



FORMAT WAV

A - Introduction

WAV, contraction de **WAVEform audio format**, est un standard de Microsoft et IBM pour stocker l'audio numérique. Le format WAV est une extension de fichiers audio, il s'agit d'un conteneur capable de recevoir des formats aussi variés que le MP3, le WMA, l'ATRAC, le PCM. C'est ce dernier qui est cependant le plus courant.

Le format PCM : C'est le format de fichier "standard" pour les samples (les enregistrements de sons), car les données sont brutes, c'est-à-dire qu'elles ne sont ni modifiées, ni compressées. Le fichier possède une en-tête de 44 octets (voir ci-dessous), permettant de connaître le type du sample : Son format, sa fréquence, le nombre de voies, etc..

B - Structure du fichier WAV

Un fichier WAV est divisé en trois blocs : un bloc déclarant que le fichier est au format WAV, un autre qui décrit le format utilisé et le dernier contenant les données (le son). Cet entête dispose de 44 octets.

Le tableau ci-dessous présente en détaille la structure du fichier WAV :

Nom	Nombre d'octets	Type	Endian	Description
Bloc de déclaration d'un fichier au format WAVE				
Chunk ID	4	4 char	big	4 caractères « RIFF » codé en ASCII, de valeur hexa : 0x52494646
Chunk Size	4	int 32	little	Taille du fichier - 8 octets
Format	4	4 char	big	4 caractères « WAVE » codé en ASCII : 0x57415645
Bloc décrivant le format audio				
SubChunk 1 ID	4	4 char	big	4 caractères « fmt » codé ASCII : 0x666d7420
SubChunk 1 Size	4	int 32	little	Taille du sous-bloc "SubCHunk 1" - 8 octets, vaut 16 pour le PCM
Audio Format	2	int 16	little	Type de compression audio, 1 pour PCM
Num Channels	2	int 16	little	Nombre de canaux (1 = Mono, 2 = Stéréo, ...)
Sample Rate (Fréquence)	4	int 32	little	Fréquence d'échantillonnage (nombre d'échantillons par seconde)
Byte Rate (BytePerSec)	4	int 32	little	Nombre d'octets par seconde = Fréquence * BitsParCanal/8 * NbCanaux
Block Align (BytePerSample)	2	int 16	little	Nombre d'octets par échantillon = NbCanaux * NbOctetsParCanal
BitsPerChannel	2	int 16	little	Nombre de bits par canal
Bloc des données				
SubChunk 2 ID	4	4 char	big	4 caractères « data » codé ASCII : 0x64617461
SubChunk 2 Size	4	int 32	little	Taille des données = NbCanaux * NbSamples * BitsParCanal/8 = taille du fichier - 44 octets (Taille de l'entête !)
Datas (Données)	FileSize - 44	...	little	Les données

Remarque : Qu'est-ce que l'endian ?

L'endian est la manière de représenter les données en binaire : ici, nous ne nous occuperons que de big Endian et de little Endian pour des entiers.

Un entier peut être représenté sur plusieurs octets s'il est supérieur à 255. Il peut être représenté sur 2, 4, 8 ou 16 octets, on parle d'entiers 8 bits, 16 bits, 32 bits, ...

Bits	Octets	Valeurs représentées (entier positif)
8	1	0 à $2^8 - 1$ [0,255]
16	2	[0, $2^{16} - 1$] [0,65535]
32	4	[0, $2^{32} - 1$] [0,4294967295]

Dans le format .wav, la plupart des entiers sont codés sur 4 ou 2 octets.

L'endian caractérise l'ordre dans lequel sont enchaînés les octets, en big-endian, l'octet de poids fort est en premier ; en little endian, c'est l'inverse.

Prenons le nombre 43262 en décimal égale à 0xA8FE en hexadécimal.

Enregistré en **big endian** sur 4 octets, il apparaîtrait comme cela dans le fichier : **00 00 A8 FE**

Enregistré en **little endian** sur 4 octets, il apparaîtrait comme cela dans le fichier : **FE A8 00 00**

C - Exemple

Ci-dessous l'exemple d'un fichier WAV ouvert à l'aide d'un logiciel éditeur hexadécimal (Hexedit)

