

CODER LES COULEURS ET LES IMAGES

Voici une partie du codage d'une image (Lena.bmp) :

```
01000010 01001101 00110110 00000000 00001100 00000000 00000000 00000000
00000000 00000000 00110110 00000000 00000000 00000000 00101000 00000000
00000000 00000000 00000000 00000010 00000000 00000000 00000000 00000010
00000000 00000000 00000001 00000000 00011000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 01110101 00001010
00000000 00000000 01110101 00001010 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00111001 00010110
01010010 00111001 00010110 01010010 00111110 00100000 01100000 00111110
00011100 01011101 01000001 00011110 01100001 00111001 00010101 01011110
00111110 00011010 01011100 00111100 00011100 01011101 01000100 00100100
01100010 01000010 00011011 01011110 00111010 00010101 01011001 01000010
00011001 01011101 01001011 00011011 01011111 01000101 00011100 01100000
01010111 00111000 01111001 01110100 01011000 10010011 10000000 10000001
10110111 10011101 10110001 11010001 10101010 10111100 11011010 10110001
10111101 11011001 10101111 10111101 11011011 10110011 11000000 .....
```

Problème : comment restituer l'image à partir de ce fichier !

En fait ce fichier commence par un entête permettant de savoir comment traiter le fichier. Le début de l'entête indique le format du fichier (ici BMP) c'est la signature du format (sous Windows c'est **BM**). Ceci permet de savoir comment l'image a été codée. En jaune (octets 11, 12, 13 et 14) l'offset : permet de savoir le début du corps de l'image : ici cela donne 54, ce qui mène aux octets en bleu.

Dans le format BMP une image est vue comme une carte de points : c'est le sens de "bitmap". Chaque "point" de l'image, appelé pixel, est décrit par sa couleur. On dit que l'on a une image matricielle (une matrice en maths est un tableau de nombres)

Par exemple, pour une image en noir et blanc, "00110100" signifierait que le premier pixel est noir, le second aussi, le troisième est blanc etc....

Encore faut-il savoir combien il y a de pixels dans chaque ligne et dans chaque colonne ! L'image précédente pourrait correspondre à :

```
00  ou  0011
11      0100
01
00  par exemple !
```

D'où l'intérêt de l'entête : on y lit la taille de l'image, c'est-à-dire sa largeur et sa hauteur, ce qui permet de retrouver la place de chaque pixel.

On y lit aussi le nombre de couleurs possibles pour un pixel.

- Un pixel en noir et blanc peut être décrit par un seul bit (0 ou 1).
- Un pixel en niveau de gris est généralement décrit par un octet (256 niveaux de gris de 0 à 255).

Le fichier précédent représenterait alors un seul pixel avec un niveau de gris égal à $0+0*2+1*4+0*8+1*16+1*32 = 52$: plus ce nombre est bas, plus le gris est foncé (car on approche de 0, et quand on ne projette aucune couleur on a un écran noir).

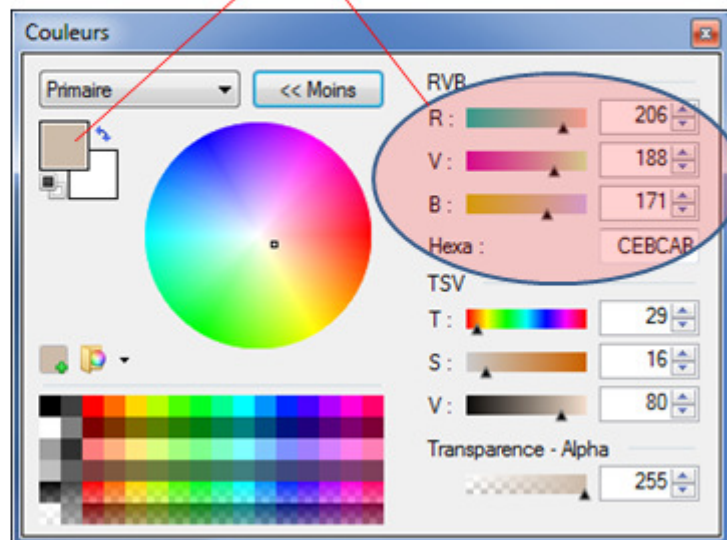
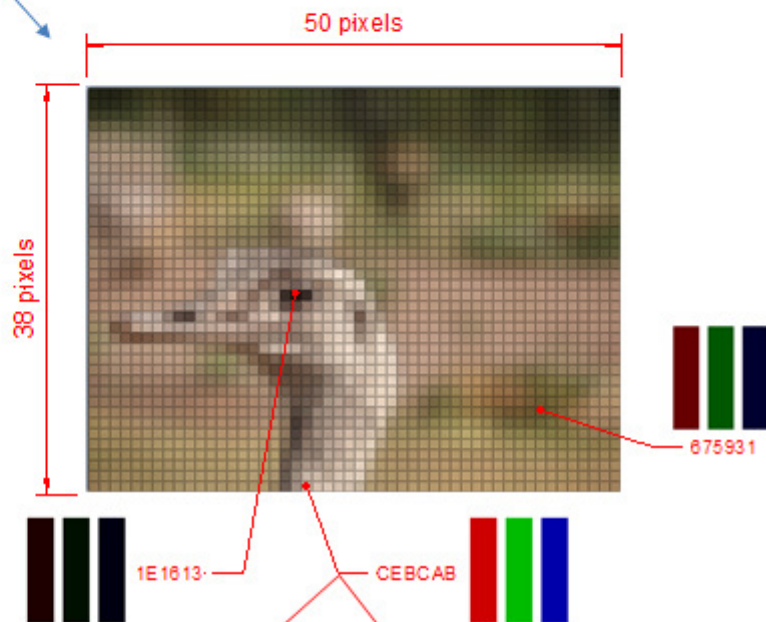
- Un pixel en "True color" est codé sur trois octets : l'un donne le niveau de rouge (de 0 à 255), l'un le niveau de vert (de 0 à 255) et le dernier le niveau de bleu (de 0 à 255). C'est le codage RVB (ou RGB en anglais). La couleur sera rouge si le niveau de vert et de bleu sont nuls, par exemple (240,0,0), le premier entier n'étant pas trop faible, sinon cela ressemblera plus à du noir ! Plus ce premier nombre est élevé, plus la couleur est claire. Noir correspond à (0, 0 , 0) et les gris à (n,n,n) où n est un entier entre 1 et 254. Par exemple (120, 120, 120) est un gris moyen. Vous pouvez utiliser "La boîte à couleur" pour visualiser les couleurs suivant leur code RGB.

Autruche1.jpg



Echelle 1 :1

Echelle 10 :1



Evidemment, ce format est très volumineux.

Pour diminuer le poids des images matricielles on utilise la compression.

Exemple : si vous devez dicter "AAAAAAHHHHHHHAA" vous allez dire "7A8H2A"... Vous avez compressé votre chaîne de caractères ... Vous avez même fait une compression sans perte, car à partir de "7A8H2A" vous pouvez très bien retrouver la chaîne de départ !

La compression en archives ZIP est aussi une compression sans perte pour tous types de fichiers.

Il existe d'autres types de compressions, appelées compressions avec perte : on comprend bien que si on change la couleur d'un seul pixel d'une image, on ne verra pas la différence avec l'image originelle (si l'image est assez grande). La compression utilisée par le format JPEG est de ce type.

Bien sûr, le type de compression est indiqué dans l'entête pour permettre la décompression !

Voici différents formats d'images matricielles:

- **bmp** : Format originel qui applique une couleur à chaque pixel (codée sur 1bit, 1 octet ou 3 octets). En général sans compression. Format très volumineux non adapté à internet.
 $2^{24} \approx 16$ millions de couleurs possibles.
- **gif** : utilise une palette de 256 couleurs (choisies selon l'image) et n'est plus tellement utilisé sauf pour des gifs animés. Au départ il y a 16 millions de couleurs comme pour le BMP, mais les 256 plus utilisées sont repérées et numérotées de 0 à 255. On attribue alors à chaque pixel la couleur de la palette la plus proche. Un seul octet suffit alors pour coder chaque pixel. Format assez léger à cause du nombre réduit de couleurs, mais peu adapté à la photographie.
- **jpg** : Idéal pour les photographies mais il faut éviter les sauvegardes consécutives avec compression, car la compression se fait avec perte. Le niveau de compression est choisi par l'utilisateur.
- **png** : Idéal pour des logos ou des icônes. Gère la transparence. Remplace maintenant souvent les gifs.

svg : Images vectérielles : qui décrit les formes géométriques présentes au lieu de donner une couleur à chaque pixel comme les images matricielles . Idéal pour des dessins géométriques (plans, logos). La taille de ce type d'images peut être modifiée sans aucune perte de qualité, ce qui n'est pas le cas pour une image matricielle (pixelisation).



image extraite du site :
<http://www.mysti2d.net/>

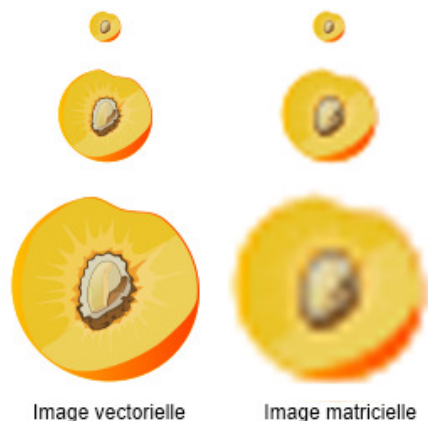


Image vectorielle

Image matricielle

Une image vectorielle est redimensionnable sans perte de qualité contrairement à une image matricielle.

image extraite du site : <http://www.mysti2d.net/>

Résumé des différents formats :

	Type (matriciel/ vectoriel)	Compression des données	Nombre de couleurs supportées	Animation	Transparence
JPEG	matriciel	Oui, réglable (avec perte)	16 millions	Non	Non
GIF	matriciel	Oui, Sans perte	256 maxi (palette)	Oui	Oui
PNG	matriciel	Oui, sans perte	Palettisé (256 couleurs ou moins) ou 16 millions	Non	Oui
SVG	vectoriel	compression possible	16 millions	Oui	Oui

Poids de cette image pour différent formats:

- BMP : 49 242 octets
- JPG : 2 873 octets
- PNG : 398 octets
- SVG : 249 octets

