

## Chapitre 4 – Utilisation des données élémentaires

<b>I. INTRODUCTION.....</b>	<b>1</b>
A. NOTION DE FICHIER.....	1
B. FICHIER AU FORMAT PROPRIETAIRE.....	2
C. FICHIER AU FORMAT OUVERT.....	2
D. EXTENSIONS DE FICHIERS.....	3
<b>II. CODAGE DES IMAGES.....</b>	<b>3</b>
A. CODAGE DES COULEURS RVB.....	3
B. IMAGES BITMAP.....	3
C. IMAGES VECTORIELLES.....	4
D. SEQUENCES VIDEO.....	5
<b>III. CODAGE DU SON.....</b>	<b>5</b>
A. PRINCIPES.....	5
<b>IV. COMPRESSION DES DONNEES.....</b>	<b>5</b>
A. PRINCIPES.....	5
B. COMPRESSION AVEC PERTE D'INFORMATION.....	6
C. COMPRESSION SANS PERTE D'INFORMATION.....	6
<b>V. CHIFFREMENT ET AUTHENTIFICATION DES DONNEES TRANSMISES.....</b>	<b>7</b>
A. CHIFFREMENT.....	7
B. AUTHENTIFICATION : SIGNATURE NUMERIQUE.....	7
<b>VI. CODES DETECTEURS ET CORRECTEURS D'ERREURS.....</b>	<b>8</b>
A. PARITE (OU IMPARITE).....	8
B. SOMMES DE CONTROLE ET CRC.....	9
C. CODE DE HAMMING.....	9
<b>VII. NOTION DE POLICES DE CARACTERES.....</b>	<b>10</b>
A. POLICES UNIVERSELLES.....	10
B. POLICES TRUE TYPE.....	10

### **I. Introduction**

---

Nous avons étudié les systèmes de codage des données élémentaires au sein de l'ordinateur. Ces données sont rarement utilisées isolément, mais regroupées pour former des ensembles utilisables dans des applications informatiques.

#### **A. Notion de fichier**

Un fichier regroupe un ensemble de données codées selon une ou plusieurs techniques. Les données qu'il contient sont relatives à un domaine d'application particulier :

- Fichier des fiches clients : champs numéro de client, raison sociale, chiffre d'affaires, etc.
- Fichier son : champs fréquence d'échantillonnage, valeurs échantillonnées, etc.
- Document HTML, etc. : texte brut (pas de champs défini dans le codage, mais structure en clair)

## Thème 1 – REPRESENTATION DE L'INFORMATION

Selon la manière dont sont codés les contenus de ces fichiers, on peut déterminer 2 grandes familles :

- les fichiers au format PROPRIETAIRE
- les fichiers au format OUVERT

### B. Fichier au format propriétaire

Dans ce type de fichier, chaque donnée a été codée selon son type :

- Un entier naturel : en binaire, sur 1, 2 ou 4 octets suivant sa grandeur
- Un entier relatif : en binaire, mais avec l'utilisation de la technique de complément à 2 si sa valeur est négative
- Un nombre flottant : en utilisant le format IEEE-754
- Un caractère : en utilisant une des tables de codification des caractères

Le décodage des données d'un tel fichier nécessite la connaissance exacte du codage de chacune de ses données : c'est ce qu'on appelle une description ou un format d'enregistrement.

Seule la personne connaissant cette description (le propriétaire du fichier) peut décoder ce fichier, à moins qu'il ait communiqué cette description à d'autres.

**Les formats de fichiers propriétaire** utilisent :

- Plusieurs systèmes de codage des données, en fonction de leur type
- Une description précise de l'utilisation de chacune des données, externe au fichier, non connue (sauf si elle est publiée)

Nous avons utilisé une description précise pour décoder un fichier image BMP, par exemple, ou une autre pour décoder des annonces de l'ANPE.

Le contenu d'un fichier formaté NE PEUT PAS être décodé en utilisant un éditeur de texte (à moins qu'il ne soit composé que de caractères...) et nécessite de toute manière des informations complémentaires pour être exploité (une description de chacun des champs).

Les fichiers les plus utilisés jusqu'à présent sont codés de cette manière.

### C. Fichier au format ouvert

Au contraire des fichiers formatés, ce type de fichier n'utilise qu'un seul type de codification pour coder les données : il s'agit d'une des tables de codification des caractères.

De plus, les données stockées dans ces fichiers sont délimitées par des séparateurs qui permettent de d'identifier clairement les données et la structure générale du contenu.

**Les formats de fichiers ouverts** utilisent :

- Un système de codage interne unique de type caractère
- Un système de balisage des données permettant de définir la structure complète du document
- Les **documents XML** représentent l'évolution actuelle des formats ouverts

Le contenu d'un fichier au format ouvert peut être décodé en utilisant un éditeur de texte (à condition que le jeu de caractère utilisé soit connu...).

La famille de fichiers issue de XML (et de son père SGML) représente l'évolution majeure en terme d'échange de données informatisé.

### D. Extensions de fichiers

Afin d'identifier facilement les fichiers d'après leur nom, un système d'extension de fichier a été mis en place : par exemple, un fichier *document.doc* sera identifié comme un document Microsoft Word.

L'extension associée au nom de fichier permet d'associer le contenu d'un fichier (et l'organisation de ce contenu) à une application qui pourra l'exploiter.

## II. Codage des images

---

Toute image affichée sur un écran ou imprimée sur une imprimante est constituée de pixels (*Picture Elements*). A chaque pixel est associé un code couleur.

Différentes organisations du contenu des fichiers images ont été inventées : il existe cependant 2 modes de définition d'images : les images bitmaps et les images vectorielles.

### A. Codage des couleurs RVB

Le codage **RVB** (Rouge Vert Bleu, *en anglais RGB, Red Green Blue*) est celui mis en œuvre sur la plupart des matériels informatique.

Ce codage indique pour chaque couleur primaire, sur 1 octet, le niveau de l'intensité de la couleur, de 0 = absence de la couleur à 255 = intensité maximale.

Le codage des couleurs d'un pixel nécessite généralement 3 octets utiles (24 bits), plus 1 octet (souvent inutilisé, parfois utilisé pour coder la transparence).

### B. Images Bitmap

Dans le mode bitmap, chaque point de l'image est mémorisé. L'image devra être utilisée dans la dimension dans laquelle elle a été créée, sinon un effet d'escalier sera inévitable.

Une image **BITMAP** est représentée par un tableau de points, chaque point représentant un pixel de l'image

La profondeur des couleurs d'une image est déterminée par le nombre de bits destinés à coder la couleur de chaque pixel :

- 1 bit : permet de coder 2 couleurs : 0 = noir, 1 = blanc
- 8 bits : permet de coder 256 couleurs
- 24 bits (3 octets) : 16 millions de couleurs

Inconvénient :

- occupe un volume important (tous les pixels sont représentés)
- perte de qualité d'image en cas d'agrandissement

Espace occupé par une image codée en 16 millions de couleurs :

	256 couleurs (1 octet)	16 millions de couleurs (3 octets)
640 X 480	300 ko	900 ko
1024 X 768	800 ko	Plus de 2 Mo

Lorsqu'on veut représenter une image en mémoire, on est donc face à un choix :

- soit privilégier le nombre de pixels (640x480 ou 1024x768), c'est à dire sa définition
- ou au contraire le nombre de couleurs utilisées (256 ou 16,5 millions)

La plupart du temps, un meilleur résultat sera obtenu en privilégiant le nombre de couleurs.

### AMELIORATION DU SYSTEME DE CODAGE DES COULEURS EN BITMAP :

On détermine une palette de 256 couleurs :

- un tableau de codage de 256 couleurs importantes ;
- chaque case du tableau occupe 3 octets : 1 octet pour coder le rouge, 1 octet pour coder le Vert et un octet pour coder le bleu.

Chaque pixel est ensuite codé sur un octet, cet octet donnant l'indice de la couleur dans la palette.

On réduit ainsi de manière très importante l'espace occupé par l'image en mémoire ou sur disque (on perdra par contre en qualité d'image)

Exemple de formats bitmap : BMP, GIF, JPEG

### C. Images vectorielles

Les images vectorielles sont représentées par les équations mathématiques des portions de droites et de courbes qui la constituent.

Un cercle, par exemple, sera déterminé complètement par les coordonnées du centre et la valeur du rayon avec éventuellement la couleur du trait et l'épaisseur de ce trait, la couleur de remplissage, etc.

L'intérêt de cette méthode est que l'on peut modifier la taille du dessin sans en altérer la définition et les proportions. De plus si le dessin est composé de plusieurs objets, ils pourront être manipulés séparément.

Une image au **format vectoriel** est décrite par un ensemble de fonctions et de coordonnées associées à ces fonctions (ligne, rectangle, polygone, cercle, etc.)

Avantages :

- Le rendu final est lisse et dépend uniquement de la résolution du périphérique.
- Les images peuvent être agrandies sans perte de qualité.

Exemple de formats vectoriels :

- SVG : Scalable Vector Graphics
- WMF : Windows Meta File

### **D. Séquences vidéo**

Le codage de la vidéo va nécessiter un codage d'images, plusieurs fois par seconde (25 fois, en général), auquel il faut ajouter le codage du son.

Afin de réduire les volumes liés aux vidéos, une technique utilisée est la suivante :

- Enregistrement d'une image
- Enregistrement des changements pour les images suivantes : si les changements sont trop importants, on enregistre à nouveau l'image complète (cas du changement de plan de prise de vue).

## **III. Codage du son**

---

### **A. Principes**

Un son est un signal analogique, c'est à dire que le signal peut prendre une infinité de valeur au cours du temps. Pour pouvoir stocker ce son dans un ordinateur, il va falloir l'échantillonner, c'est-à-dire en capturer des valeurs à des intervalles déterminés (= à une fréquence donnée).

Pendant cette opération, on va forcément perdre de l'information. Afin que cette perte soit le moins sensible possible, on s'attachera à choisir une fréquence d'échantillonnage très grande (beaucoup de valeurs pour une courte durée).

Pour un CD audio, la fréquence d'échantillonnage est de 44 100 Hz, c'est à dire qu'une seconde de musique est représentée par 44 100 valeurs différentes !

Chacune de ces valeurs sera alors codée en binaire sur 8 ou 16 bits selon la qualité désirée (radio ou CD), ce qui signifie qu'un son enregistré en stéréo sur 16 bits va nécessiter pas moins de 172 KO par seconde (il faudra 30 méga pour un morceau de musique de 3 minutes !)

## **IV. Compression des données**

---

### **A. Principes**

Le stockage du son, de l'image ou plus généralement des fichiers binaires posent des problèmes de place mémoire et de temps de transmission : c'est pourquoi, avec l'ère du multimédia, les techniques de compression sont d'actualité.

La **compression** est une **technique consistant à réduire la taille des données** d'un fichier en lui appliquant un algorithme spécialisé : l'objectif étant

- La réduction de l'occupation mémoire
- La réduction des temps de transmission

Les techniques utilisent des méthodes différentes selon les cas :

- un simple codage des séquences répétitives sous la forme [valeur-nb d'occurrence],
- des méthodes plus subtiles qui font intervenir un dictionnaire des séquences les plus utilisées et leur remplacement par des codes binaires,
- des calculs qui permettent la suppression de certaines données

Il en existe 2 grandes catégories : la compression avec perte d'informations et la compression sans perte d'informations.

## **B. Compression avec perte d'information**

Certains types de fichiers peuvent se passer d'une restitution parfaite après compression : il s'agit des fichiers Image et son, l'œil ne percevant pas toutes les nuances de couleurs, l'oreille n'entendant pas toutes les fréquences de son.

La **compression AVEC perte d'information** permet de réduire le volume de données des fichiers de type image et son, essentiellement

Exemple :

- JPEG (Joint Photographic Expert Group) pour les images fixes
- MPEG (Moving Picture Expert Group) pour la vidéo.

Pour la vidéo, les techniques utilisées englobent le découpage d'images en blocs, mais aussi la prédiction de mouvements pour déterminer les changements à opérer d'une image à l'autre.

## **C. Compression sans perte d'information**

Certains fichiers, par contre, ne supporte aucune perte ou altération de donnée : il s'agit des fichiers de données, des fichiers binaires d'exécutables, etc.

Les outils de compression devront restituer l'état original du fichier après décompression d'un fichier compressé.

La **compression SANS perte d'information** permet de réduire le volume de données des fichiers de type données et fichiers exécutables, et **restitue le fichier tel qu'il était avant compression**

Exemple de formats ou outils de compression :

- logiciels de compression et formats : ARJ, PKZIP, Winzip, 7Zip, RAR, etc.
- DRIVESPACE intégré à MS-DOS.

## **V. Chiffrement et authentification des données transmises**

Avec l'accroissement des échanges de données entre systèmes hétérogènes, et l'extension des formats de fichiers ouverts, des problèmes de confidentialité et d'authentification des données transmises ont été soulevés et ont trouvé des solutions dans le chiffrement et la signature numérique..

### **A. Chiffrement**

Le **chiffrement** (*anglais : encryption*) est le procédé grâce auquel on peut **rendre la compréhension d'un document impossible à toute personne qui ne possède par la clé** de (dé)chiffrement.

Les pirates informatiques vont utiliser des techniques afin de décrypter ces messages (on parle alors de techniques de cryptanalyse).

Le principe du chiffrement repose sur l'utilisation de clef permettant le chiffrement et le déchiffrement.

Un **système de chiffrement** est dit :

- **symétrique** : la même clef est utilisée pour chiffrer et déchiffrer.
- **asymétrique** : une paire de clefs est nécessaire, une **clé publique**, servant au chiffrement, et une **clé privée**, servant à déchiffrer.

Les méthodes les plus connues sont :

- DES (Data Encryption Standard), le Triple DES (ou 3DES) : symétrique
- AES (Advanced Encryption Standard) : symétrique,
- RSA : asymétrique
- Blowfish : symétrique

Les solutions de chiffrement sont utilisées aussi bien pour les transmission d'informations sur les réseaux que sur le poste de travail : chiffrement des fichiers sur les disques durs et autres supports, ou bien chiffrement complet des supports de stockage.

### **B. Authentification : signature numérique**

En complément du chiffrement des messages, les techniques de signatures des messages permettent de garantir l'authenticité d'un document électronique (comme on garanti un document papier avec une signature manuscrite).

La **signature numérique** est un **mécanisme permettant d'authentifier** l'auteur d'un document électronique et de garantir son intégrité, par analogie avec la signature manuscrite d'un document papier.

Ce mécanisme est basé sur une **fonction de hachage** et permet l'**obtention d'un résumé du message** transmit (ou condensat ou en *anglais digest*).

Ce résumé est construit et envoyé avec le message, puis construit à la réception et comparé au premier afin de s'assurer qu'ils sont identiques.

Les algorithmes les plus utilisés actuellement sont :

- MD5 (Message Digest 5) : utilisé pour vérifier l'authenticité des fichiers téléchargés
- SHA (Secure Hash Algorithm)

## **VI. Codes détecteurs et correcteurs d'erreurs**

Lors de son stockage en mémoire ou de sa transmission, une données numérique peut subir une altération (parasites ou 'bruits' lors d'une transmission).

Des méthodes de détection d'erreurs qui permettent de détecter si une information a été altérée : c'est ce qu'on appelle des codes détecteurs d'erreurs.

Il existe également des codes beaucoup plus complexes qui permettent de savoir quel bit exactement a été modifié, ces codes sont appelés codes correcteurs d'erreurs.

Les **codes détecteurs** d'erreurs permettent la **détection** d'une erreur

Les **codes correcteurs** d'erreurs peuvent **corriger** certaines erreurs simples.

On parle de **contrôles par redondance** car ils nécessitent l'ajout de données au message transmis

### **A. Parité (ou imparité)**

L'un de ces codes est obtenu très facilement en collant un bit (dit 'bit de parité' ou 'bit d'imparité') à chaque caractère transmis ou stocké. Il existe 2 techniques dites « à parité paire » ou « à parité impaire ».

Le **bit de parité** est un bit qui est accolé à un ensemble de bits formant un caractère ou un message :

- dans le cas de la **parité paire**, ce bit est positionné de telle manière que le **nombre de bits total soit pair**
- dans le cas de la **parité impaire** (imparité), ce bit est positionné de telle manière que le **nombre de bits total soit impair**



## Thème 1 – REPRESENTATION DE L'INFORMATION

Exemple cas de la **parité paire** :

- Code ASCII sur 7 bits **0010101** : 3 bits sont à 1, j'ajoute 1 bit à 1 pour obtenir un nombre total de bits pair : **10010101**
- Code ASCII sur 7 bits **0010111** : 4 bits sont à 1, j'ajoute 1 bit à 0 pour conserver un nombre total de bits pair : **00010111**

Exemple cas de la **parité impaire** :

- Code ASCII sur 7 bits **0010101** : 3 bits sont à 1, j'ajoute 1 bit à 0 pour conserver un nombre de bits total impair : **00010101**
- Code ASCII sur 7 bits **0010111** : 4 bits sont à 1, j'ajoute 1 bit à 1 pour obtenir un nombre total de bits impair : **10010111**

Lors d'un contrôle, après une transmission par exemple, le système vérifiera que le nombre de bits à 1 est toujours pair (en cas de parité paire) :

- si ce n'est pas le cas, l'information a été altérée,
- si ce nombre est pair, c'est que tout c'est bien passé (normalement...)

Ce système de contrôle est très simpliste : si 2 bits viennent à être modifiés, l'erreur n'est pas détectée.

### B. Sommes de contrôle et CRC

La **somme de contrôle** (anglais Checksum) et le **CRC** (Contrôle de Redondance Cyclique) regroupent des techniques permettant de détecter des erreurs de transmission en ajoutant un nombre calculé par un algorithme sur le message transmis.

### C. Code de Hamming

Le code de Hamming est un code **correcteur** d'erreurs. Il sert donc à la détection et à la correction des erreurs lors de communications numériques.

## VII. Notion de polices de caractères

---

Les **polices de caractères** représentent un **style de dessin** appliqué aux caractères alphanumériques et symboles (on peut parler de glyphe) affichables ou imprimables.

### A. Polices Universelles

Ces polices sont codées dans un format bitmap. Leur redimensionnement provoque une perte de qualité.

Ces polices sont enregistrées dans des fichiers donc l'extension est .FON.

### B. Polices True Type

Les **polices True Type** sont codées dans un **format vectoriel** : elles peuvent donc être redimensionnées sans déformation et leur rendu est identique à l'écran comme à l'impression.

Les polices True Type sont enregistrées dans des fichiers donc l'extension est .TTF.