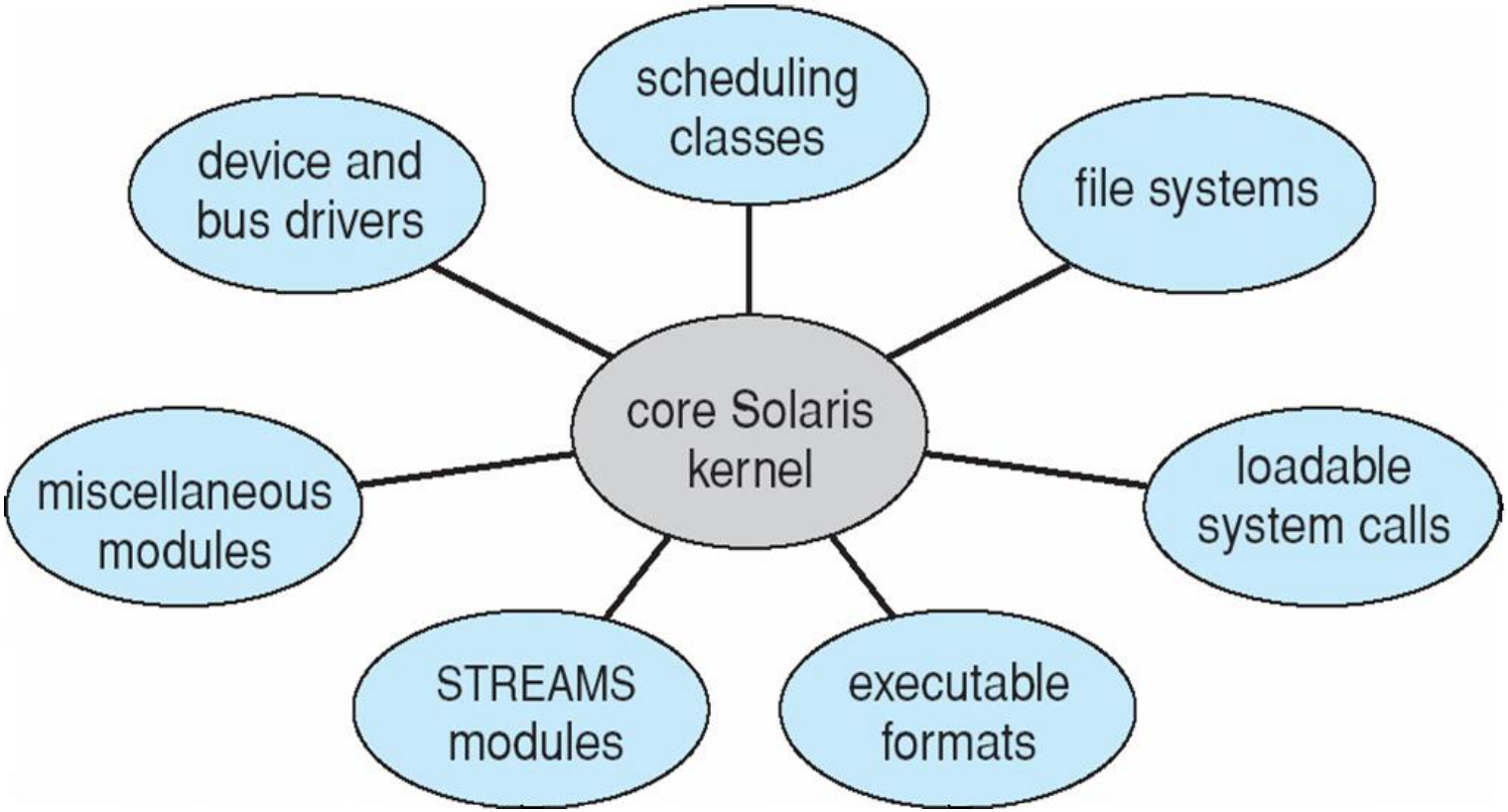
هيكل النظام النوية



وحدات

- وحدات النواة loadable معظم أنظمة التشغيل الحديثة تنفيذ
  - يستخدم نهج وجوه المنحى
  - كل مكون أساسي منفصل
  - كل المحادثات مع الآخرين عبر واجهات معروفة
  - حسب الحاجة داخل النواة loadable كل واحدة
- عموما، على غرار طبقات ولكن مع أكثر مرونة
  - لينكس، وسولاريس، الخ

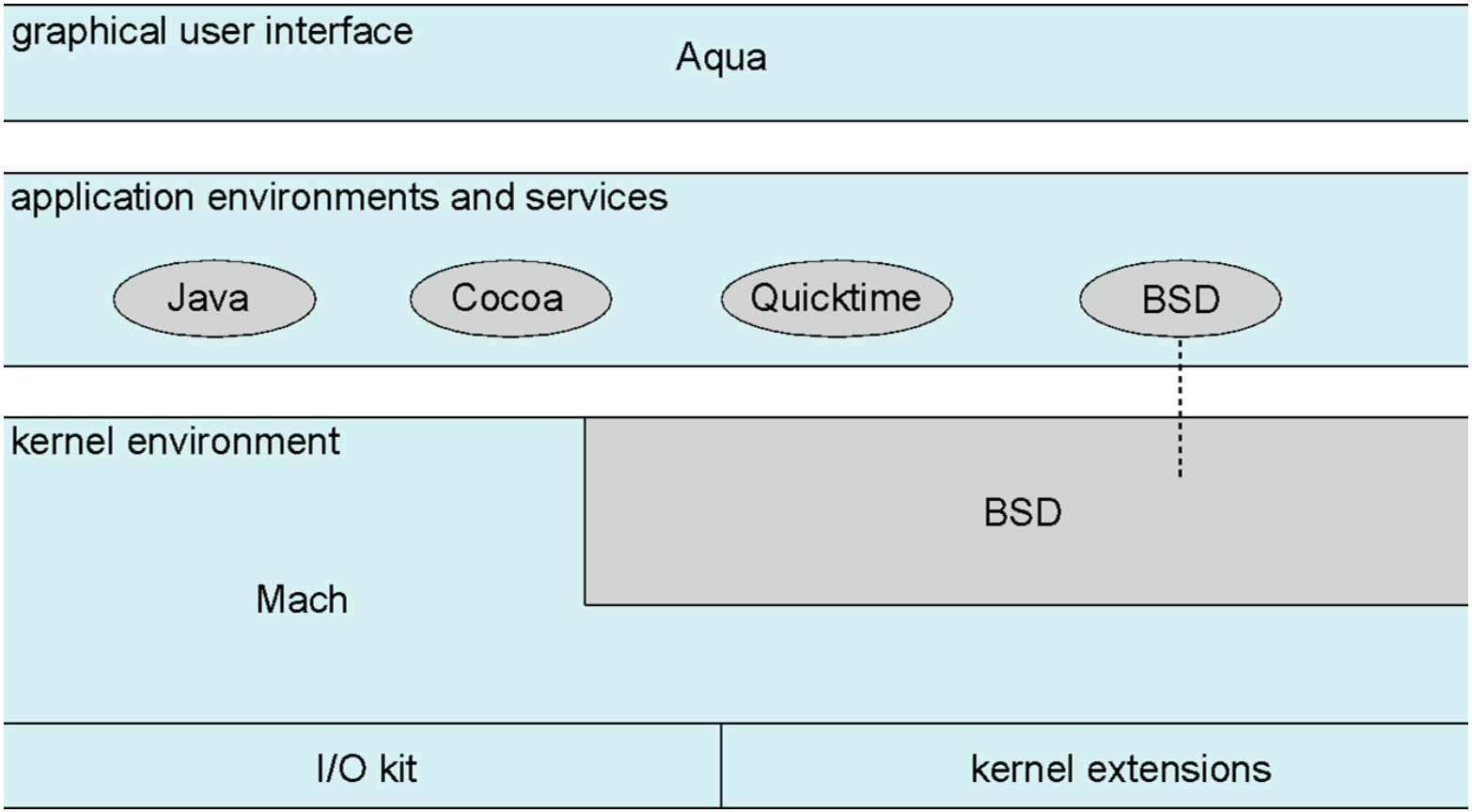
سولاريس وحدات المنهج



أنظمة هجينة

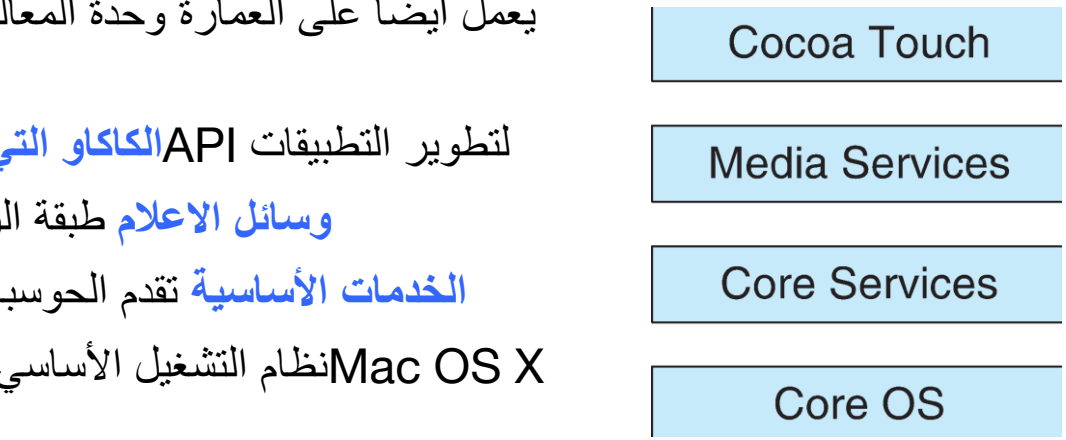
- معظم أنظمة التشغيل الحديثة في الواقع ليست النموذج النقي واحد
  - هجين يجمع بين مناهج متعددة لمعالجة الأداء، والأمن، واحتياجات الاستخدام
  - لينكس، وسولاريس حبات في الفضاء نواة عنوان متجانسة جدا، بالإضافة إلى وحدات للتحميل الديناميكي وظائف
  - نوافذ معظمهم متجانسة، بالإضافة إلى النوية لفرعي مختلف ***شخصيات***
- الهجين، الطبقات، **أكوا** واجهة المستخدم بالإضافة إلى **كاكاو** بيئة البرمجة OS Xأبل ماك
  - مجموعة الإخراج وحدات /يونكس، بالإضافة إلى BSDأقل من نواة تتألف من ماخ النوية وأجزاء loadable حيوي(وتسمى **تمديدات النواة**)

الهيكل Mac OS Xنظام التشغيل



دائرة الرقابة الداخلية

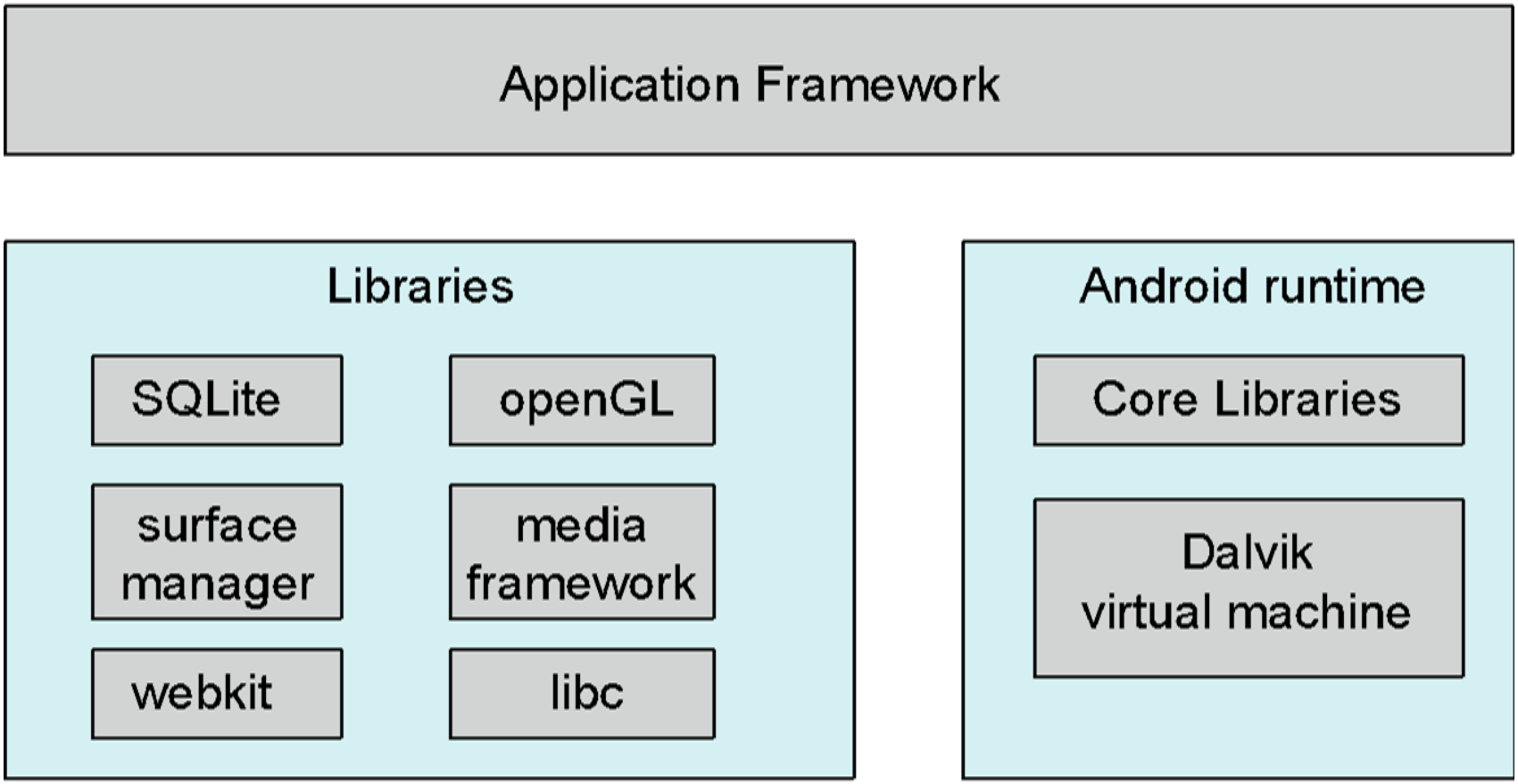
- أبل نظام التشغيل المحمول ل ***ايفون، اى باد***
  - ، إضافة وظائفMac OS Xمنظم على نظام التشغيل
  - أصلا OS X لا يتم تشغيل التطبيقات
    - يعمل أيضا على العمارة وحدة المعالجة المركزية المختلفة (مقابل إنتل ARM)
  - لتطوير التطبيقات APIالكاكاو التي تعمل باللمس الهدف جيم
  - وسائل الاعلام طبقة الرسومات والصوت والفيديو
  - الخدمات الأساسية تقدم الحوسبة السحابية، وقواعد البيانات
  - Mac OS Xنظام التشغيل الأساسي، استنادا إلى نظام التشغيل النواة



الروبوت

- (ومعظمهم من جوجل)وضعت من قبل الاتحاد المفتوح للهواتف النقالة
  - مفتوحة المصدر
- كومة مماثلة لIOS
- وبناء على نواة لينكس ولكن تعديل
  - توفر عملية، والذاكرة، وإدارة جهاز سائق
  - ويضيف إدارة الطاقة
- آلة افتراضية Dalvikوتشمل بيئة تشغيل مجموعة أساسية من مكتبات وفي
  - التطبيقات المتقدمة في جاوة بالإضافة إلى واجهة برمجة التطبيقات الروبوت
    - ثم ترجمة ملفات فئة جافا المترجمة لجافا بايت كود إلى قابل للتنفيذ من يعمل في في Dalvik VM
- LIBC، والوسائط المتعددة، (سكليتي)، قاعدة بيانات (بكت)وتشمل المكتبات الأطر لمتصفح الويب أصغر

الروبوت العمارة



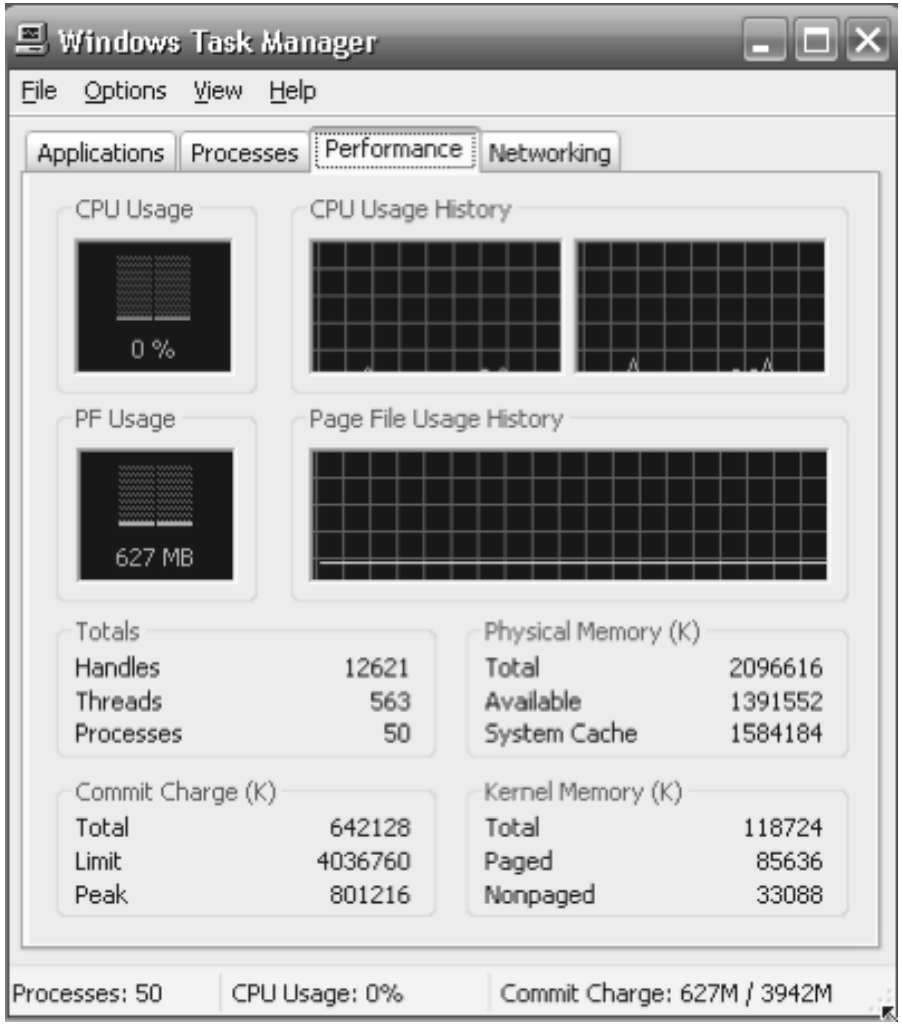
تصحيح التشغيل نظام

- **التصحيح** وإيجاد وإصلاح الأخطاء، أو **الأخطاء**
- تولد انظمة التشغيل **تسجيل ملفات** تحتوي على معلومات خطأ
- يمكن أن فشل تطبيق تولد **تفريغ الأساسية** ملف الذاكرة التقاط عملية
- فشل نظام التشغيل ويمكن أن تولد **اصلاح العطل** ملف يحتوي على ذاكرة kernel
- ما وراء حوادث، ويمكن ضبط الأداء تحسين أداء النظام
  - في بعض الأحيان باستخدام ***قوائم التتبع*** من الأنشطة، وسجلت للتحليل
  - **جانبي** هو أخذ العينات الدورية للمؤشر التعليمة للبحث عن الاتجاهات الإحصائية

قانون كيرنيغان في: "التصحيح هو ضعف الجهد الذي يبذله كتابة التعليمات البرمجية في المقام الأول. لذا، إذا كنت أكتب الكود كما بذكاء وقت ممكن، كنت، بحكم التعريف، وليس ذكي بما فيه الكفاية لتصحيح ذلك".

ضبط الأداء

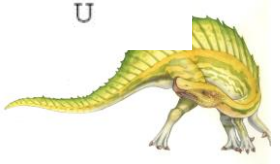
- تحسين الأداء عن طريق إزالة الاختناقات
- نظام التشغيل يجب أن يكون وسيلة للحوسبة والتدابير التي تظهر من سلوك النظام
- أو "أعلى"على سبيل المثال، برنامج ويندوز مدير المهام



DTrace

- أداةDTrace في سولاريس، فري، نظام التشغيلMac OS X يسمح الأجهزة يعيش على نظم الإنتاج
- تحقيقات النار عند تنفيذ التعليمات البرمجية ضمن مزود، واستولت على بيانات حالة وإرساله إلى المستهلكين من تلك المجسات
- مثال التاليةXEventsQueued خطوة استدعاء النظام من مكتبةLIBC إلى نواة والعودة

```
# ./all.d 'pgrep xclock' XEventsQueued
dtrace: script './all.d' matched 52377 probes
CPU FUNCTION
  0 -> XEventsQueued                          U
  0   -> _XEventsQueued                       U
  0     -> _X11TransBytesReadable             U
  0     <- _X11TransBytesReadable             U
  0     -> _X11TransSocketBytesReadable       U
  0     <- _X11TransSocketBytesreadable       U
  0     -> ioctl                              U
  0       -> ioctl                            K
  0         -> getf                           K
  0           -> set_active_fd                K
  0           <- set_active_fd                K
  0         <- getf                           K
  0         -> get_udatamodel                 K
  0         <- get_udatamodel                 K
...
  0         -> releasef                       K
  0           -> clear_active_fd              K
  0           <- clear_active_fd              K
  0           -> cv_broadcast                 K
  0           <- cv_broadcast                 K
  0         <- releasef                       K
  0       <- ioctl                            K
  0     <- ioctl                              U
  0   <- _XEventsQueued                       U
  0 <- XEventsQueued                          U
```



DTrace

- إلى كمية قياسية من DTraceكود الزمن كل عملية مع هوية المستخدم على وحدة)في تشغيل وضع 101 في نانو ثانية (المعالجة المركزية

```
sched:::on-cpu
uid == 101
{
   self->ts = timestamp;
}

sched:::off-cpu
self->ts
{
   @time[execname] = sum(timestamp - self->ts);
   self->ts = 0;
}
```

```
# dtrace -s sched.d
dtrace: script 'sched.d' matched 6 probes
^C
    gnome-settings-d              142354
    gnome-vfs-daemon              158243
    dsdm                          189804
    wnck-applet                   200030
    gnome-panel                   277864
    clock-applet                  374916
    mapping-daemon                385475
    xscreensaver                  514177
    metacity                      539281
    Xorg                         2579646
    gnome-terminal               5007269
    mixer_applet2                7388447
    java                        10769137
```

Figure 2.21 Output of the D code.

الجيل نظام التشغيل

- تم تصميم أنظمة التشغيل لتشغيلها على أي فئة من الآلات؛ يجب أن يتم تكوين النظام لكل موقع كمبيوتر معين

- برنامج يحصل على معلومات بشأن تكوين محدد من نظام الأجهزة **SYSGEN**
  - تستخدم لبناء نظام خاص بتجميع نواة أو نظام ضبطها
  - يمكن رمز أكثر كفاءة عام من نواة واحدة عامة

التمهيد نظام

- عندما قوة تهيئة على النظام، ويبدأ التنفيذ في موقع ذاكرة ثابت
  - تستخدم لعقد رمز التمهيد الأولي ROMالبرامج الثابتة
- يجب أن يتم نظام التشغيل المتاحة على الجهاز حتى الأجهزة يمكن تشغيله
  - **محمل التمهيد**، مخزن في **ذاكرة للقراءة فقط** أو -قطعة صغيرة من التعليمات البرمجية يقع النواة، يحمل في الذاكرة، وبدء تشغيله **EEPROM**
  - ROM ، أحيانا عملية من خطوتين حيث **كتلة التمهيد** في موقع ثابت تحميلها من قبل كود الذي يحمل محمل التمهيد من القرص
- محمل التمهيد المشتركة، **نكش**، يسمح اختيار نواة من أقراص متعددة، إصدارات، وخيارات النواة
- يحمل نواة والنظام هو ثم **تشغيل**