

# Exemples de SGF

# FAT - VFAT (1)

- Partitions 2 Go
- 3 parties:
  - FAT, éventuellement dupliquée
  - répertoire racine, de taille bornée: 512 entrées de 32 octets
  - objets externes
- Allocation par bloc de taille fixe (cluster) 255 secteurs
- Numéro de cluster sur 16 bits ( 65535)
  - au plus 65535 fichiers ou répertoires
  - quantum de 32 Ko pour les disques de 2 Go
- FAT = état du cluster et chaînage des clusters d'un objet
  - 128 Ko s'il y a 65535 clusters

# FAT - VFAT (2)

- Répertoires => entrées sur 32 octets
  - nom en 8.3, insensible à la casse
  - attributs de l'objet
  - heure et date de dernière modification
  - numéro de premier cluster
  - longueur utile sur 32 bits
  - 10 octets inutilisés
  - entrées "." et ".."
- Extension VFAT => noms sur 255 octets
  - 2 noms: 8.3 et nom long
  - nom long découpé en tranches de 13 octets occupant chacune 1 entrée
  - ⇒ limitation du nombre d'objets dans le répertoire racine

# HFS - HFS Plus de MacOS (1)

- 6 parties pour une partition
  - informations de démarrage: localisation système, taille tampons,...
  - descripteur de volume: nom et informations sur la structure
  - table bitmap d'état d'allocation
  - catalogue des fichiers et répertoires
  - fichier de description des extensions
  - espace des objets externes
- Création d'un objet (fichier ou répertoire) => Num-ID
- Allocation par zone, en nombre et taille quelconque
  - quantum = nombre entier de secteurs
  - blocs numérotés sur 16 bits → 65536 blocs
  - ⇒ table bitmap 8 Ko, quantum taille partition / 65536 → 64 Ko pour 4 Go

# HFS - HFS Plus de MacOS (2)

## ● Fichier

2 parties, partie ressource et partie données, 2 Go chacune  
espaces alloués indépendants

## ● Extensions

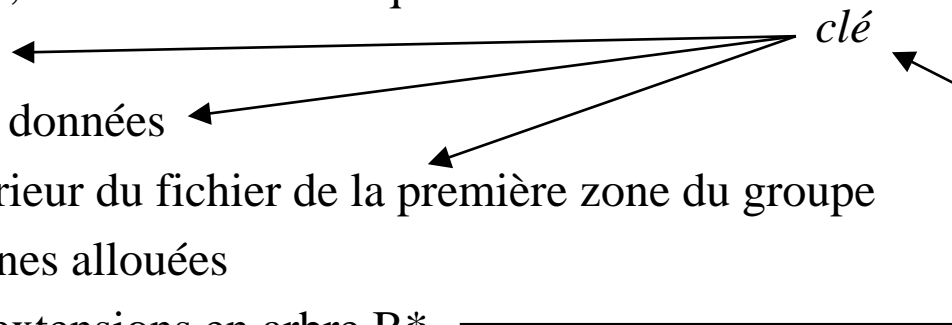
les 3 premières dans le descripteur de fichier

les autres regroupées, 3 par 3, dans le fichier unique des extensions:

- Num-ID du fichier
- indicateur ressource ou données
- numéro logique à l'intérieur du fichier de la première zone du groupe
- description des trois zones allouées

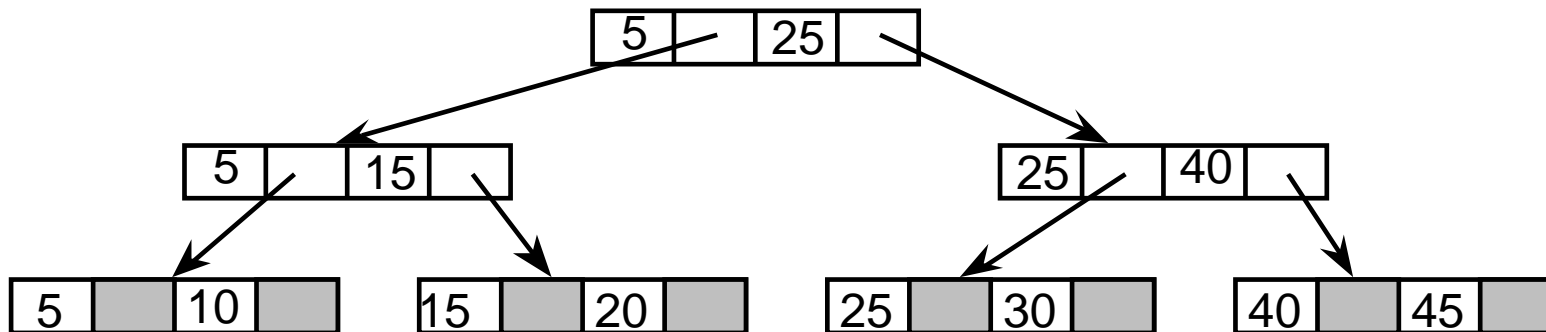
organisation du fichier des extensions en arbre B\*

utilisation du fichier des extensions pour conserver les blocs invalides



# HFS - HFS Plus de MacOS (3)

- Exemple d'arbre B\* utilisé dans HFS



# HFS - HFS Plus de MacOS (4)

- Les répertoires => arbre B\* unique

Num-ID du répertoire parent

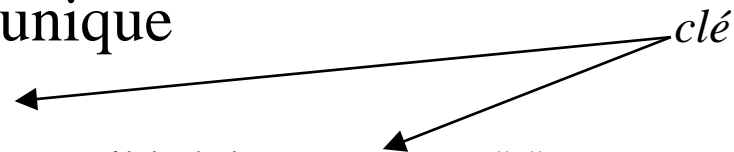
nom de l'objet sur 31 caractères, insensible à la casse, sans ":"

type de l'objet

dates de création, modification, sauvegarde

Num-ID de l'objet lui-même

s'il s'agit d'un fichier, descripteur: tailles, 3 premières zones



- Lien vers le père

Num-ID du répertoire

"" , ou chaîne vide

type lien

Num-ID de son parent

nom du répertoire lui-même

# HFS - HFS Plus de MacOS (5)

- Cohérence du SGF

informations secondaires

informations primaires

table bitmap

zones allouées à des objets

dernier Num-ID attribué

les Num-ID des objets existants

⇒ indicateur de démontage "propre" du volume (vérification automatique)

- HFS Plus

numérotation des blocs sur 32 bits => diminution du quantum

taille des parties de fichier portées à  $2^{63}$  octets

groupage des extensions par 8 au lieu de 3

noms sur 255 caractères Unicode (16 bits)

ouverture vers des parties nommées de fichier (nouvel arbre B\*)



# NTFS (1)

- Tout est fichier (sauf 1er secteur de partition)
- Formatage = création de fichiers prédéfinis
  - fichier des descripteurs de fichiers ou *Master File Table* (MFT)
    - contient son propre descripteur,
    - localisé depuis 1er secteur
    - dupliqué pour raisons de sécurité
  - fichier du volume, nom de volume
  - fichier bitmap, état d'allocation du volume
  - répertoire racine du volume
  - fichier journal pour garantir la fiabilité de la structure

# NTFS (2)

## ● Descripteurs de fichiers

taille fixée à la création du volume, entre 1 et 4 Ko

constitué d'attributs résidents ou non résidents

- nom de l'objet, 255 caractères Unicode
- informations de base habituelles (dates,...)
- informations de protection (liste de contrôle d'accès)
- le ou les contenus des fichiers ou répertoires

plusieurs attributs "nom" (liens physiques)

si fichier petit, tous les attributs dans le descripteur

attributs non résidents

- allocation par zones en nombres et tailles quelconques
- numéros de clusters sur 64 bits

plusieurs contenus distincts pour un même fichier (attributs nommés)

# NTFS (3)

- Les répertoires

contenu = arbre B+

- nom de l'objet
- numéro de l'objet dans la MFT
- informations de base: dates
- taille de l'objet
- numéro dans la MFT du répertoire parent

les trois dernières informations dupliquées depuis la MFT

- accès rapide sans avoir à accéder au descripteur
- mais, cohérence de données dupliquées

table bitmap associée définissant les entrées libres du répertoire

# NTFS (4)

- Compression de données

- indicateur dans le descripteur de fichier

- compression - décompression implicite

- par groupe de 16 clusters

- Sécurité par fichier journal

- opérations de structures concernent plusieurs secteurs disques

- en général, ordre des modifications minimisant les risques d'incohérence

- problème aggravé par l'écriture paresseuse

- prise en compte dans NTFS sous forme de transaction, avec fichier journal

- début transaction, modification des tampons, fin de transaction

- mémorisation la liste des pages non encore écrites

- à la reprise après panne, parcours du fichier journal => refaire ou défaire

# ext2fs de Linux (1)

- Représentation de l'espace

- allocation par bloc de taille fixe, à plusieurs niveaux (type Unix)

- descripteurs d'objets regroupés dans une table indicée par le i-nœud

- tables bitmap pour les blocs et pour les i-nœud

- partition découpée en groupes de même taille

- superbloc = informations sur la structure du volume
    - liste des descripteurs de groupe (localisation des tables des groupes)
    - table bitmap d'état d'allocation des blocs du groupe (1 bloc)
    - table bitmap d'état d'allocation des i-nœud du groupe (1 bloc)
    - table des i-nœuds du groupe
    - blocs de données

# ext2fs de Linux (2)

- Conséquences des groupes

taille d'un groupe limité car table bitmap du groupe limitée à 1 bloc

grosse partition ( 4 To) => beaucoup de groupes

allocation des i-nœuds de préférence dans le groupe du répertoire

allocation des blocs de préférence

- après le dernier alloué,
- dans un voisinage  $\pm 32$ ,
- dans le groupe ou ailleurs

- Les répertoires

couples <i-nœud, nom>, nom limité à 255 caractères

liens physiques

liens symboliques => contenu du i-nœud est le chemin d'accès à l'objet

# ext2fs de Linux (3)

- Sécurité

- duplication des super blocs dans les groupes

- écriture paresseuse des blocs

- attribut "synchrone" d'un fichier

- impose l'écriture synchrone des blocs pointeurs et du i-nœud

- attribut "secret" d'un fichier:

- à la destruction, remplissage des blocs de données aléatoires

- attribut "récupération": permettre la récupération du fichier détruit

# Structuration en couches

- Couche haute, proche application
  - implante les opérations nécessaires aux programmes d'application
  - propose un *système de fichier virtuel* (SFV) aux programmes
- Couche intermédiaire (lié au SGF)
  - assure l'interface entre le SFV et un SGF particulier
  - implante les opérations nécessaires au SFV sur le SGF
  - utilise des tampons mémoire (le cache)
- Gestionnaire de cache mémoire
  - lance les opérations nécessaires vers un disque abstrait
- Pilotes de disques
  - implantent les opérations du disque abstrait en terme d'opérations concrètes