Implementasi Sistem Operasi

MS-DOS & Windows
MS-DOS

What is MS-DOS?

• MS-DOS stands for Microsoft Disk Operating System.

• MS-DOS controls the computer’s hardware and provides an environment for programs to run.
  – Run in **single** user, **stand alone** desktop computer
  – **Simple**, only have few instructions
  – For Intel 8086, 8088, 80186, & 80286
MS-DOS functionality

• 1. MS-DOS **controls** the flow of information between you and the computer (translator).
• 2. MS-DOS allows you to **store** information on your computer.
• 3. MS-DOS allows you to **retrieve** information stored on your computer.
• 4. MS-DOS **interprets** and **translates** the software you have on your computer.
• 5. MS-DOS gives you **access** to all its function (i.e. saving, copying, and printing files).
History of MS-DOS

• Seattle Computer Products 86-DOS was loosely based on CP/M (Control Program for Microcomputers).

• In 1981, Microsoft purchased 86-DOS, and renamed it as “MS-DOS.” From the start, MS-DOS was intended for IBM’s personal computers, designed for the Intel 8086/8088 CPUs.

• MS-DOS shipped with the original IBM PC in 1981, and quickly grew to be the OS of choice for both IBM PC’s and “clones.”

• MS-DOS came to provide not only the basic disk features provided by CP/M, but also some UNIX-like features (I/O redirection, piping, filters).

• IBM chose MS-DOS and called it PC-DOS for 16-bit computer
MS-DOS Version

- 1.0 in 1981 -> CP/M, support only one directory
- 1.1 in 1982 -> in double sided 5.5 inch disks
- 2.0 in 1983
- 3.0 in 1984 -> memory requirement to 36K
- 3.1 in 1984 -> support networking
- 3.2 in 1986 -> in 3.5 inch disks
- 3.3 in 1987
- 4.0 in 1988 -> support HDD > 32MB
- 5.0 in 1991
- 6.0 in 1993
- 6.22 in 1994 -> the last DOS until now
Major Features NOT Included in DOS

1. multitasking, CPU scheduler
2. multiprocessor support
3. threads
4. multiuser
5. hard/soft real-time
6. fault tolerance
7. deadlock management
8. paged memory
9. graphical user interface
10. file system security

Major Features Included in DOS

FAT file system:
- file attributes
- simple file locks

Direct, complete hardware control:
- memory/device access
- interrupt vector overrides
3 layer of DOS

- BIOS : berhubungan dengan I/O
- DOS KERNEL : berisi runtime untuk berhubungan dengan disk
  Berisi fungsi-fungsi utama (system calls), proprietary Microsoft
  (MSDOS.SYS)
- Command.com -> Shell, berfungsi mengeksekusi perintah

DOS menjadi dasar Windows
3.0, Windows 3.1, Windows
95, Windows 98 dan 98SE
serta Windows Me
Command.com

• Merupakan DOS shell
• Mengeksekusi perintah-perintah dan mengembalikan respon atas perintah tersebut
• Command.com **bukanlah** MS-DOS, hanya merupakan salah satu komponen utamanya
• Jeleknya user harus menulis perintah secara lengkap
  – Sampai akhirnya ada perintah **DOSKEY**
• Sekarang MS-DOS sudah menjadi bagian dari Windows (bukan inti)
  – Memiliki fitur **copy/paste**
  – Berupa program **external** emulasi bernama **cmd.exe**
Memory management

- **Simple** job karena single user
- Menggunakan **first fit** memory allocation
- ROM untuk menyimpan perintah **BIOS**
- RAM untuk menyimpan perintah **utama**
- Bagian awal memory digunakan untuk menyimpan *interrupt, BIOS interface, dan kernel*
- Bagian selanjutnya untuk menyimpan *program-resident* dan *program user* yg bisa *dioverwrite* program lainnya
DOS Memory Organization and OS Loading

Diagrams derived from Ray Duncan, *Advanced MS-DOS Programming*

<table>
<thead>
<tr>
<th>ROM Bootstrap Program</th>
<th>top of RAM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disk Bootstrap Program</td>
<td></td>
</tr>
<tr>
<td>DOS kernel, from MSDOS.SYS</td>
<td>(temporary location)</td>
</tr>
<tr>
<td>SYSINIT, from IO.SYS</td>
<td></td>
</tr>
<tr>
<td>BIOS, from IO.SYS</td>
<td></td>
</tr>
<tr>
<td>Interrupt vectors</td>
<td>0x00400</td>
</tr>
<tr>
<td></td>
<td>0x00000</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ROM Bootstrap Program</th>
<th>top of RAM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transient part of COMMAND.COM</td>
<td></td>
</tr>
<tr>
<td>TSR</td>
<td></td>
</tr>
<tr>
<td>Resident part of COMMAND.COM</td>
<td></td>
</tr>
<tr>
<td>Drivers, file control blocks and disk buffer cache</td>
<td></td>
</tr>
<tr>
<td>DOS kernel</td>
<td></td>
</tr>
<tr>
<td>BIOS</td>
<td></td>
</tr>
<tr>
<td>Interrupt vectors</td>
<td>0x00400</td>
</tr>
<tr>
<td></td>
<td>0x00000</td>
</tr>
</tbody>
</table>
Main Memory management

• Versi MS DOS pertama mengalokasikan semua memory ke resident application, tapi akhirnya tidak mencukupi
• Kemudian hanya program yang berekstensi COM saja yang mendapat jatah memory Terminate & Standby Resident
• Program EXE mendapat jatah sepenuhnya saja
  – Memiliki min dan max allocation (statis)
  – Jika mungkin diberi minimum allocation
  – Jika kebutuhan lebih besar dari max, maka tidak bisa dijalankan (out of memory exception)
• Khusus untuk COM, memory TSR boleh diisi oleh lebih dari 1 file / program
  – Gunanya untuk mempercepat kinerja program aplikasi
  – Bisa jadi hanya sebagian file / program yang dieksekusi pada suatu saat
• DOS tidak mengijikan program compiler untuk menambah memory sesuka hati, kecuali ditulis dgn bahasa C & assembly
Memory Block Allocation

- DOS awal menggunakan **first fit & linked list**
- Namun versi 3.3 keatas menggunakan sudah menggunakan **best fit**
- Ukuran blok bervariasi, paling kecil 16 bytes
- **5 byte pertama** mendefinisikan karakteristik struktur blok secara keseluruhan

![Diagram of memory block allocation]
Overlays -- The Concept

- **Overlaying** is the technique of loading different portions of a program into the same memory area.

- Overlay programming techniques were first developed and refined on mainframes in the **1960’s**.

- This allowed MS-DOS developers to split a program up, that was otherwise would not fit in conventional memory.
Process Management

• MS-DOS tidak support multitasking, karena hanya bersifat single user, single task environment

• Satu pekerjaan harus selesai terlebih dahulu baru dilanjutkan dengan perkejaan lain
  – Walaupun parent program memanggil child, namun ketika child dijalankan, parent akan sleep, setelah child selesai, baru parent akan run lagi
Interrupt & device management

• MS-DOS menghandle 3 interrupt:
  – Internal interrupt: akibat dari kernel
  – External interrupt: akibat dari periferal device controller
  – Software interrupt: akibat dari software yang mengakses fungsi DOS dan BIOS

• MS-DOS dapat menghandler driver untuk periferal sederhana, pada jaman ini DOS belum mengenal istilah “plug n play device”
  – Driver beberapa dikenali dan diproduksi oleh MS
  – Vendor dapat membuat driver sendiri namun tidak bisa bersifat generic seperti driver-driver sekarang
  – Harus selalu diload saat booting
File management

• DOS support sequential, direct, dan index sequential file organizations.
  – Sequential -> fixed/variabel length records
  – Direct & index sequential -> fixed length records
• Filename convention
  – 8 filename, dan 3 extensions
  – Jika lebih dari 8 huruf akan diganti jadi ~1, ~2 dst
  – Tidak mendukung spasi, harus menggunakan “ “
• Mendukung full path dan relative path
• FORMAT creates: the boot record, root directory, dan informasi File Allocation Table
Keterangan

- **Boot** record: first sector of every logical disk (whether it’s a physical / virtual disk)
  - Contains: disk boot program / table disk characteristics
- **Root** directory: direktori kerja utama
  - Contains: list of system subdirectories and files, system generated configuration files, user generated booting instruction
  - Informasi penting: filename, extension, size, date time, starting cluster, attributes codes
- **FAT**: File Allocation Table, table used to track segment of a file
  - Contains: sector capabilities to be allocated, informasi sector/cluster allocated
File management

- File / Disk **Compaction**: DEFRAG.EXE
  - Mulai DOS 6.x
- File **checker**: CHKDSK untuk memeriksa kondisi disk dan file-filanya
- DOS juga support **Batch File**: .BAT
  - Misalnya AUTOEXEC.BAT
- Mendukung **Redirection**: > dan >>
- Mendukung **Pipe**: |
Brief Description of Internal MS-DOS commands

• CALL Calls one batch program from another.
• CHDIR Displays the name of or changes the current directory.
  CD Displays the name of or changes the current directory.
• CLS Clears the screen.
• COPY Copies one or more files to another location.
• DATE Displays or sets the date.
• DEL Deletes one or more files.
• ERASE Deletes one or more files.
• DIR Displays a list of files and subdirectories in a directory.
• ECHO Displays messages, or turns command echoing on or off.
• EXIT Quits the COMMAND.COM program (command interpreter).
• GOTO Directs MS-DOS to a labelled line in a batch program.
• IF Performs conditional processing in batch program.
• MKDIR Creates a directory.
• MD Creates a directory.
• PATH Displays or sets a search path for executable files.
• PAUSE Suspends processing of a batch file and displays a message.
• PROMPT Changes the MS-DOS command prompt.
• REM Records comments (remarks) in a batch file or config.sys.
• RENAME Renames a file or files. REN Renames a file or files.
• RMDIR Removes a directory. RD Removes a directory.
Brief Description of Internal MS-DOS commands continues....

• SET Displays, sets, or removes MS-DOS environment variables.
• TIME Displays or sets the system time.
• TYPE Displays the contents of a text file.
• VER Displays the MS-DOS version.
• VOL Displays a disk volume label and serial number.
EXTERNAL commands

• External commands are MS-DOS utilities / programs.
• These are the .EXE or .COM programs located on your hard drive.
• They are normally placed under C:\DOS, the default directory or C:\Windows
• MS-DOS will load external commands if and only if you instruct to execute manually them at the DOS prompt and /or in a batch program.
Windows Personal Family

- Berbasis GUI
- Windows 1.0 -> DOS (1985)
- Windows 3.0 -> GUI Pertama, butuh DOSE (1990)
- Windows 3.1 -> Peningkatan dari 3.0 (1992)
- Windows for Workgroups -> GUI for networking (1992)
- Windows 95 (1995)
- Windows 98 (32 bit 1998)
- Windows ME (2000)
- Windows XP Home & Professional (2001)
- Windows Vista (2007)
- Windows 7 (2009)
Windows Networking Family

- Windows NT 3.1 (1993)
- Windows NT 3.5 (1994) RAM 4MB
- Windows NT 4 (1996) DCOM
- Windows 2003 (2003) (.NET 1.1)
- Windows 2008 (2008) reduced power consumption
- Windows 2008 R2 (2009)
Keunggulan Windows

• **Extensibility**: bersifat layered architecture
  – user mode & kernel mode

• **Portability**: running on different machine & processors configuration (32 & 64 bit)
  – Written in C & C++
  – Support Hardware Abstraction Layer (HAL)

• **International support**
  – Supports different locales via the national language support (NLS) API
Keunggulan Windows

• **Reliability**: robustness
  – Menggunakan NTFS: bisa recovery file dan security
  – hardware protection for virtual memory, software protection mechanisms for OS resources

• **Compatibility**
  – Applications that follow the IEEE 1003.1 (POSIX) standard can be complied to run on XP without changing the source code
  – Support FAT, CDFS, dan NTFS
  – Verification on hardware’s vendor

• **Performance**: respond quickly to CPU-bound application

• **User interface**: easy to learn and understand
Kelemahan Windows

• Tidak gratis
• Rentan virus dan keamanan
• Kurang stabil
• Update termasuk lambat
• Tidak memiliki proteksi akses yang baik secara default
• Closed source
• Dst.. Dst...
Windows NT in General

**USER MODE**

- System support processes
- Service processes
- User applications

**KERNEL MODE**

- Environment subsystem server processes
- Core Windows subsystem DLLs
  - Kernel32.dll, Advapi32.dll, User32.dll, Gdi32.dll
  - NTDLL.DLL (used by all except User32.dll and Gdi32.dll)

**System Service Dispatcher**

- EXECUTIVE API (Kernel-Mode Callable Interfaces)
  - I/O Manager
  - File System
  - Device Drivers
  - File System Cache
  - Object Manager
  - PnP Manager
  - Security Reference Monitor
  - Virtual Memory Manager
  - Processes & Threads
  - Configuration Manager
  - Local Procedure Call

**KERNEL**

- NTOSKRNL.EXE

**WIN32K.SYS**

**HARDWARE ABSTRACTION LAYER (HAL)**

- HAL.DLL

**WINDOWS USER, GDI**

- Graphics Drivers
System Components — Kernel

- Foundation for the *executive* and the *subsystems*
- Never paged out of memory; execution is never preempted
- Four main responsibilities:
  - thread scheduling
  - interrupt and exception handling
  - low-level processor synchronization
  - recovery after a power failure
- Kernel is **object-oriented**, uses two sets of objects
  - *dispatcher objects*: control dispatching and synchronization
    - events, mutexes, semaphores, threads and timers
  - *control objects*
    - asynchronous procedure calls, interrupts, power notify, power status, process and profile objects
Executive components – Privileged mode

• Many components supported
  – Object Manager
  – Security Reference Manager
  – Process Manager
  – Plug and Play Manager
  – Virtual Memory Manager
  – Local Procedure Call facility
  – I/O Manager
    • Device Drivers
  – Window Manager
Executive

• Object manager
  – Manages all **objects** known to OS
    • (processes, threads files, directories, semaphores, I/O devices, timers)
  – Allocates a block of **virtual memory** from the **kernel** space when object is created, returns it to the free list when object de-allocated;
  – Keeps **track** of objects
  – Manages **name space** in which newly created objects may be placed so that they **can be referred** to later
Components of Executive

• I/O Manager
  – Framework for managing I/O devices, provides device-independent I/O services
  – File systems are technically device drivers under control of I/O manager
    • FAT and NTFS

• Process Manager
  – Handling of processes, threads, including creation and termination
  – Key to multiprogramming in Windows
Components of Executive

• Memory Manager
  – Demand-paged virtual memory architecture
  – Mapping of virtual pages onto physical page frames
  – Enforces protection rules that restrict each process to use its own pages

• Security manager
  – Elaborate security mechanisms, e.g., authenticated login

• Cache Manager
  – Keeps most recently used disk blocks in memory

• Power Manager
  – Manages resources for power-savings
  – Turns off monitor, disks after they have been idle for a while
  – Manages battery usage on laptops and takes action when the battery is about to run out
**HAL (Hardware Abstraction Layer)**

<table>
<thead>
<tr>
<th>Device registers</th>
<th>Device addresses</th>
<th>Interrupts</th>
<th>DMA</th>
<th>Timers</th>
<th>Spin locks</th>
<th>BIOS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>2.</td>
<td>3. Printer</td>
<td>RAM</td>
<td>Disk</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **Device registers**
- **Device addresses**
- **Interrupts**
- **DMA**
- **Timers**
- **Spin locks**
- **BIOS**

- MOV EAX, ABC
- MOV EAX
- MOV EAX, BAX
- MOV EAX, [LABEL]
- MOV EAX, [LABEL]
- MOV EAX
- MOV EAX, [LABEL]
- MOV EAX, [LABEL]
- MOV EAX
- MOV EAX, BAX
- MOV EAX, [LABEL]
HAL

- Layer to **hide** many of the machine dependencies
- Present OS with **abstract** hardware devices in form of **machine-independent services**
- **HAL does not provide** services to I/O devices such as keyboard, disk, mice.
Environmental Subsystems (Cont.)

- **Logon and Security** Subsystem
  - Authenticates *all* users logged on to Windows XP systems
  - Users are required to have account names and passwords
  - The authentication package authenticates users whenever they attempt to access an object in the system
  - Windows XP uses **Kerberos** as the default authentication package
File System — Recovery

• All file system data structure updates are performed inside transactions that are logged
  – Before a data structure is altered, the transaction writes a log record that contains redo and undo information
  – After the data structure has been changed, a commit record is written to the log to signify that the transaction succeeded
  – After a crash, the file system data structures can be restored to a consistent state by processing the log records
File System — Recovery (Cont.)

• This scheme **does not guarantee** that all the user file data can be recovered after a crash, just that the file system data structures (the metadata files) are **undamaged** and reflect some **consistent** state prior to the crash.

• The log is stored in the **third** metadata file at the **beginning** of the volume.

• The logging functionality is provided by the XP **log file service**.
Programming Interface of Windows

Win32 application program

Win32 Application Programming Interface

- Win32s
  - Windows 3.x
- Windows 95/98/Me
- Windows NT
- Window 2000
Win32 API

- System calls is not public in Windows
- Library procedures (function calls) public
- Win32 philosophy: provide a very comprehensive interface, often with 3-4 ways of doing the same thing, including functions that are not system calls
  - Many Win32 calls create kernel objects (e.g., files, processes, threads).
  - Every call creating an object returns handle to the caller
  - Handles are used to perform any operation on the object
- Windows > 2000 sometimes called ‘object-oriented’ OS
Win32

• Thousands and thousands of calls for the graphical interfaces.
  – GDI
• Calls for creating, destroying, managing, using windows, menus, tool bars, status bars, scroll bars, dialog boxes, icons,
• Calls for drawing geometric figures, filling them in, managing the color palettes they use, dealing with fonts, placing icons on the screen
• Calls for dealing with keyboard, mouse, etc.
Interfaces

• **GDI** – Graphics Devices Interface
  – Handles image management for printer and monitor

• **System services** – interface to executive
  – Accepts true Windows 2000 system calls and calls other parts of executive for execution
Bootstrap of Windows 2000

• Load Windows 2000 as a collection of files into memory
  – Main part of OS – kernel and executive – located in ntoskrnl.exe
  – HAL is shared library located in separate file hal.dll
  – Win32 and GDI are in third file win32k.sys
  – Many device drivers (separately) are loaded (have extension .sys)
    • Device drivers are not part of the ntoskrnl.exe binary, so they can be loaded dynamically when system boots
END

• Presentasi Program (Tugas 2)
• TAS