ORAL HTI - UTF & LES FORMES D'ENCODAGE - 02 FEVRIER

1a°) DÉFINITION _ Chloé _ Slide 1 à 4

Le format d'encodage est un traducteur de code qui transforme du binaire en caractère lisible.

Pourquoi faut-il des formats d'encodage ?
En informatique les données sont généralement sous forme binaire elle représente de la musique du texte des vidéos des caractères.
C'est la méthode que l'on utilise pour interpréter 0 et 1 en texte.

Comment fait-on alors pour écrire du texte ? La réponse est toute bête : on associe à chaque caractère (une lettre, un signe de ponctuation, une espace…) un nombre. Un texte est alors une suite de ces nombres, on parle de chaîne de caractères.

Cet exemple nous montre comment les informaticiens inventent une façon de coder un texte en mémoire.
Premièrement, on décide de l'ensemble des caractères dont on a besoin, et on assigne à chacun un identifiant numérique unique appelé code.
Cet ensemble est appelé jeu de caractères codés.

Ensuite, il faut déterminer l'encodage, c'est-à-dire la façon de transcrire un texte grâce aux codes des caractères qui le composent, selon un jeu de caractères donné. Le moyen le plus simple est d'écrire directement chaque code (auquel cas on parle de page de code — charmap) ; le jeu et l'encodage sont alors confondus.

1b°) PRES DIFFERENTS FORMATS _ Yasmine _ Slide 5

Au fil du temps de nombreux formats ont étés crées pour stocker de pus en plus de caractères.

Historique :

Code Baudot: 1874 Plus ancien que l'SCII, c'est 1 code binaire codé pour 1 série de 5 bit.

ASCII : (American Standard Code for Information Interchange), norme américaine, inventé en 1961. Codé sur 7 bits. Il dispose de 128 caractères. Ils sont suffisants pour mémoriser notre alphabet, les chiffres, les ponctuations mais pas les caractères spéciaux (cédilles , accents).

- ISO 8859 : 1998 8bits pour les langues latines. 8 bits donc 256 caractères max : les caractères spéciaux sont disponibles.
- ISO 8859-15 : 1998 ajoute l'**€**.
- ISO 2022 : multi octet pour les langues asiatiques.
- Windows 1252 : 1990 propre aux Windows.

- Mac Roman : 2000 propre aux Mac, codé en 8 bits.
- UNICODE : 1991, permet les échanges de texte dans différents languages
- UTF-16 : 16 bits, les caractères ne font pas tous la meme taille.
- UTF-8 : les caractère sont encore plus variables, compatible avec ASCII. Actuellement le plus utilisé.

2a°) UTF _ Joe _ Slide 7

Universal Character Set Transformation Format soit Format de Transformation des caractères universel.
Il s'agit d'un codage de caractères informatiques conçu pour coder l'ensemble des caractères du répertoire universel. Initialement développé par l'**ISO** (Organisation Internationale de Normalisation) dans la norme internationale ISO/CEI 10646

Aujourd'hui totalement compatible avec le standard **Unicode**:
un standard informatique qui permet des échanges de textes dans différentes langues, à un niveau mondial. Également compatible avec la norme **ASCII** limitée à l'anglais de base.

2b°) UTF8 & UTF16 _ Joe _ Slide8

UTF-8 Joe

Comme son nom l'indique, l'encodage UTF-8 ne peut excéder 8 bits. Techniquement, il s'agit de coder les caractères Unicode sous forme de séquences d'un octet chacune. La norme Unicode définit entre autres un ensemble ou répertoire de caractères. Chaque caractère est repéré dans cet ensemble par un index entier aussi appelé « point de code ». Par exemple le caractère « € » est le 8365e caractère du répertoire Unicode, son index, ou point de code, est donc 8364.

La principale caractéristique d'UTF-8 est qu'elle est rétro-compatible avec la norme ASCII, c'est-à-dire que tout caractère ASCII se code en UTF-8 sous forme d'un unique octet, identique au code ASCII. Par exemple « A » (A majuscule) a pour code ASCII 65 et se code en UTF-8 par l'octet 65.

Le répertoire Unicode peut contenir plus d'un million de caractères, ce qui est bien trop grand pour être codé par un seul octet (limité à des valeurs entre 0 et 255). La norme Unicode définit donc des méthodes standardisées pour coder et stocker cet index sous forme de séquence d'octets : UTF-8 est l'une d'entre elles, avec UTF-16, UTF-32 et leurs différentes variantes.

L'UTF-8 est utilisé par 82,2 % des sites web en décembre 2014, puis 87.6 % en 2016 et près de 90.4 % en 2017.

UTF-16 Jean-Yves

L'UTF-16 est un bon compromis lorsque la place mémoire n'est pas trop restreinte, car la grande majorité des caractères Unicode assignés pour les écritures des langues modernes (dont les caractères les plus fréquemment utilisés) le sont dans le plan multilingue de base et peuvent donc être représentés sur 16 bits.

L'UTF-32 est utilisé lorsque la place mémoire n'est pas un problème et que l'on a besoin d'avoir accès à des caractères de manière directe et sans changement de taille (hiéroglyphes égyptiens).

2c°) ASCII _ Jean-Yves _ Slide 9

La norme ASCII est largement utilisée en informatique pour coder les caractères. Ce nom apparu dans les années 1960, provient de l'acronyme anglais "American Standard Code for Information Interchange" qui signifie en français "Code américain normalisé pour l'échange d'information".

ASCII définit 128 codes à 7 bits, comprenant 95 caractères imprimables : les chiffres arabes de 0 à 9, les lettres minuscules et capitales de A à Z, et des symboles mathématiques et de ponctuation. ASCII suffit pour représenter les textes en anglais, mais il est trop limité pour les autres langues, dont le français et ses lettres accentuées. Les limitations du jeu de caractères ASCII sont encore sensibles au XXè siècle, par exemple dans le choix restreint de caractères généralement offerts pour composer une adresse email. L'ASCII étendu est néanmoins codé sous 8 bits dans le but d'ajouter des caractères, tel que des caractères spéciaux et les lettres accentuée utilisée en français.

Questions/Réponses

Comment encoder avec la table ASCII ? (Principe de chiffrement)
La conversion ASCII consiste a remplacer chaque caractère par sa valeur dans la table ASCII. Les caractères n'existant pas dans la table ne peuvent pas être codés.
Exemple : dCode s'écrit 1100100 1000011 1101111 1100100 1100101 en binaire (7-bit) et 100 67 111 100 101 en décimal.

Comment décoder par table ASCII ? (Principe de déchiffrement)
Le déchiffrement consiste à remplacer chaque valeur par le caractère correspondant dans la table ASCII.
Exemple : 1100100 1000011 1101111 1100100 1100101 devient dCode.

Comment reconnaitre le chiffre ASCII ?
Le message est généralement écrit soit en binaire, soit en décimal, soit en hexadécimal (ou plus rarement en octal).
Les valeurs les plus courantes doivent correspondre aux caractères habituels tels que les lettres majuscules ou minuscules (entre 65 et 122 en décimal)

Sur combien de caractères est représenté un code ASCII ?
En binaire on utilise soit 7 bits, soit 8 bits (1 octet) pour représenter un caractère ASCII.
En octal, donc sur 1 octet, on utilise 3 caractères (de 000 à 177).
En décimal, le nombre est compris entre 1 et 128 (de 1 à 3 caratères).
En héxadecimal, on utilise généralement 2 caractères (de 00 à 7f).

Comment passer d'une lettre ASCII minuscule à une majuscule ?
Dans le code ASCII il y a une différence de 32 entre une lettre majuscule et une lettre minuscule. Il faut donc ajouter 32 au code ASCII d'une majuscule pour obtenir une minuscule et soustraire 32 au code ASCII d'une minuscule pour avoir une majuscule.

3a°) Les caractères spéciaux _ Alison _ Slide 11

Le HTML dit classique sollicite de respecter le codage des caractères ASCII (American Standard Code for Information Interchange, Code Américain Standard pour l'Echange d'Informations). Ces caractères sont codés sur une base de 7 bits, autrement dit un code composé de sept chiffres qui sont tous égaux à 0 ou à 1. En outre, la représentation de 27 de caractères différents. Ces 128 caractères sont assez pour l'alphabet, certaines ponctuations, les chiffres .. Cependant pour les caractères qui sont accentués, ils ne sont point autorisés. En effet, il n'est pas suffisant pour mémoriser certains caractères spéciaux (accents, cédilles...). Pour cela, il faut donc faire preuve d'un codage particulier.

À savoir que les fichiers HTML, sont prédéfinis pour être codés sans caractères spéciaux, ainsi donc en ASCII. Afin de remédier à cela, on utilise un code alpha numérique pour traduire les caractères spéciaux au sein du langage HTML.

Ici voici quelques exemples :

Ceci permet de prévoir les incompatibilités, par exemple entre Mac et un PC où on risque de voir certains caractères sous la forme d'un code incompréhensible.

L'avantage de l'UTF-8 est un encodage dit « unicode », il est codé sur 8 à 32 bits, autrement dit cela permet d'encoder un nombre quasi illimité de caractères quelque soit son alphabet (latin, cyrillique, asiatique...) qui est compatible sur toutes les plateformes (Windows, Mac, Unix).

3b°) Détection & Manipulation _ Mathieu _ Slide 12

Un problème posé par la diversité des encodages existants est la détermination de l'encodage utilisé par un fichier. Les renseignements associés à un fichier particulier (sa date de création par exemple) n'indiquent rien sur son encodage.
On doit donc tenter de le « deviner », au moyen d'algorithmes compliqués qui analysent le contenu du fichier. Ces algorithmes sont efficaces la plupart du temps, mais s échouer, et sont compliqués.

C'est ce qui explique les affichages bizarres de certains fichiers ou pages web : le programme n'a pas réussi à déterminer le bon encodage.
Un moyen plus simple serait d'inclure cette indication directement dans le contenu du fichier, au tout début (afin de diminuer les risques de perturbation). On utilise pour cela le fait que tous les encodages actuels soient compatibles avec l'ASCII. C'est ce qu'on verra tout à l'heure pour les pages HTML.