

# Le DOS



GIF-1001 Ordinateurs: Structure et Applications, Hiver 2016  
Jean-François Lalonde

image: <http://www.computerhistory.org/>

# Le DOS

- DOS: **D**isk **O**perating **S**ystem
- Il existe plusieurs versions de DOS:
  - Apple a sorti le DOS 3.2 en 1979.
  - Le MSDOS 1.0 sortira en 1981...
  - Le MSDOS 8.0 sortira en 2000... c'était la dernière version
  - Aujourd'hui nous parlerons de la version ~2.0

# Pourquoi le DOS?

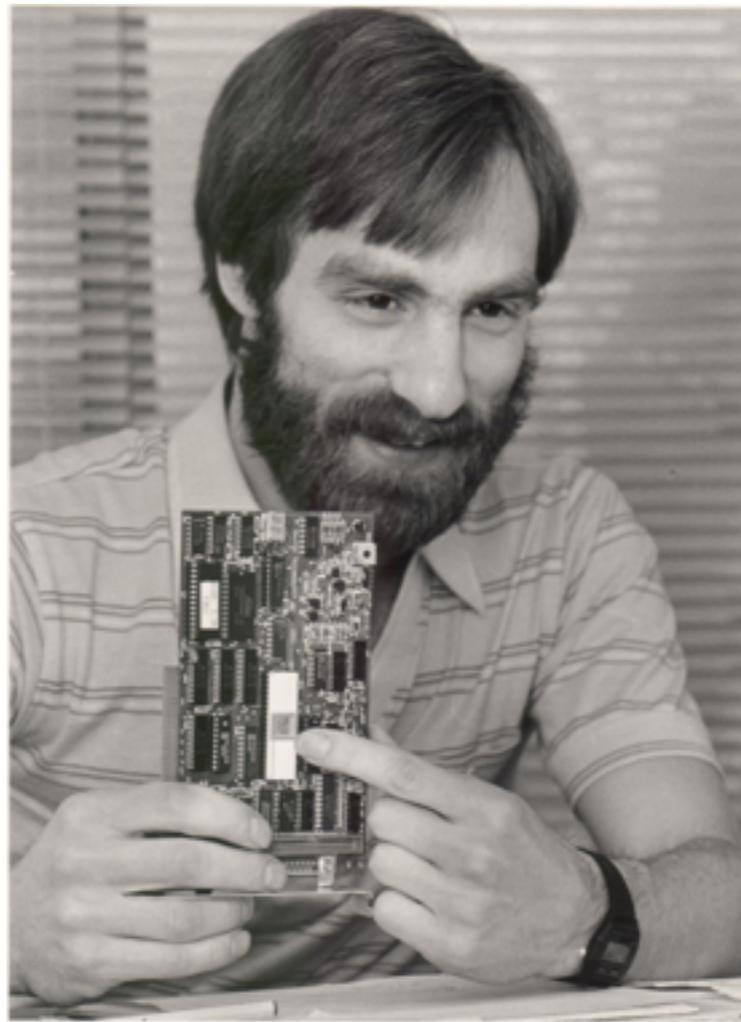
- Le plus simple!
  - Un seul utilisateur à la fois
  - Un seul processus à la fois
  - Le DOS “abandonne” le contrôle du CPU aux programmes

# Un triumvirat désormais “célèbre”

Gary Kildall



Tim Paterson



Bill Gates



# Historique



CP/M (1973)

“Control Program for  
Microcomputers”

**CP/M™**  
LOW-COST  
MICROCOMPUTER  
SOFTWARE

**CP/M™ OPERATING SYSTEM:**

- Editor, Assembler, Debugger and Utilities.
- For 8080, Z80, or Intel MDS.
- For IBM-compatible floppy discs.
- **\$100**-Diskette and Documentation.
- **\$25**-Documentation (Set of 6 manuals) only.

**MAC™ MACRO ASSEMBLER:**

- Compatible with new Intel macro standard.
- Complete guide to macro applications.
- **\$90**-Diskette and Manual.

**SID™ SYMBOLIC DEBUGGER**

- Symbolic memory reference.
- Built-in assembler/disassembler.
- **\$75**-Diskette and Manual.

**TEX™ TEXT FORMATTER**

- Powerful text formatting capabilities.
- Text prepared using CP/M Editor.
- **\$75** Diskette and Manual.

**DIGITAL RESEARCH**  
P.O. Box 579 • Pacific Grove, CA 93950  
(408) 649-3896

QDOS  
Quick & Dirty OS



Gary Kildall



Bill Gates



Tim Paterson



# Historique

<http://www.bloomberg.com/news/articles/2004-10-24/the-man-who-could-have-been-bill-gates>

# Succès commercial!

- A permis à Microsoft de devenir un joueur important
  - (presque) tous les ordinateurs pouvaient être utilisés de la même façon
- Les versions originales de Windows ont été construites “par-dessus” DOS



# Les parties du DOS

- Le MS-DOS est constitué principalement des fichiers suivant:
  - COMMAND.COM: programme gérant l'interface usager et l'exécution/chargement d'applications.
  - IO.SYS: fichier contenant des routines pour les entrées/sorties (des ajouts au BIOS!).
  - MSDOS.SYS: fichier contenant les routines de gestion de fichiers.

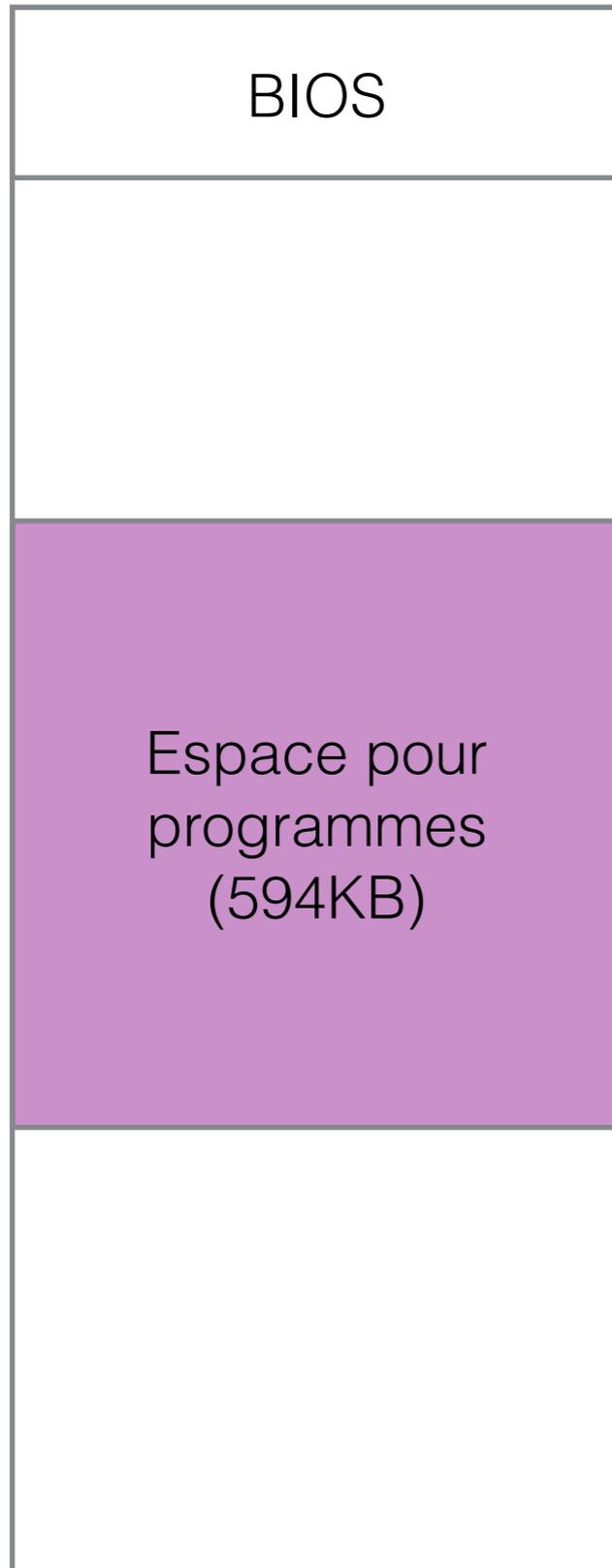
# Démarrage du DOS

- Après le POST, le BIOS lit la Piste 0 du Secteur 0 sur le disque. Les informations contenues sur ce secteur (boot sector) disent si le disque contient un système d'exploitation.
- Si le BIOS trouve un système d'exploitation, il charge le boot sector en mémoire, puis fait un saut vers les instructions du boot sector.
- Les instructions du boot sector chargent IO.SYS et MSDOS.SYS en mémoire.
- Le boot sector passe le contrôle à un programme de MSDOS.SYS.
- MSDOS.SYS exécute CONFIG.SYS qui est un fichier texte que vous pouvez modifier. CONFIG.SYS contient principalement de l'information sur la gestion des fichiers et des périphériques.
- MSDOS.SYS charge COMMAND.COM en mémoire et lui donne le contrôle.
- COMMAND.COM exécute Autoexec.bat qui contient une liste de commande DOS exécutée au démarrage.
- COMMAND.COM affiche un prompt à l'écran et attend des instructions de l'utilisateur.

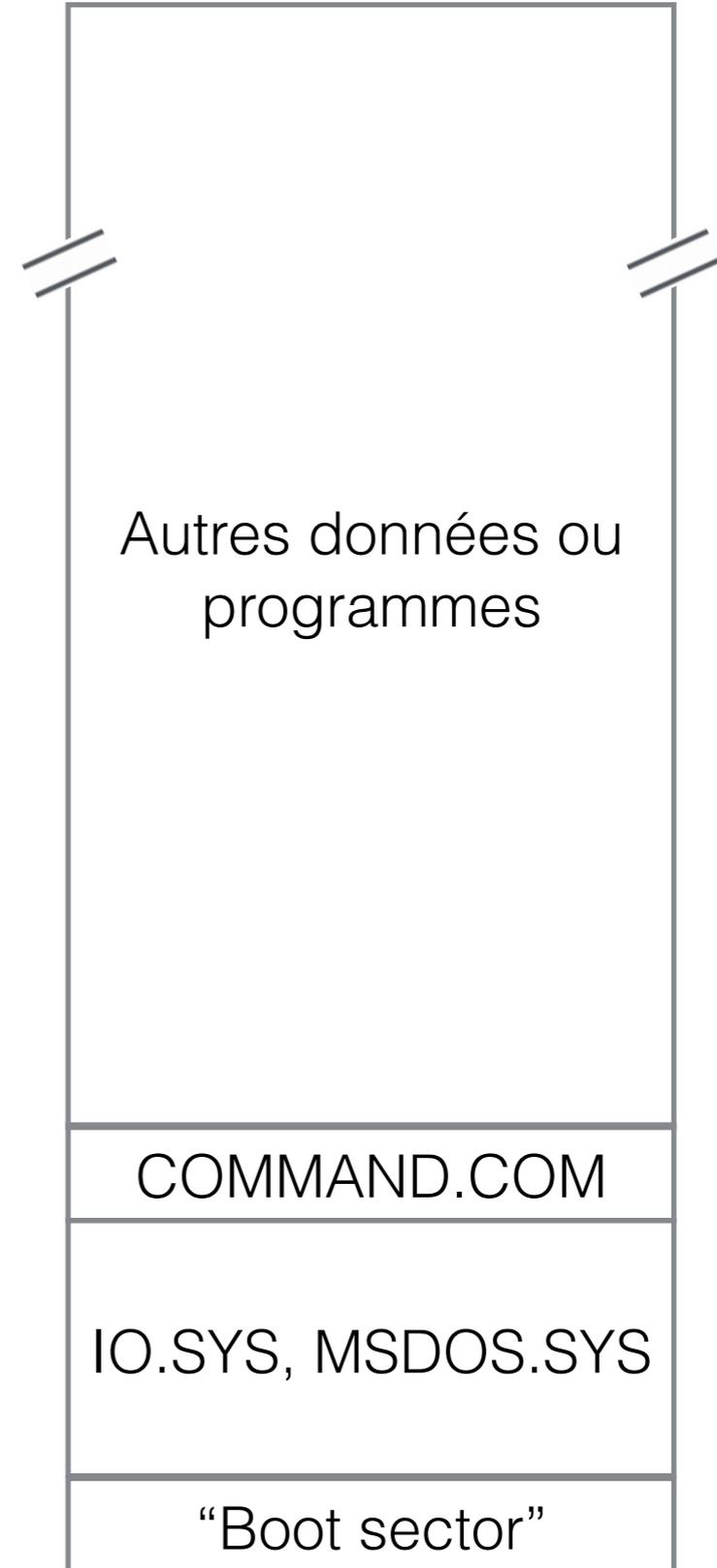
# 1. POST

Mémoire principale (ROM/RAM)

ROM  
(64KB)



Mémoire secondaire (disque dur)

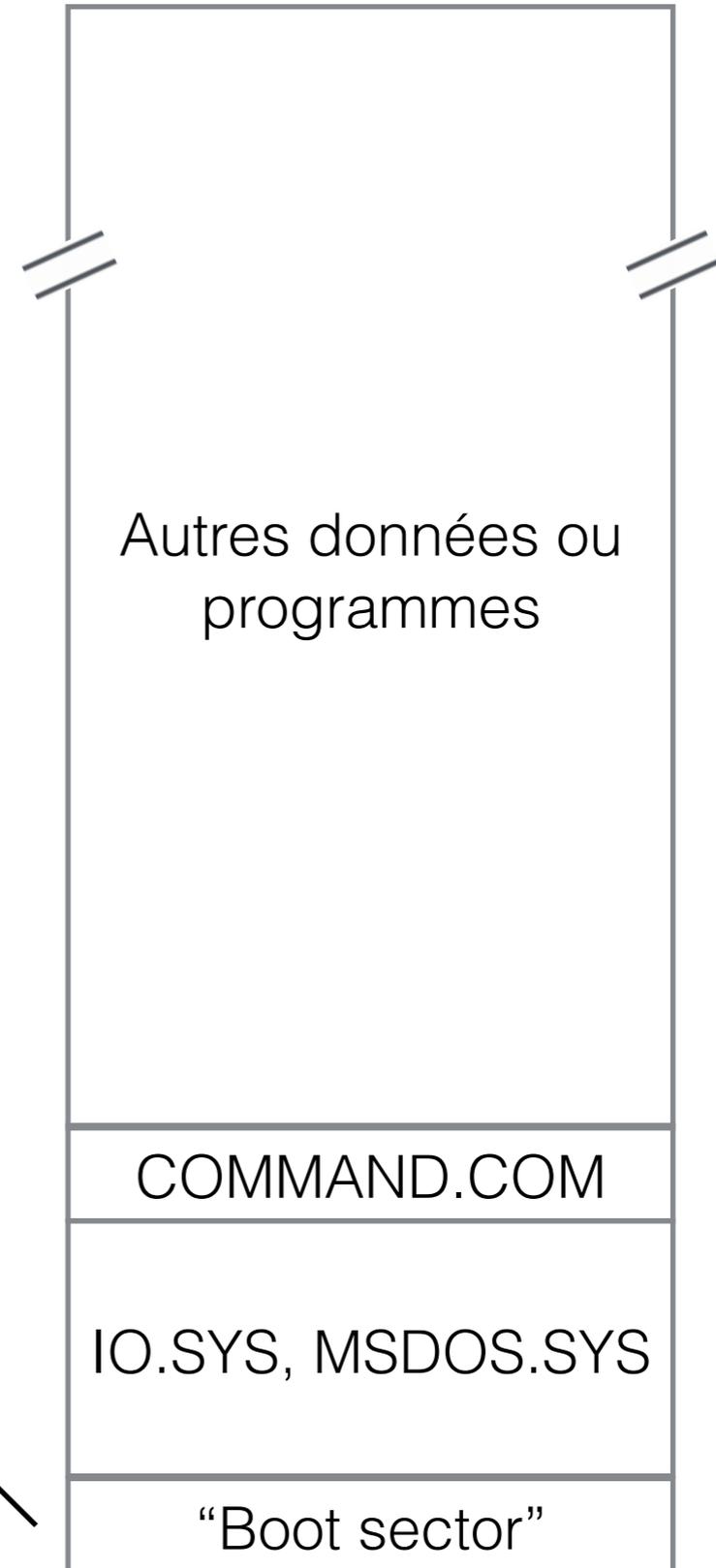
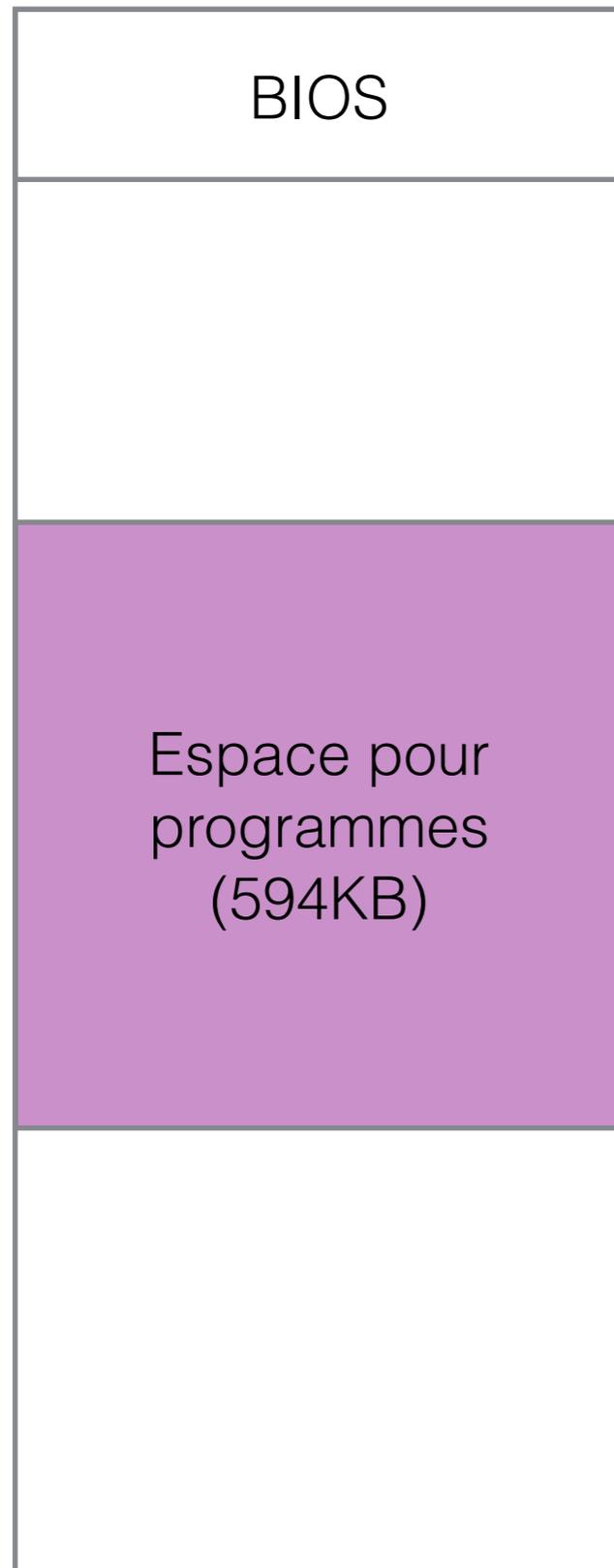


## 2. Copie "Boot sector" en RAM

Mémoire principale (ROM/RAM)

Mémoire secondaire (disque dur)

ROM  
(64KB)

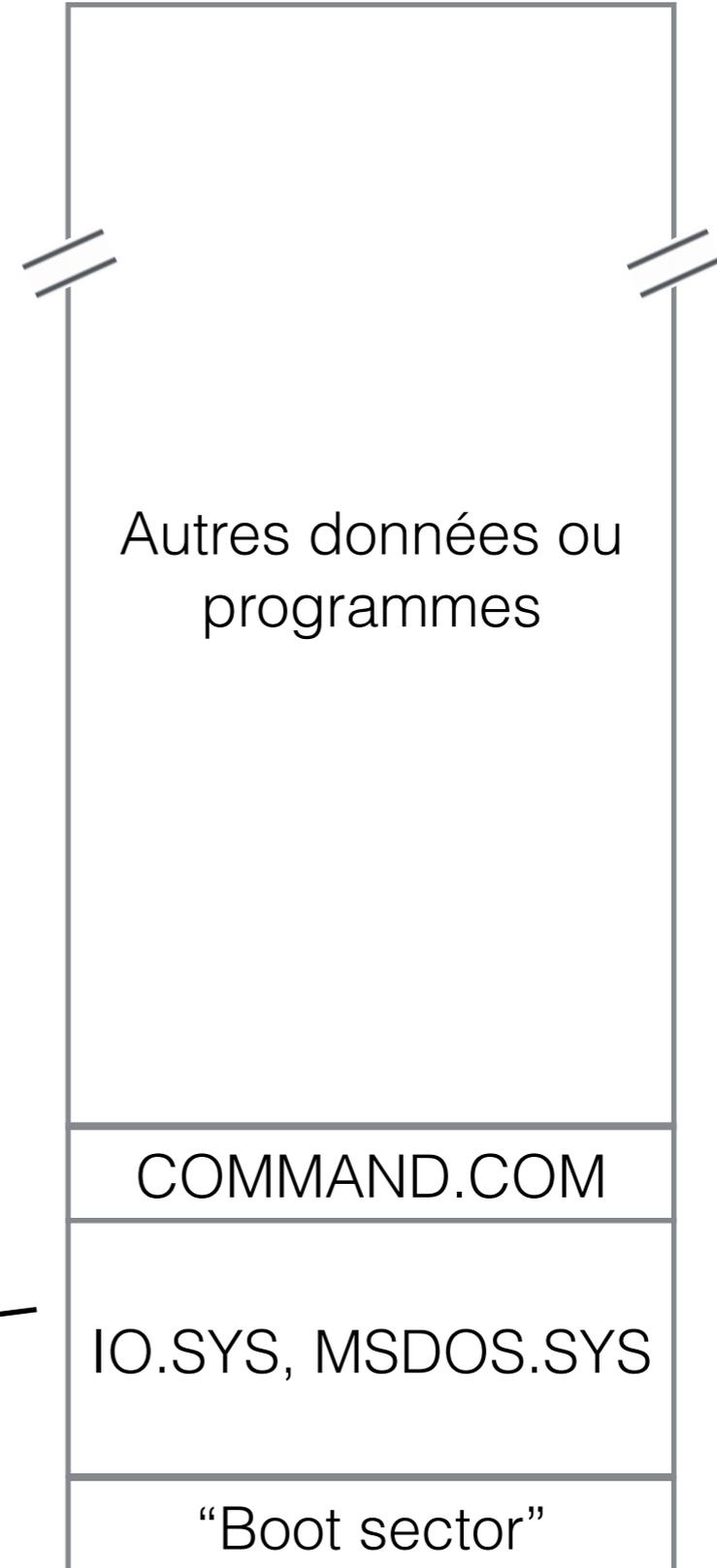


### 3. Le "boot sector" copie IO.SYS et MSDOS.SYS en RAM, exécute MSDOS.SYS

Mémoire principale (ROM/RAM)

Mémoire secondaire (disque dur)

ROM  
(64KB)

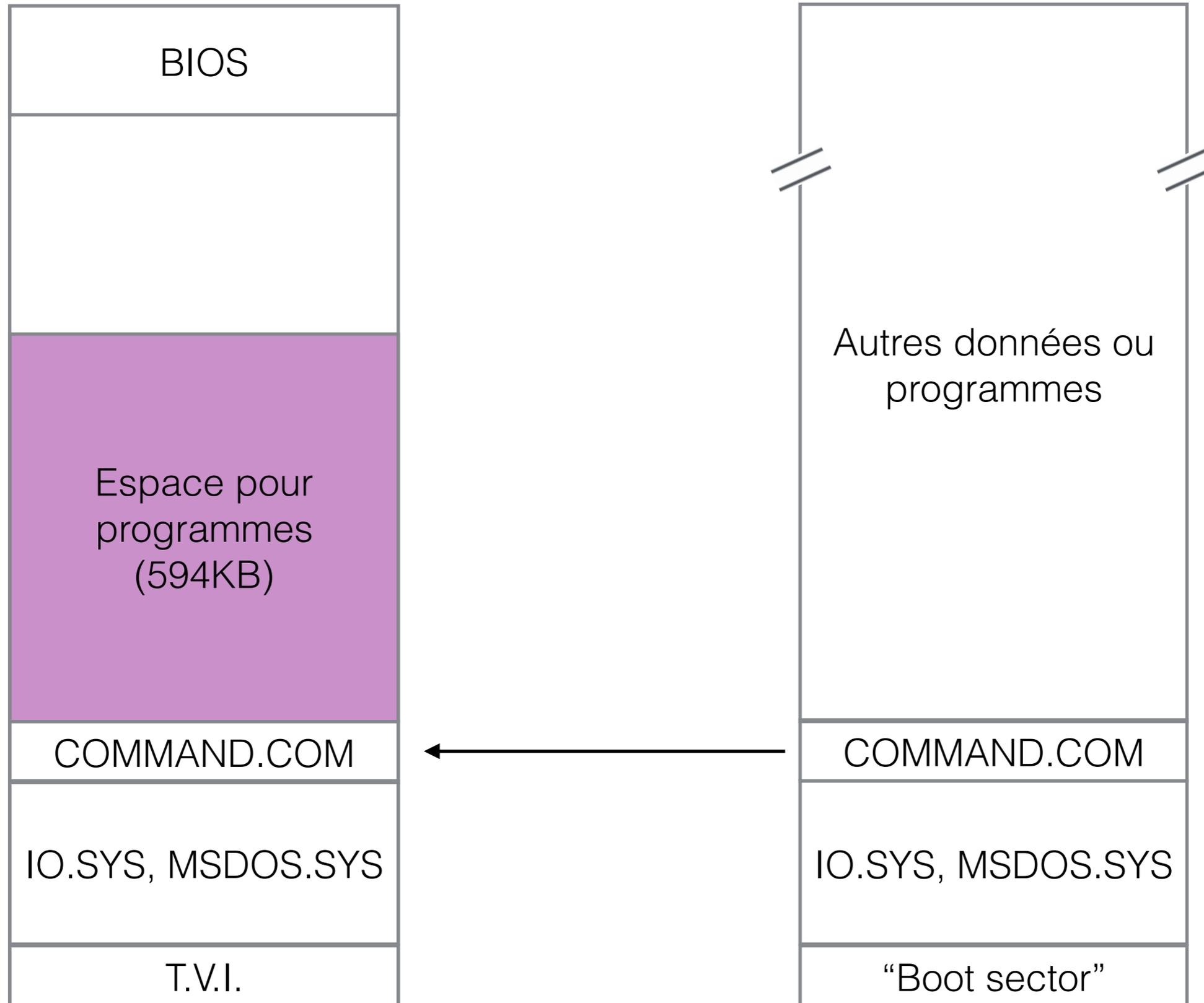


# 4. MSDOS.SYS copie COMMAND.COM en mémoire

Mémoire principale (ROM/RAM)

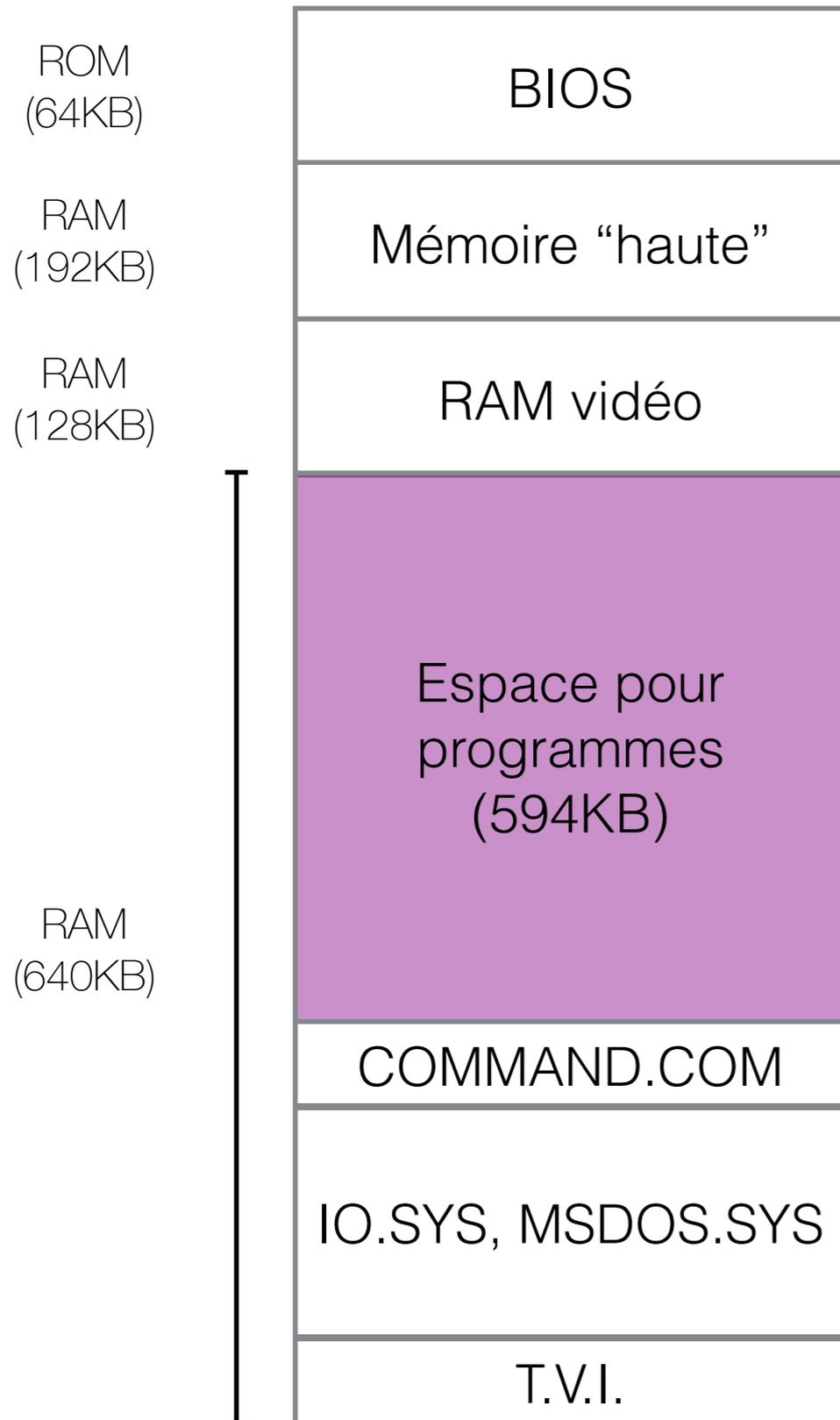
Mémoire secondaire (disque dur)

ROM  
(64KB)



# 5. COMMAND.COM attend les commandes de l'utilisateur

Mémoire principale (ROM/RAM)



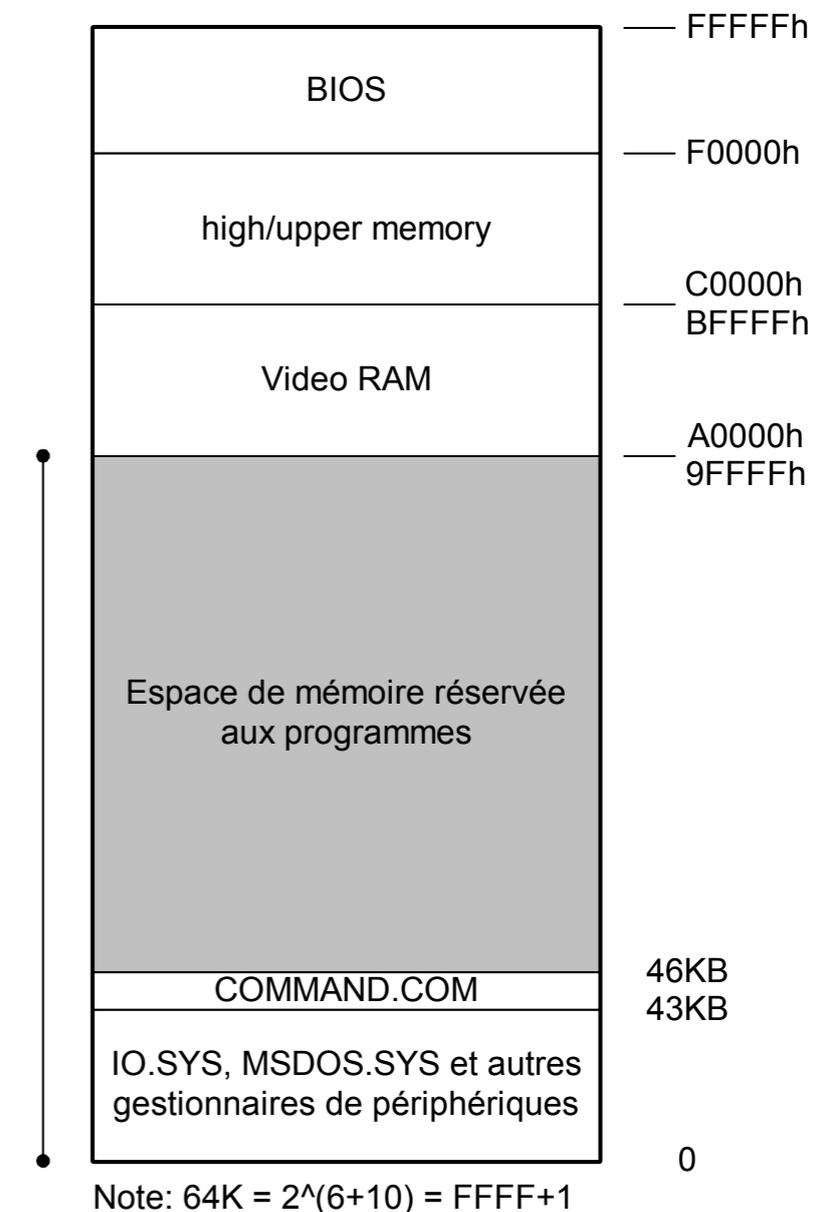
Mémoire secondaire (disque dur)



# Carte mémoire typique de MS-DOS

- Après le démarrage, le bas de la mémoire (46Ko) contient le système d'exploitation.
- Les programmes sont chargés dans un espace restreint de 594Ko.
- La RAM VIDEO sert à l'affichage.
- L'espace mémoire entre 768Ko et 960Ko est la mémoire haute. Cette mémoire peut servir pour augmenter la taille des programmes.
- Le BIOS, dans le haut de la mémoire, est en ROM.

ROM  
64K  
ROM ou RAM  
192K  
RAM  
128K  
RAM  
640K

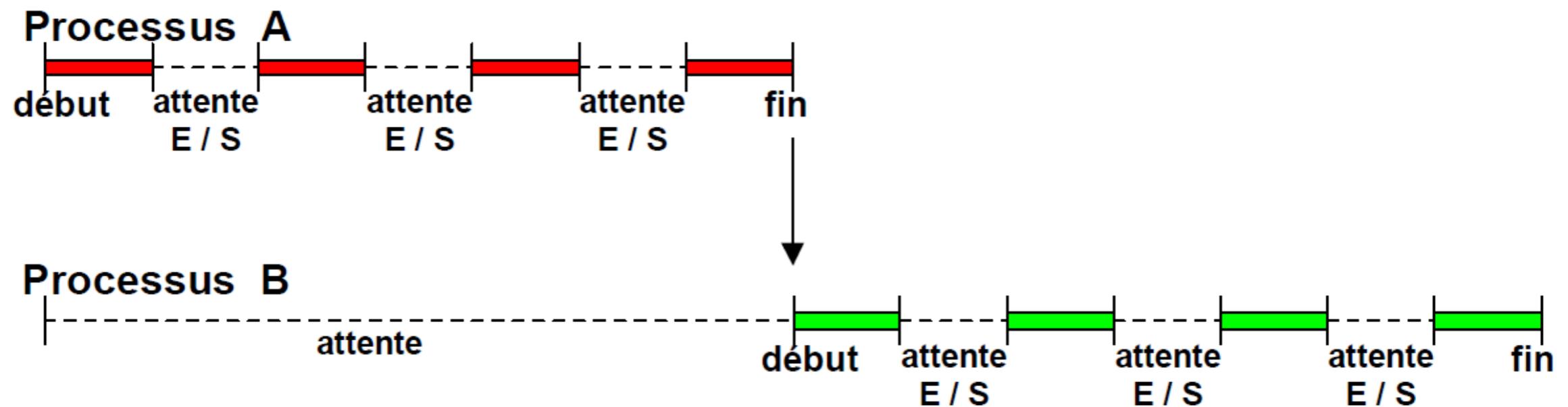


# Particularité (importante) #1

- Un seul programme peut être exécuté à la fois. Ceci inclut le système d'exploitation lui-même!
  - On dit que le système d'exploitation est « non préemptif », c'est-à-dire qu'il ne peut pas interrompre le programme.
  - Le programme est responsable de retourner le contrôle au système d'exploitation (INT 20H ou INT 21H).
- Quels problèmes cela peut-il poser?

# Le DOS est « non-préemptif »

- Partage des ressources?



# Le DOS est « non-préemptif »

- Il y a tout de même quelques façons de retourner au système d'exploitation:
  - Le programme termine et redonne le contrôle au système d'exploitation.
  - Le programme effectue une opération non conforme entraînant une interruption (ex. division par 0).
  - Une interruption externe au programme l'arrête. Par exemple, si l'utilisateur appuie sur certaines touches et que le programme n'a pas désactivé les interruptions, l'interruption causée par l'utilisateur peut arrêter le programme.

# Particularité (importante) #2

- Le DOS est toujours en mémoire (il est « résident »).
- Les programmes peuvent être retirés de la mémoire (non-résident) lorsque leur exécution est terminée.
  - Pour être plus précis, le DOS peut écrire « par-dessus » un programme
- Quels problèmes cela peut-il poser?

# Les programmes peuvent être « écrasés »

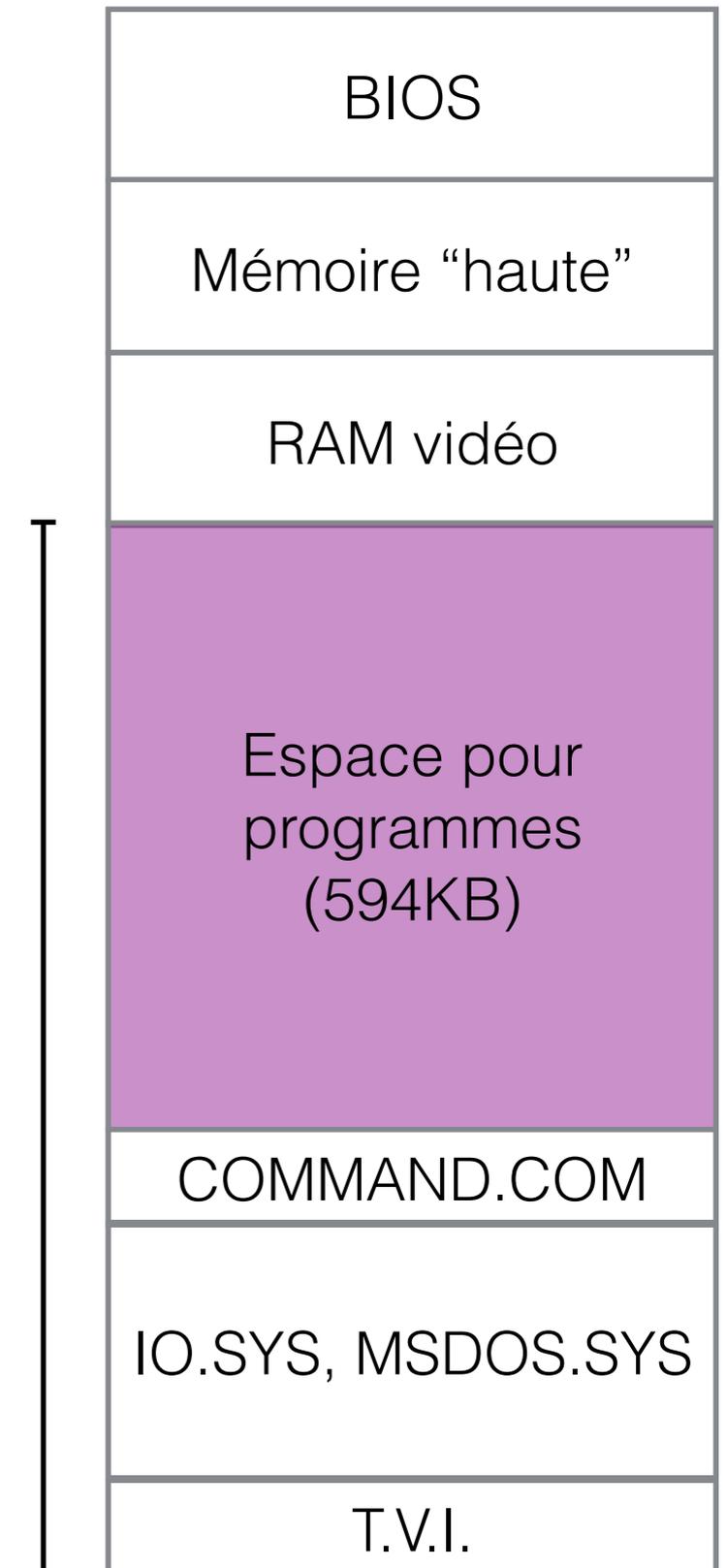
- Si on veut exécuter le même programme à nouveau, on doit constamment le recopier en mémoire à partir du disque dur!
- Pour pallier à ce problème, les designers de DOS ont créé un nouvel interrupt:
  - TSR: Terminate and Stay Resident
  - Le programme peut rester en RAM (mais est forcé à n'utiliser un espace limité)
  - Mais les programmeurs en ont abusé...

ROM  
(64KB)

RAM  
(192KB)

RAM  
(128KB)

RAM  
(640KB)



# L'interface de commande

- MS-DOS a une interface usager de type ligne de commande. Il faut taper la commande à exécuter puis appuyer sur Enter pour envoyer la commande.
- L'interface de commande utilise les routines d'I/Os pour afficher de l'information à l'écran ou obtenir des directives de l'utilisateur (via le clavier).
- COMMAND.EXE interprète les commandes reçues de l'utilisateur. Il les valide, puis il les exécute. Il s'agit d'un interpréteur de commande.
- Pour l'interpréteur de commande, il y a deux types de commandes: les commandes MS-DOS et le reste. Les commandes non reconnues (qui ne sont pas MS-DOS) sont traitées comme des programmes à exécuter.

# Réponse à une commande de l'interface

- Lorsque l'interpréteur de commande reçoit une commande MS-DOS, il l'exécute, puis retourne le contrôle à l'utilisateur.
- Lorsque l'interpréteur de commande reçoit une commande qui n'est pas MS-DOS. Il demande au gestionnaire de fichiers d'exécuter un programme dont le nom et l'emplacement est fourni par l'interpréteur de commande.
- Le gestionnaire de fichiers fait une requête au disque approprié afin de lire vérifier si le répertoire est bon. Une routine d'I/O est utilisée pour accéder au disque qui contient l'arborescence des fichiers.
- La routine d'I/O fournit l'arborescence des fichiers au gestionnaire de fichier.
- Le gestionnaire de fichier vérifie si le programme à exécuter existe à l'emplacement spécifié. Si oui, il fait une requête au disque (une routine d'I/O encore!) afin d'aller chercher le programme.
- La routine d'I/O met le programme en mémoire à l'emplacement spécifié par le gestionnaire de fichier.
- Le gestionnaire de fichier exécute le programme
  - modification des registres, puis branchement au programme
- Le programme s'exécute. Il peut faire des requêtes d'I/Os, de fichiers, ou même exécuter des commandes MS-DOS.
- Le programme se termine (voir plus loin) et une nouvelle commande de l'utilisateur est attendue.



# Exécution d'un programme

- Lors de l'exécution du programme, le système d'exploitation est mis de côté. On dit que le système d'exploitation est "non préemptif", c'est-à-dire qu'il ne peut pas interrompre le programme.
- Le programme doit accéder aux I/Os de l'ordinateur avec les interrupts du BIOS ou du DOS (dans IO.SYS).
- Le programme doit accéder à d'autres fichiers avec les fonctions contenues dans le gestionnaire de fichiers.
- Le programme est responsable de retourner le contrôle au système d'exploitation (INT 20H ou INT 21H).