Cahiers GUTenberg

CODAGE DES CARACTÈRES ET MULTI-LINGUISME : DE L'ASCII À UNICODE ET ISO/IEC-10646

Jacques André, Michel Goossens

Cahiers GUTenberg, nº 20 (1995), p. 1-53.

<http://cahiers.gutenberg.eu.org/fitem?id=CG_1995___20_1_0>

Codage des caractères et multi-linguisme : de l'ASCII à UNICODE et ISO/IEC-10646

Jacques ANDRÉ[a] et Michel GOOSSENS[b]

[a] *Irisa/Inria-Rennes*
Campus de Beaulieu
F-35042 Rennes cedex, France
Jacques.Andre@irisa.fr

[b] *Division CN*
CERN
CH-1211 Genève 23, Suisse
Michel.Goossens@cern.ch

Résumé. Après avoir rappelé les notions de glyphe et de caractère, nous étudions les normes classiques d'échange de caractères, telles que ASCII ou ISOLATIN-1. Puis, nous décrivons UNICODE, une norme de codage 16-bits qui a comme but de représenter tous les caractères des langues vivantes pour permettre l'échange sans problèmes de textes rédigés dans les langues parlées des différentes parties du monde. ISO/IEC-10646 est une généralisation à quatre octets – dont les 2 premiers octets coincident avec ceux d'UNICODE – qui permet aussi la représentation de caractères spéciaux et anciens en les codant sur 32 bits.

Abstract. *After revieiwing the difference between glyphs and characters, we discuss character exchange standards, like* ASCII *and* ISO-LATIN *1. Then we turn our attention to* UNICODE*, a 16-bit encoding standard that will eventually represent the characters of all living languages and thus will make it possible to exchange without problems texts written in the languages spoken in various parts of the world.* ISO/IEC*-10646 is a 4-byte generalisation—the first two bytes coinciding with* UNICODE*—but whose full 32-bits wide encoding space allows the representation of special or ancient characters.*

1. Introduction

La base de données *Ethnologue* [24] mentionne plus de 6000 langues parlées dans le monde (l'arborescence génétique de toutes ces langues est disponible sur Internet, tout comme la base de données elle-même[1]). La plupart de ces langues n'a qu'une tradition orale et, selon des spécialistes lors d'un récent congrès organisé par l'UNESCO, plus de la moitié d'entre elles auront disparu avant la fin du siècle sous l'influence de la radio et de la télévision, voire de pressions culturelles, sociales et économiques de langues plus « fortes ». Nous donnons en annexe, table 11, les codes permettant de désigner les langues les plus importantes adoptés par la norme ISO 639 et la proposition d'extension à trois lettres.

[1] Information générale : `gopher://sil.org/11/gopher_root/ethnologue/`,
arborescence : `ftp://nestroy.wu-wien.ac.at/pub/src/Languages/linguist/wgt.lst.gz`,
base de données : `ftp://ftp.std.com/obi/Ethnologue/eth.Z`.

Malgré tout le tapage médiatique fait autour des « autoroutes de l'information » et le monde qui serait devenu un « village planétaire », il reste encore beaucoup de difficultés à surmonter pour conserver ou transmettre des textes saisis dans de nombreuses langues. Un exemple type est un texte comportant des lettres accentuées françaises ou des *umlauts* allemands, les deux codés en 8-bits LATIN-1 (ISO/IEC-8859-1). Très souvent après avoir transité par le courrier électronique, ces textes arrivent avec des caractères sans accents ni *umlauts* ou, pire, remplacés par d'autres caractères. D'ailleurs, même pour de simples textes anglais (c'est-à-dire n'utilisant que les seules 2×26 lettres de l'alphabet latin), on rencontre des problèmes pour les échanger entre ordinateurs de marques différentes ou pour les transmettre à travers les réseaux.

Depuis très longtemps, les hommes ont donc essayé de définir des normes[2] pour les caractères. Mais il faut bien voir qu'un caractère entre dans plusieurs processus et qu'il y a donc divers types de normes, pas toujours cohérentes d'ailleurs ! Regardons ce qui se passe si un Français veut envoyer à un Américain le message laconique suivant : « Œil ». Les diverses questions qui suivent correspondent pratiquement toutes à une norme spécifique :

– quelles touches faut-il taper ?
– où se trouvent ces touches sur le clavier ?
– que va voir ce Français sur son écran ?
– comment ces caractères sont-ils codés dans son ordinateur ?
– que va en faire le *mailer* ?
– comment ces caractères vont-ils transiter sur les réseaux ?
– que va en faire le *mailer* à l'arrivée ?
– comment ces caractères sont-ils codés dans l'ordinateur de l'Américain ?
– que va-t-il voir sur son écran ?
– s'il veut imprimer le message, qu'est-ce qui sera imprimé ?

La dernière question montre bien le problème : notre Américain n'aura probablement pas ce caractère « Œ » sur son imprimante. Ou plutôt, il faudra qu'il utilise un filtre ou qu'il « bidouille » quelques tables de codage (car en fait, contrairement à ce que beaucoup croient, ces caractères « Œ, œ » existent bel et bien dans la majorité des fontes commercialisées). Mais que dire si un Chinois envoie ce même mot « Œil », écrit en chinois, vers un pays arabe ?

Bien qu'il y en ait beaucoup d'autres, les principales normes qui ont été définies sont

1. les normes pour la saisie des caractères (notamment pour les claviers) ;

[2]Nous utilisons ici le mot « norme » de préférence à celui de « standard », non pas parce que ce dernier n'est pas français, mais parce qu'il a pris une connotation de « normal », « banal » (comme le four) et parce qu'il ne reflète pas, en français, tout ce qu'implique de « normatif » le mot norme ! Les « normes » sont définies par des institutions de normalisation, en France l'Afnor (Association Française de NORmalisation, Tour Europe, 92049 Paris La Défense cedex), en Allemagne DIN *(Deutsches Institut für Normalisation)*, aux USA l'ASA *(American Standard Association)* – les photographes connaissent bien ces deux dernières ! – etc., tous ces organismes étant regroupés au sein d'un organisme mondial : l'ISO *(International Standard Organization)*, dont le siège est à Genève. D'autres organismes, comme le CCITT (Comité Consultatif International des Télégraphes et Téléphones), ou l'Ecma (Consortium des constructeurs européens d'ordinateurs) ont aussi défini des normes liées aux caractères. Enfin, il existe des « standards » *de facto* (par exemple EBCDIC ou PostScript) mais qui n'ont pas le statut légal d'une norme !

2. les normes pour l'échange de caractères entre ordinateurs ou autres matériels ;

3. les normes pour la restitution (impression, affichage) des caractères.

Ici, nous allons essentiellement parler des normes d'échange, dont les plus connues sont ASCII, ISOLATIN 1 et maintenant UNICODE ou ISO/IEC-10646. Nous citerons toutefois d'autres normes, réelles ou *de facto* (voir par exemple les sections 6.3 ou 7.1).

Mais pour commencer, voici quelques rappels.

2. Concepts de base

2.1. Systèmes d'écriture

Le principe de l'écriture[3] de la plupart des langues utilisant l'alphabet latin est très simple : les caractères s'écrivent horizontalement, de gauche à droite, sans chevauchement ni changement de direction ; les seuls éléments hors de la ligne sont les signes diacritiques qui se placent au-dessus ou au-dessous de certaines lettres. Même dans ce dernier cas, le nombre de combinaisons différentes dans une langue donnée est en général assez limité pour qu'on puisse donner un code spécifique à chaque lettre combinée à un signe diacritique.

Cependant, en général les systèmes d'écriture ne sont pas si simples. Certaines langues, bien qu'utilisant une écriture basée sur l'alphabet latin, ont une structure plus complexe. Ainsi le vietnamien nécessite-t-il souvent deux signes diacritiques sur une seule lettre, l'un étant un signe tonal. L'alphabet phonétique international (IPA) positionne des éléments diacritiques en indice inférieur ou supérieur ou utilise des signes liant plusieurs lettres. Par ailleurs, il n'est pas possible d'énumérer toutes les combinaisons entre lettres et signes diacritiques pour IPA, puisque le système permet la création de nouvelles combinaisons inédites si nécessaire.

L'arabe et l'hébreu s'écrivent de droite à gauche, mais d'une part les chiffres et d'autre part l'insertion de caractères latins peuvent nécessiter un changement du sens d'écriture dans la même ligne. Seulement les consonnes et voyelles longues sont notées dans ces deux langues ; les voyelles courtes, si elles sont exprimées, sont notées avec des points au-dessus ou en-dessous des consonnes. Les lettres arabes relèvent plus de la tradition calligraphique que de la typographie ; elles ont des formes initiales, médiales, finales et isolées distinctes[4] ; l'hébreu connaît aussi quelques-unes de ces formes ; le grec en garde deux.

[3] Au mot « écriture » est attaché une polysémie évidente. Mais, même aujourd'hui, ou surtout aujourd'hui ?, des linguistes se posent des questions sur l'acceptation du sens de ce mot. Voir par exemple les travaux de Jacques Anis à Nanterre [6].

[4] Certaines propositions de simplification de l'écriture arabe essaient de remédier à cette situation, en utilisant des lettres à deux formes (cf. [45] et [51, fig. 76]) ou même une seule forme (cf. [51, fig. 57], caractères particulièrement beaux). On reviendra là-dessus dans le Cahier GUTenberg dédié à TEX et l'écriture arabe.

Parmi les autres systèmes d'écriture complexes, citons ceux dérivés du *brâhmî* (la plus ancienne écriture de l'Inde classique) qui contiennent beaucoup de ligatures, ou le syllabaire hangul du coréen, où les lettres de base se combinent pour former des blocs syllabiques [13, 15, 18, 19, 30, 47, 48].

2.2. Casses et tables de codage

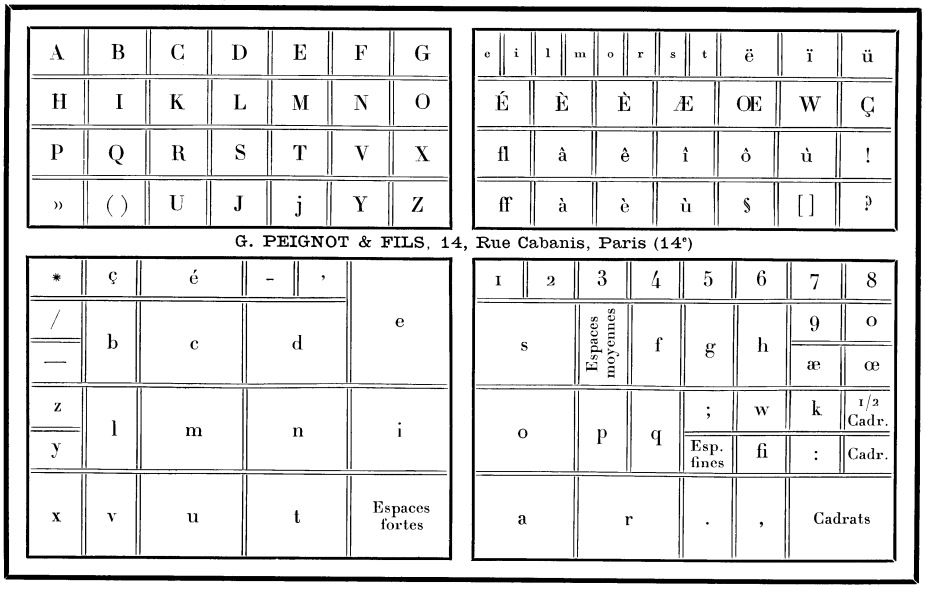


Figure 1 – Casse parisienne

Autrefois, les caractères en plomb étaient mis dans des casses (comme celle de la figure 1) qui étaient « normalisées » : un compositeur trouvait toujours les mêmes lettres au même endroit dans une casse parisienne. On peut dire qu'à chaque caractère était implicitement associé un certain code (*x, y*), *x* et *y* étant les coordonnées de la case correspondante dans la casse. On trouvera dans [9] et [59] quelques éléments sur le contenu de ces casses qui avaient environ entre 90 et 150 cassetins.

Vers 1900, les porte-matrices de la Linotype (figure 2), puis les films des photocomposeuses firent de même : chaque caractère (matrice ou lettre gravée sur le film) était assigné à une place précise. Au lieu de parler de *x, y*, on a simplement numéroté ces cases de 1 à *n* (ou plutôt de 0 à $n \Leftrightarrow 1$), *n* étant en général une puissance de 2.

Mais entre la Linotype et les premières photocomposeuses est apparu quelque chose dont on n'a peut-être pas bien signalé l'importance : en 1932 le système TeleTypeSetter (TTS) utilisa les premières bandes perforées (à 6 canaux) pour conserver et transmettre les textes « composés » [59]. Même si cette technologie n'a pas vraiment été

Figure 2 – Châssis porte matrice de la Monotype (d'après [49])

utilisée avant 1950, c'est, à notre avis, la première apparition de la notion d'échange de textes balisés. Le soucis d'homogénéiser les diverses balises utilisées par la suite par les photocomposeuses a d'ailleurs conduit à deux types extrèmes de normes (dont nous ne parlerons plus ici) : SGML et les PDL (langages de description de page comme PostScript ou SPDL ; voir [20]). Par contre, cette notion de texte balisé correspond plutôt à la notion d'« échange » tandis que la notion de casse ou de matrice correspond à celle de table de codage des fontes (voir section 2.4).

2.3. Caractère, nom, œil et glyphe

La majorité des normes de codage des caractères définissent deux choses :

1. un numéro de code associé à un caractère, par exemple (en octal) 073 pour « ; », 101 pour « A », 102 pour « B », 340 pour « à », etc.

2. un nom pour chaque caractère, par exemple *semicolon* pour « ; », *A* pour « A », *B* pour « B », *Agrave* pour « À » et *agrave* pour « à » – avec bien sûr des traductions officielles pour chaque langue (en France, c'est l'Afnor qui fait ce travail).

2.3.1. Qu'est ce qu'un caractère ?

Le mot « caractère » a plusieurs sens, allant du « type » en plomb à la trace imprimée que laisse ce dernier en passant par l'élément d'un alphabet. Même avec ce dernier sens, il subsiste quelques problèmes.

Les différentes opérations exécutées sur les textes par ordinateur (saisie, affichage, recherches, tris etc.) ont souvent des exigences incompatibles pour le codage de l'information. Par exemple, l'affichage serait simplifié si les formes de présentation correspondant aux ligatures comme « fi » ou « fl » étaient codées explicitement, mais ceci compliquerait la saisie et rendrait recherches et tris inutilement difficiles. Ceci n'est qu'un exemple trivial, mais il montre bien qu'un codage représente un compromis entre les exigences des différents types de traitement envisagés.

Dans l'alphabet latin, savoir ce qui est ou non un caractère se pose principalement pour les signes diacritiques ou les ligatures. Pour garantir une flexibilité maximale on doit inclure tous les signes diacritiques (pour permettre la création de nouvelles combinaisons, comme pour l'IPA ou la large variété de langues écrites à l'aide de l'alphabet latin), mais aussi des codes pour certains caractères précomposés (comme en turc). Des ligatures « typographiques », comme « fi », ne devraient pas être inclues mais la ligature « æ » qui fait partie de certains alphabets scandinaves a sa place dans une séquence de tri. Même le « œ » français doit être considéré comme un caractère unique (et non comme une ligature) avec une majuscule correspondante : on écrit « Œuvres de BALZAC » ; à la rigueur, faute de « Œ », on acceptera « OEuvres de BALZAC » mais sûrement pas « Oeuvres de BALZAC » ! Mais le problème est que toutes les séquences françaises « oe » ne sont pas à écrire systématiquement « œ » (par exemple dans « coexistence »). Faute de pouvoir les déterminer automatiquement, ces séquences doivent être codées explicitement.

Des variantes dépendant de la position dans les mots (comme en arabe) sont manifestement des formes de présentation qui ne devraient pas être codées séparément, alors qu'en hébreu et en grec, où on a peu de variantes, celles-ci sont codées explicitement. Ceci montre que dans chaque système d'écriture il y a des situations où on ne peut distinguer clairement entre ce qu'il faut considérer comme un caractère ou pas. Aussi doit on prendre en compte l'existence des systèmes de codage actuels et s'assurer que tous les textes codés précédemment à l'aide d'un de ces systèmes pourront aussi être codés dans tout nouveau système.

On est donc amené à faire une distinction très nette entre la notion abstraite de caractère et celle de représentation, appelée désormais glyphe.

2.3.2. Différence entre un caractère et un glyphe

Un « caractère » est une unité d'information utilisée pour coder du texte, alors qu'un « glyphe » est une forme géométrique (une collection homogène de telles formes constitue une police) utilisée pour présenter un texte. Le processus de présentation nécessite une application (par nécessairement bi-univoque) des caractères vers des glyphes.

Table 1 – À un caractère peuvent correspondre plusieurs glyphes, et réciproquement.

Caractère(s)	Glyphe(s) possible(s)
Lettre majuscule A	A, *A*, 𝙰, $\mathcal{A}$
Lettre majuscule ALPHA	A, *A*, 𝙰, $\mathcal{A}$
Minuscule f suivie de minuscule i	ﬁ, fi

En fait cette notion correspond à celle d'« œil » en typographie française. L'encyclopédie *La chose imprimée* donne par exemple comme première définition à œil : « Quelle que soit l'origine d'une composition (chaude ou froide), l'œil des caractères est ce que l'on voit sur le papier. L'œil d'un A ou d'un a, d'un B ou d'un b, etc. est le signe imprimé permettant d'identifier chacune de ces lettres respectivement en tant que A, a, B, b, etc. » [16]. Mais le néologisme américain « glyphe » commence à être très employé, ainsi le gardons nous ici. D'autant qu'il a quand même l'avantage de supprimer la polysémie du mot œil !

Cette distinction conduit à deux principes pour la création d'un jeu de caractères (dans le cadre de normes d'échange) :

1. même si des candidats au codage sont visuellement identiques (comme le A majuscule latin et le alpha majuscule grec de la table 1) et peuvent de ce fait être représentés par un même glyphe, ils doivent quand-même être codés séparément pour avoir une correspondance bi-univoque entre majuscules et minuscules dans un alphabet donné et pour garantir une invariance aller-retour des données avec les normes existantes ;

2. les variations de forme (des glyphes multiples) exigées par une présentation de qualité supérieure d'un texte ne *doivent pas* être codées comme des caractères séparés si leurs significations sont identiques.

Notons également que cette notion permet de traiter le cas des caractères composites utilisés, par exemple, pour les formules mathématiques de TEX : un symbole intégrale, une grande accolade, etc. correspondent à un caractère ; mais leur représentation est formée de glyphes en nombre variable. Pour une intégrale, par exemple, on aura la crosse supérieure, plusieurs barres verticales et la crosse inférieure. Il n'y a pas de glyphe « intégrale », mais trois glyphes formant une intégrale.

Une anecdote : comme nous le verrons, le code ASCII comprend des caractères (comme celui dont le nom est BELL ; voir ces noms au début de la table 10 en annexe) qui ne sont pas imprimables, c'est-à-dire qui n'ont pas de glyphe associé. Mais UNICODE a ajouté une série de caractères *(Pictures for Control Codes)* dont le glyphe est le nom du caractère (voir par exemple, dans ce *Cahier* [9, figure 3]).

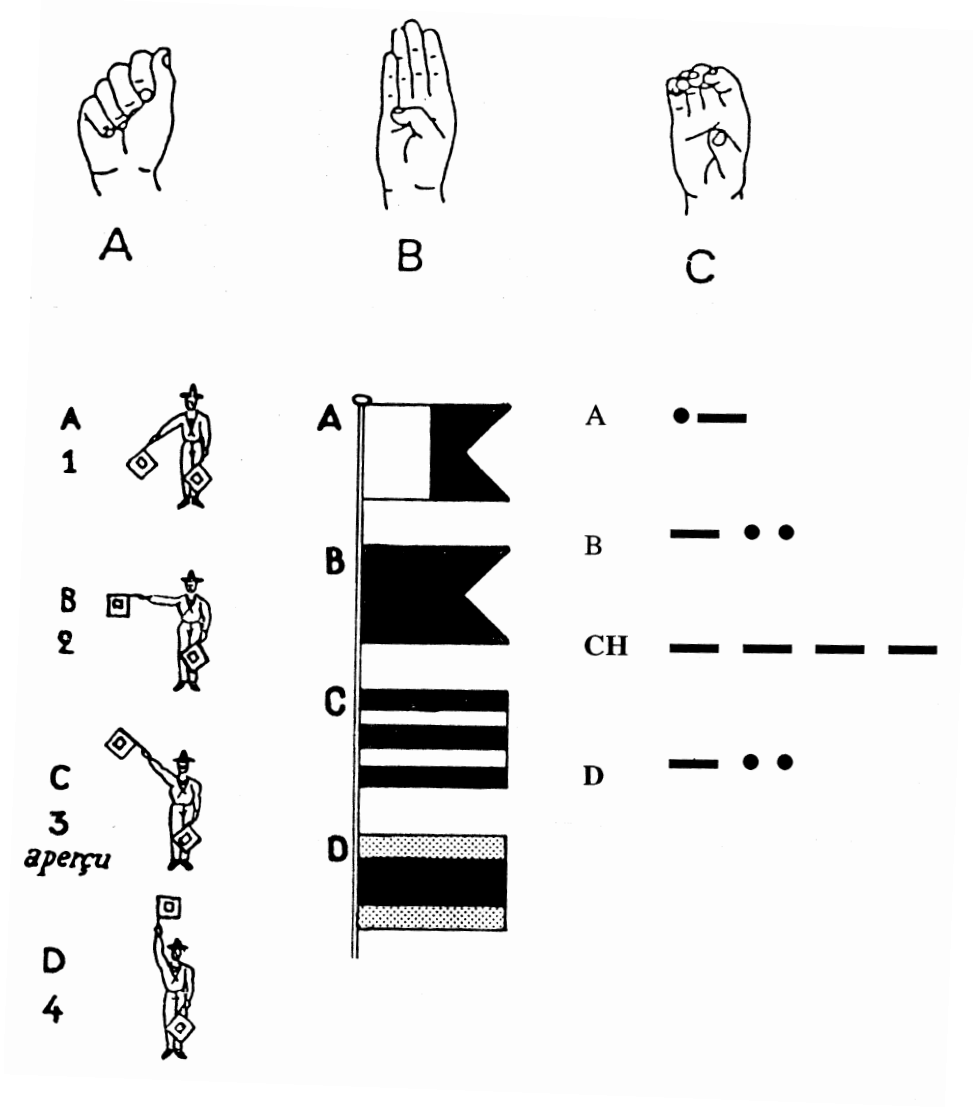


Figure 3 – Quelques codes idéographiques : langage des sourds, sémaphore, fanions de la marine et morse.

2.4. Normes d'échange de caractères

Les normes permettant l'échange de données alphabétiques ne sont pas récentes. Leur importance en communication explique le rôle prépondérant joué par les organismes de télécommunication des divers pays dans les instances de normalisation. En effet, les premières « normes » ont été les idéogrammes des codes du télégraphe de

Chappe ou les fanions de la marine (voir figure 3) ; mais ce sont les alphabets Morse et ceux du Télex qui auront été les premiers « alphabets internationaux »[5]. Ces alphabets, à 6 moments (donc avec $2^6 = 64$ caractères) ne comprenaient que les lettres majuscules, les chiffres, quelques signes de ponctuation et certains codes réservés (le Télex utilise par exemple un code pour « sonnerie »). Parmi eux, il convient aussi de citer le « standard » d'IBM appelé BCD *Binary Coded Decimal* d'où est issu EBCDIC (voir section 7.2).

C'est donc dans l'esprit d'échange que plusieurs normes pour l'informatique ont été définies, à 7 moments ($2^7 = 128$ caractères) comme l'ASCII *(American Standard Coded Information Interchange)*. Ce dernier a été légèrement modifié et adopté comme « alphabet international Numero 13 » IA5 en 1963 par l'ISO *(International Standard Organization)* et le CCITT (Commité Consultatif International du Télégraphe et Téléphone) sous le nom de norme ISO 646. Depuis, de nombreuses normes (voir [50]) ont été redéfinies pour coder les caractères avec plus de bits ou moments ou pour s'adapter à diverses langues, dont les célèbres ISO/IEC-8859-n sur 8 bits et UNICODE à 16 bits, étendu à 32 bits pour devenir ISO/IEC-10646. Ces normes assurent toutes la compatibilité avec ISO 646, d'où la très grande importance de cette norme même si maintenant elle a vieilli ! Ces normes ont les tailles du tableau ci-après.

Nom du code	nombre de bits ou moments	nombre de caractères
Telex	6	64
ASCII	7	128
ISO-LATIN 1	8	256
UNICODE	16	65536
ISO/IEC 10646	32	>2 milliards

Voyons maintenant quel est le contenu de ces trois principales normes.

3. ASCII ou ISO 646

Jusqu'à ce jour le seul codage universellement utilisé est l'ASCII *(American Standard for Information Interchange)*. Comme son nom l'indique, ce codage a vu le jour en Amérique vers 1967 et a fourni pendant plus de deux décennies le seul codage non-ambigu à 7-bits.

Remarque : il s'agit donc d'une norme d'échange. Mais, par abus de langage et sans doute par méconnaissance des principes de codage, certains informaticiens ont tendance à utiliser ce mot avec le sens de « non formaté » : par exemple, pour eux,

```
\section{Benoît est-il allé à Canossa ?}
Dans ce chapitre, nous faisons allusion au fait que
```

[5] Respectivement connus sous le nom de IA1 et IA2. Voir [50] à ce sujet et de façon plus générale pour toutes ces normes.

Table 2 – Le codage ASCII dans sa version finale de 1988 (ISO 646).

octal	0	1	2	3	4	5	6	7
/0 x	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL
/1 x	BS	HT	LF	VT	FF	CR	SO	SI
/2 x	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB
/3 x	CAN	EM	SUB	ESC	GS	FS	RS	US
/4 x		**!**	**"**	**#**	**$**	**%**	**&**	**'**
/5 x	**(**	**)**	*****	**+**	**,**	**–**	**.**	**/**
/6 x	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
/7 x	**8**	**9**	**:**	**;**	**<**	**=**	**>**	**?**
/10x	**@**	**A**	**B**	**C**	**D**	**E**	**F**	**G**
/11x	**H**	**I**	**J**	**K**	**L**	**M**	**N**	**O**
/12x	**P**	**Q**	**R**	**S**	**T**	**U**	**V**	**W**
/13x	**X**	**Y**	**Z**	**[**	****	**]**	**^**	**_**
/14x	**‘**	**a**	**b**	**c**	**d**	**e**	**f**	**g**
/15x	**h**	**i**	**j**	**k**	**l**	**m**	**n**	**o**
/16x	**p**	**q**	**r**	**s**	**t**	**u**	**v**	**w**
/17x	**x**	**y**	**z**	**{**	**|**	**}**	**~**	DEL

```
\textsc{Canossa} est c\'el\‘ebre pour \ldots ;
```

est « codé en ASCII » alors qu'il n'en est rien (comme nous allons le voir de suite « î », « é », « à », etc. ne font partie de ce codage). À plus forte raison, faut-il s'élever contre l'emploi d'expression comme « en ASCII 8 bits » !

Le principe de la norme ISO 646, reprenant la norme ASCII, est une structure à 7 moments (7 bits) permettant donc le codage de 128 caractères. Cette table comprend en fait 2 parties (table 2) :

34 caractères[6] dits (à tort) « de contrôle »(tels que *Carriage Return*, *Line Feed*, *Bell*, etc. voir table 10) et
94 caractères dits « graphiques » car on peut les afficher sur un écran ou les imprimer.

Ces 94 derniers caractères sont eux-mêmes répartis en 3 groupes :

— 82 caractères obligatoires :
52 lettres : A–Z et a–z,
10 chiffres : 0–9,
20 signes de ponctuation ou autres : ! " % & ' ( ) * + , - . / : ; < = > ? _

[6]Les 32 premiers, le suivant qui est en fait l'espace et le dernier (de code binaire 1111111) pour DEL.

Table 3 – Les caractères optionnels de la version de référence IVR de la norme ISO 646 et de quelques variantes nationales – d'après Marti : [50, page 243])

Version de référence	#	¤	@	[	\	]	^	‘	{	|	}	~
Allemagne (DIN66003)	#	$	§	Ä	Ö	Ü	^	‘	ä	ö	ü	ß
Belgique	#	$	à	°	ç	§	^	‘	é	ij	è	~
Espagne	#	$	·	¡	Ñ	Ç	¿	‘	’	ñ	ç	"
France (NF Z62010/1982)	£	$	à	°	ç	§	^	μ	é	ù	è	"
Grande Bretagne	£	$	@	[	\	]	^	‘	{	|	}	~
Suisse romande			à		ç				é	ù	è	~
USA (norme ASCII)	#	$	@	[	\	]	^	‘	{	|	}	~

— deux caractères « au choix » [7] :
  # ou £
  $ ou ¤ (symbole monétaire international *currency*[8])

— 10 positions réservées à des caractères d'usage national.

La norme ISO 646 comprenait donc à l'origine :

— des variantes nationales (parfois plusieurs pour un même pays – c'est le cas de la France) ;

— une version internationale de référence, IRV, où les positions optionnelles sont affectées d'un caractère précis.

La table 3 montre la version de référence IRV et quelques exemples de ces versions nationales. Ces normes n'ont en fait pas été très suivies car mal adaptées ou flouées.

— La version internationale de référence n'était pas la version américaine ASCII (d'ailleurs appelée, à l'époque, USASCII) : IRV contenait le symbole ¤ tandis que USASCII utilisait le dollar. On ne trouve donc le symbole ¤ sur pratiquement aucun matériel informatique[9].

— Les informaticiens américains se sont mis à utiliser nombre des caractères optionnels (#, @, les accolades, etc.) dans leurs programmes ce qui a donné un poids anormalement fort à la version américaine ASCII de la norme 646.

[7] Ce « choix » a été demandé par divers états, dont l'URSS et la Grande Bretagne, lors de la Guerre froide afin de contrer l'hégémonie du dollar ! On verra plus bas, que lors de la Perestroïka, le dollar a repris sa place au dépend du symbole monétaire international..

[8] Le glyphe de ce symbole représente une pièce d'or où brillent quatre rayons de soleil.

[9] Faisant partie du codage standard Macintosh, on trouve ce signe dans toutes les polices de caractères commerciales. Par contre, en ce qui concerne le clavier, on ne rencontre ce symbole que sur certains matériels à vocation internationale, notamment lorsqu'ils assurent la compatibilité avec ISOLATIN1. Dans le cas des claviers français des stations SUN, on a non seulement cette touche *currency* mais aussi le μ de la norme française (voir table 3) qui est resté dans ISOLATIN-1 (tableau 5).

— Il y avait une grande incohérence d'un pays francophone à l'autre à tel point d'ailleurs que la France a abandonné cette norme Z62010 au profit de l'ASCII en 1983.

C'est pourquoi, en 1988, ISO 646 a pris exactement le codage ASCII de la table 2.

Par ailleurs, les caractères de commande (dits de « contrôle ») correspondent à une technologie périmée : certaines commandes sont complètement dénuées de sens aujourd'hui[10], d'autres ont été très mal définies[11]. Les fabricants de matériel ou de logiciel ont donc pris l'habitude d'y mettre des codes à eux[12], mais avec la plus grande anarchie. Elle a donc été remplacée par de nouvelles normes d'échange.

Néanmoins, compatibilité oblige, cette norme sert de base à toutes les autres normes et en particulier à ISO-LATIN-1. Par ailleurs, comme elle est suffisante pour la majorité des Américains, cette norme ASCII reste très importante !

À propos des caractères spéciaux

C'est donc cette compatibilité à l'ASCII (et accessoirement à ISOLATIN-1) qui fait que l'on a sur nos claviers ces 94 caractères et tous ces caractères spéciaux « & @ £ $ # ». Or, bien peu de Français connaissaient ces symboles il y a quelques années à peine ; on ne savait pas leurs noms, ni à quoi ils servaient ; et on ne sait toujours pas bien les dessiner du premier coup. Mais à lire les questions de *comp.font*[13], on se rend compte que beaucoup d'Américains ignorent eux aussi ce que sont ces symboles. Voici donc quelques explications rapides (voir [5]).

Tous ces symboles sont utilisés en comptabilité américaine (il n'est pas inutile de rappeler que BM dans IBM veut dire *Business Machines*). Plus exactement, ce sont des symboles issus du temps où, dans les chancelleries, les scribes écrivaient à la main et avaient l'habitude de faire des abréviations et des ligatures, lesquelles ont pris des connotations très officielles au XVIII^{e} siècle et qui ne sont restées en usage que dans des mondes éloignés comme « l'Amérique » et fermés comme la comptabilité, connotations qui ont pris un poids très fort lorsque d'une part « l'Amérique » et d'autre part sa puissance économique ont fait surface. Ce qui était ringard voire complètement oublié ou périmé chez nous s'est donc trouvé présenté comme une nouveauté moderne qu'il a bien fallu suivre !

Bref, ces symboles comptables anglo-américains se sont imposés dans les jeux de caractères des ordinateurs. Mais comme tous les informaticiens, américains ou non, ne font pas de la gestion et comme ils ont toujours besoin de symboles « spéciaux », ceux

[10] Notamment à cause de changements de technologie, comme l'abandon des rubans papier pour lesquels DEL était prévu pour annuler un caractère erroné en perforant toutes ses positions dans la colonne.

[11] Typiquement, ce codage permettait, pour certains écrans, d'afficher un caractère accentué en saisissant successivement le code ESC, le caractère, l'accent et enfin le code BS de retour en arrière. Mais ceci ne permettait pas de faire le moindre tri.

[12] Ainsi, TEX y place beaucoup de caractères comme fi, ffi, etc. comme on le voit dans le standard de Cork, table 9.

[13] Il s'agit de l'un du millier (sinon plus) de groupes actifs USENET, qui est un système de conférence international, auquel tous les utilisateurs du réseau planétaire Internet ont la possibilité de s'abonner. Voir par exemple [46].

qui développaient des logiciels se sont appropriés ces symboles. Aussi, même si ces symboles n'ont vraiment aucune raison d'être sur tous les ordinateurs en tant que symboles comptables, on aurait du mal aujourd'hui à s'en passer pour les langages informatiques.

Dollar $ C'est évidemment le plus connu de ces symboles : c'est le symbole monétaire des États-Unis d'Amérique.

Le mot *dollar* vient du nom populaire d'une monnaie mexicaine *dolera* (dont le vrai nom était *peso*, du latin *pensum*, poids) qui vient de l'allemand *Thaler* (en bas-allemand *Daler*) du nom d'une monnaie frappée (et rendue plus ou moins européenne en 1537, déjà !, par Charles Quint) dans la vallée de Joachim (en allemand *Joachimthal*).

L'origine du signe n'est pas connue[14]. Contrairement à ce que disent beaucoup d'Américains, ce symbole n'est pas dessiné en surimposant le U et le S de US : cette explication n'est pas possible car le symbole $ est attesté avant la création des États Unis d'Amérique ! ; plusieurs hypothèses ont été émises. La première est qu'il s'agit d'un 8 déformé : cette monnaie espagnole *dolera* s'appelait aussi « la pièce de 8 » car elle valait 8 réales. La seconde est que cette monnaie d'origine espagnole portait au revers le symbole de Gibraltar (Jebel Tarek) : deux barres verticales pour symboliser le détroit (les colonnes d'Hercule), et un drapeau flottant comme un S. Enfin, ce symbole viendrait du « p » de *peso* avec une barre en biais, déformation du « s » pluriel en exposant, comme dans beaucoup de monnaies. En tout cas son origine espagnole est quasi certaine !

Livre £ C'est le symbole monétaire *livre sterling* britannique. Son nom vient d'une ancienne monnaie d'argent qui valait une livre (poids : *pound*) d'argent. On retrouve donc la dualité poids/valeur-monétaire pour *pounds* comme pour *peso* et nos livres (tournois ou de beurre). Le mot livre vient du latin *libra* qui a aussi donné l'italien *lira*. Le mot *sterling*, nom adopté par ISOLATIN-1, est d'origine obscure ! Le dessin représente un « $\mathcal{L}$ », abréviation de *Libra*, et ressemble donc à celui de la lire italienne.

Esperluette & Il s'agit de la très vieille ligature « et » qui a fait l'objet d'études célèbres de Ian TSCHICHOLD [60] et de Gérard BLANCHARD [10]. Ce caractère est très utilisé aux États-Unis (plus qu'en France) dans les noms de sociétés commerciales (par exemple *Bigelow & Holmes* et plus généralement sous la forme *& Co.*).

Son nom français est « esperluette » ; mais il y a beaucoup de variantes : « perluète » pour ISOLATIN, « perluette » ou « eperluette » ; il est aussi appelé *et commercial* (voir ci-dessous le *a commercial*) ce qui confirme son origine comptable. L'origine de ce mot esperluette n'est pas non plus bien connue : le Grevisse dit que ce caractère s'appelait « ète » (c'est-à-dire « et » prononcé à la latine) et qu'il était placé dans l'alphabet après le z, alphabet que les enfants chantaient « a, b, ...,

[14] On trouvera dans [11], citée par Mark Brader dans *comp.font* le 3 avril 1995, l'étude la plus sérieuse sur le sujet.

z et puis le ète » ce qui aurait donc donné « éperluette » [23, art. 71]. Son nom anglais est *ampersand* et est en fait un mélange de latin et d'anglais : *and per se and* (et à lui tout seul « et »). On raconte aussi la même histoire qu'en français d'alphabet chanté [61].

A commercial @ Voici en tout cas un caractère qui était pratiquement inconnu en France il y a quelques années à peine.

Comme le &, ce caractère est aussi issu des chancelleries ; c'est la ligature latine *ad* (« à » en français) où le a et le d cursifs de l'onciale (*að*) ont fini par se confondre. Ce caractère n'est utilisé, aux USA, qu'en comptabilité pour indiquer les prix unitaires : ainsi « deux livres à 1 dollar pièce » s'écrit dans une facture « 2 books @ $ 1 ».

Le nom français de ce caractère, selon la version française (Afnor) d'ISOLATIN est « a commercial ». Cependant, le nom que lui donnent les informaticiens français tourne autour de sa forme : a-rabesque, a-rondi, a roulé, a-arrondi. Mais le nom le plus fréquemment employé, du moins dans les milieux universitaires, est « arobas ». Ce mot vient d'une confusion avec le symbole d'une unité de poids espagnol (*arroba*, poids de 25 livres espagnoles, soit 11, 502 kg, dont le vrai nom français d'ailleurs est « arobe » selon le Robert).

Numéro # Le symbole # est aussi issu d'une ligature latine : *numerus* (nombre), un « n » surmonté d'une barre, c'est-à-dire « $\overline{n}$ » et dont la barre est descendue peu à peu au bas des jambes du « n ». En comptabilité américaine, ce signe sert à indiquer des numéros (de pièce, de série, de compte bancaire, etc.) ; dans un hôtel, on numérote les chambres « # 101, # 102, etc. » et dans une facture, on écrira par exemple « 4 gonds numéro 78-9253 » sous la forme « *4 hinges # 78-9253* ».

Ce caractère est souvent appelé dièse en français à cause de sa ressemblance avec le signe musical, mais « # ≠ ♯ ! ». Son nom américain est *number* (« numéro », c'est d'ailleurs la traduction officielle adoptée par l'Afnor pour ISOLATIN) ; mais dans le langage Ada il s'appelle aussi *sharp* (dièse). En Grande-Bretagne, il est parfois appelé *hash* (hacher).

4. ISO-LATIN et ISO-8859

L'anglais étant la seule langue utilisable avec l'ASCII, de nombreux organismes ont bien sûr tenté de définir des normes plus riches. Outre divers « standards *de facto* » comme EBCDIC (voir ci-après section 7.2), il convient de citer ici la norme la plus importante pour les langues européennes, définie par l'ISO et connue sous le nom d'ISO/IEC8859-n (avec n de 1 à 12), qui est une extension à 8 bits de l'ASCII. Le seul fait de passer de 7 à 8 bits permettait de doubler le nombre de caractères, donc de passer à 256 caractères (moins les fameux caractères de contrôle !). Comme en Europe, il y a plus de 256 caractères différents utilisés, il a été décidé de les regrouper par affinités … commerciales. C'est ainsi qu'il y a le premier groupe, ISOLATIN-1 pour la zone occidentale, LATIN-2 pour la zone orientale, etc. (voir [36], la figure 4 et la table 4). Pour des raisons

Table 4 – Les normes ISO/IEC-8859-n.

8859-1	*Europe occidentale, Amérique latine (*ISOLATIN-1*)* allemand, anglais, danois, espagnol, féroïen, finnois, français, islandais, italien, néerlandais, norvégien, portugais, suédois
8859-2	*Europe orientale (*ISOLATIN-2*)* albanais, allemand, anglais, croate, hongrois, polonais, roumain, slovaque, slovène, tchèque
8859-3	*autres langues utilisant l'alphabet latin (*ISOLATIN-3*)* afrikaans, anglais, allemand, catalan, espagnol, esperanto, italien, maltais, néerlandais, turc
8859-4	*Europe du nord (*ISOLATIN-4*)* allemand, anglais, danois, estonien finnois, groenlandais, letton, lituanien, norvégien, suédois, sami
8859-5	*latin/cyrillique* anglais, bulgare, byelorusse, macédonien, russe, serbe, ukrainien
8859-6	*latin/arabe*
8859-7	*latin/grec*
8859-8	*latin/hebreu*
8859-9	*variante de Latin-1 pour le turc (*ISOLATIN-5*)*
8859-10	*sami/nordique/eskimo (*ISOLATIN-6*)* allemand, anglais danois, estonien, féroïen, finnois, groenlandais, islandais, letton, lituanien, norvégien, suédois, sami
8859-11	*latin/thailandais (en préparation)*
8859-12	*latin/devanagari (en préparation)*

politico-économiques, un codage spécial (LATIN-5) a dû être ajouté pour la Turquie et ses partenaires !

Il faut remarquer néanmoins que ces jeux ne sont pas utilisés universellement et qu'en plusieurs pays des variantes incompatibles coexistent. De plus, des problèmes liés à la traduction entre les différents codages subsistent. Seul le codage ISO/IEC8859-1 (« LATIN-1 ») a été implanté généralement et est devenu un remplacement *de facto* de la norme ASCII en Europe ; nous allons donc y revenir un peu.

ISOLATIN-1

Les caractères du codage LATIN-1 sont montrés dans les tables 5 et 10 et en figure 4. On y remarquera plusieurs choses :

1. Les 128 premiers caractères sont ceux de la norme ASCII. C'est vrai aussi pour ISO8859-*n* quelque soit *n*.

Table 5 – Le codage des caractères d'Europe occidentale : ISOLATIN 1 (les cases blanches correspondent aux caractères de contrôle non imprimables).

	!	"	#	$	%	&	’
(	)	*	+	,	–	.	/
0	1	2	3	4	5	6	7
8	9	:	;	<	=	>	?
@	A	B	C	D	E	F	G
H	I	J	K	L	M	N	O
P	Q	R	S	T	U	V	W
X	Y	Z	[	\	]	^	_
‘	a	b	c	d	e	f	g
h	i	j	k	l	m	n	o
p	q	r	s	t	u	v	w
x	y	z	{	|	}	~	
ı	`	´	ˆ	˜	¯	˘	˙
¨		°	¸		˝	˛	ˇ
	¡	¢	£	¤	¥	¦	§
¨	©	ª	«	¬	-	®	¯
°	±	²	³	´	µ	¶	·
¸	¹	º	»	¼	½	¾	¿
À	Á	Â	Ã	Ä	Å	Æ	Ç
È	É	Ê	Ë	Ì	Í	Î	Ï
Ð	Ñ	Ò	Ó	Ô	Õ	Ö	×
Ø	Ù	Ú	Û	Ü	Ý	Þ	ß
à	á	â	ã	ä	å	æ	ç
è	é	ê	ë	ì	í	î	ï
ð	ñ	ò	ó	ô	õ	ö	÷
ø	ù	ú	û	ü	ý	þ	ÿ

Partie inférieure (positions 33–128) commune à ASCII et toutes les normes ISO/IEC-8859-x

	!	"	#	$	%	&	'	(	)	*	+	,	-	.	/
0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
@	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
p	q	r	s	t	u	v	w	x	y	z	{	|	}	~	

ISO–8859–1

	¡	¢	£	¤	¥	¦	§	¨	©	ª	«	¬	-	®	¯
°	±	²	³	´	µ	¶	·	¸	¹	º	»	¼	½	¾	¿
À	Á	Â	Ã	Ä	Å	Æ	Ç	È	É	Ê	Ë	Ì	Í	Î	Ï
Ð	Ñ	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú	Û	Ü	Ý	Þ	ß
à	á	â	ã	ä	å	æ	ç	è	é	ê	ë	ì	í	î	ï
ð	ñ	ò	ó	ô	õ	ö	÷	ø	ù	ú	û	ü	ý	þ	ÿ

ISO–8859–2

	Ą	˘	Ł	¤	Ľ	Ś	§	¨	Š	Ş	Ť	Ź	-	Ž	Ż
°	ą	˛	ł	´	ľ	ś	ˇ	¸	š	ş	ť	ź	˝	ž	ż
Ŕ	Á	Â	Ă	Ä	Ĺ	Ć	Ç	Č	É	Ę	Ë	Ě	Í	Î	Ď
Đ	Ń	Ň	Ó	Ô	Ő	Ö	×	Ř	Ů	Ú	Ű	Ü	Ý	Ţ	ß
ŕ	á	â	ă	ä	ĺ	ć	ç	č	é	ę	ë	ě	í	î	ď
đ	ń	ň	ó	ô	ő	ö	÷	ř	ů	ú	ű	ü	ý	ţ	˙

ISO–8859–3

	Ħ	˘	£	¤		Ĥ	§	¨	İ	Ş	Ğ	Ĵ	-		Ż
°	ħ	²	³	´	µ	ĥ	·	¸	ı	ş	ğ	ĵ	½		ż
À	Á	Â		Ä	Ċ	Ĉ	Ç	È	É	Ê	Ë	Ì	Í	Î	Ï
	Ñ	Ò	Ó	Ô	Ġ	Ö	×	Ĝ	Ù	Ú	Û	Ü	Ŭ	Ŝ	ß
à	á	â		ä	ċ	ĉ	ç	è	é	ê	ë	ì	í	î	ï
	ñ	ò	ó	ô	ġ	ö	÷	ĝ	ù	ú	û	ü	ŭ	ŝ	˙

ISO–8859–4

	Ą	ĸ	Ŗ	¤	Ĩ	Ļ	§	¨	Š	Ē	Ģ	Ŧ	-	Ž	¯
°	ą	˛	ŗ	´	ĩ	ļ	ˇ	¸	š	ē	ģ	ŧ	Ŋ	ž	ŋ
Ā	Á	Â	Ã	Ä	Å	Æ	Į	Č	É	Ę	Ë	Ė	Í	Î	Ī
Đ	Ņ	Ō	Ķ	Ô	Õ	Ö	×	Ø	Ų	Ú	Û	Ü	Ũ	Ū	ß
ā	á	â	ã	ä	å	æ	į	č	é	ę	ë	ė	í	î	ī
đ	ņ	ō	ķ	ô	õ	ö	÷	ø	ų	ú	û	ü	ũ	ū	˙

ISO–8859–5

	Ё	Ђ	Ѓ	Є	Ѕ	І	Ї	Ј	Љ	Њ	Ћ	Ќ	-	Ў	Џ
А	Б	В	Г	Д	Е	Ж	З	И	Й	К	Л	М	Н	О	П
Р	С	Т	У	Ф	Х	Ц	Ч	Ш	Щ	Ъ	Ы	Ь	Э	Ю	Я
а	б	в	г	д	е	ж	з	и	й	к	л	м	н	о	п
р	с	т	у	ф	х	ц	ч	ш	щ	ъ	ы	ь	э	ю	я
№	ё	ђ	ѓ	є	ѕ	і	ї	ј	љ	њ	ћ	ќ	§	ў	џ

ISO–8859–6

					¤							،	-		
												؛				؟
	ء	آ	أ	ؤ	إ	ئ	ا	ب	ة	ت	ث	ج	ح	خ	د
ذ	ر	ز	س	ش	ص	ض	ط	ظ	ع	غ					
ـ	ف	ق	ك	ل	م	ن	ه	و	ى	ي	ً	ٌ	ٍ	َ	ُ
ِ	ّ	ْ													

ISO–8859–7

	‘	’	£	€	₯	¦	§	¨	©	ͺ	«	¬	-		―
°	±	²	³	΄	΅	Ά	·	Έ	Ή	Ί	»	Ό	½	Ύ	Ώ
ΐ	Α	Β	Γ	Δ	Ε	Ζ	Η	Θ	Ι	Κ	Λ	Μ	Ν	Ξ	Ο
Π	Ρ		Σ	Τ	Υ	Φ	Χ	Ψ	Ω	Ϊ	Ϋ	ά	έ	ή	ί
ΰ	α	β	γ	δ	ε	ζ	η	θ	ι	κ	λ	μ	ν	ξ	ο
π	ρ	ς	σ	τ	υ	φ	χ	ψ	ω	ϊ	ϋ	ό	ύ	ώ	

ISO–8859–8

			¢	£	¤	¥	¦	§	¨	©	×	«	¬	-	®	¯
°	±	²	³	´	µ	¶	·	¸	¹	÷	»	¼	½	¾	
																‗
א	ב	ג	ד	ה	ו	ז	ח	ט	י	ך	כ	ל	ם	מ	ן	נ
ס	ע	ף	פ	ץ	צ	ק	ר	ש	ת							

ISO–8859–9

	¡	¢	£	¤	¥	¦	§	¨	©	ª	«	¬	-	®	¯
°	±	²	³	´	µ	¶	·	¸	¹	º	»	¼	½	¾	¿
À	Á	Â	Ã	Ä	Å	Æ	Ç	È	É	Ê	Ë	Ì	Í	Î	Ï
Ğ	Ñ	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú	Û	Ü	İ	Ş	ß
à	á	â	ã	ä	å	æ	ç	è	é	ê	ë	ì	í	î	ï
ğ	ñ	ò	ó	ô	õ	ö	÷	ø	ù	ú	û	ü	ı	ş	ÿ

ISO–8859–10

	Ą	Ē	Ģ	Ī	Ĩ	Ķ	§	Ļ	Đ	š	Ŧ	ž	-	Ū	Ŋ
°	ą	ē	ģ	ī	ĩ	ķ	·	ļ	đ	š	ŧ	ž	―	ū	ŋ
Ā	Á	Â	Ã	Ä	Å	Æ	Į	Č	É	Ę	Ë	Ė	Í	Î	Ï
Ð	Ņ	Ō	Ó	Ô	Õ	Ö	Ũ	Ø	Ų	Ú	Û	Ü	Ý	Þ	ß
ā	á	â	ã	ä	å	æ	į	č	é	ę	ë	ė	í	î	ï
ð	ņ	ō	ó	ô	õ	ö	ũ	ø	ų	ú	û	ü	ý	þ	ĸ

Figure 4 – Parties inférieure (commune) et supérieures (positions 161–265) des normes ISO/IEC8859-1 à 8859-10 (voir aussi la table 4).

2. On n'y trouve pas les deux caractères œ et Œ. On raconte que lorsque l'ISO a adopté cette norme, le représentant français était malade et qu'un autre pays en a profité pour imposer un caractère à lui (thorn?) prétextant que Œ pouvait s'obtenir par crénage de O et de E (ce qui est faux : voir [5]). Marti [50] signale qu'il est probable que ceci vient plutôt d'un certain désintérêt des Français en matière de normalisation au contraire des Allemands qui eux ont bien tous leurs *umlaut* et le ß. Mais œ et Œ figurent bien dans UNICODE (section 5).

3. Les 32 premiers caractères sont, comme pour l'ASCII, des caractères « de contrôle », en général peu utilisés. D'où la tendance de nombreux organismes à récupérer ces positions pour y mettre, au moins dans les normes de codage des fontes, des caractères manquants. C'est ainsi que le codage de TEX dit de Cork y place les ij hollandais et nos œ français (voir table 9).

Beaucoup de codages, filtres, logiciels, etc. ont été écrits pour échanger des textes écrits en ISOLATIN-1 en utilisant des réseaux où seul ASCII est connu ou pour passer d'un codage à l'autre[15].

5. UNICODE et ISO-10646

Les pays de l'Asie de l'est ont également fait l'inventaire de tous leurs caractères respectifs (voir [50]). Ainsi la Chine [40], le Japon [38, 39], la Corée [42, 43] et Taiwan [12] ont développé des normes multi-octets nationales qui contiennent plusieurs dizaines de milliers de caractères (en tout plus de cent vingt mille). Comme avec les codages ISO/IEC 8859 et EBCDIC, il est difficile de transformer des documents codés dans une norme pour les traiter avec un logiciel qui utilise une autre norme de codage de l'information.

ISO [37] et le Consortium UNICODE [62], un groupement de constructeurs d'ordinateurs, ont développé conjointement un jeu de caractères multinational regroupant la majeure partie des caractères utilisés dans le monde [8].

5.1. Le jeu de caractères universel ISO (UCS)

ISO a publié la norme ISO/IEC 10646 (noter les trois derniers chiffres 646, qui ont été choisis pour correspondre à ceux de l'ancienne norme ASCII). Cette norme propose des représentations de codages à deux ou quatre octets. Le Consortium UNICODE a nommé sa norme UNICODE. C'est un sous-ensemble 16-bits (deux-octets) de ISO/IEC 10646 avec les deux octets les plus significatifs égaux à zéro et correspond au plan multi-lingue de base (BMP, *Basic Multilingual Plane*).

[15] Voir notamment : `ftp://ftp.vlsivie.tuwien.ac.at/pub/8bit/FAQ-ISO-8859-1`, les *Newsgroups:* soc.culture.french, soc.culture.quebec, soc.culture.belgium, can.francais, fr.news.8bits, etc. ; ou les échanges sur le réseau de l'Association GUTenberg.

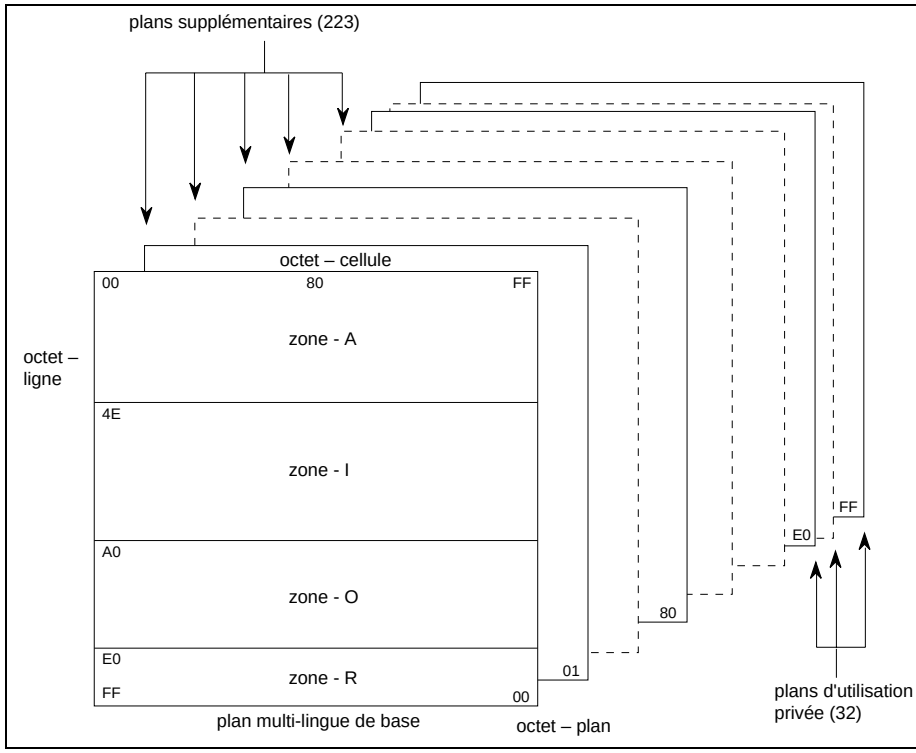


Figure 5 – Structure d'un groupe ISO/IEC10646 qui consiste en 256 plans.

En fait, la norme ISO/IEC 10646 peut être considérée comme un codage tri-dimensionnel de caractères, qui consiste en 256 groupes avec chacun 256 plans, le BMP correspondant au plan 0 du groupe 0 (voir figure 5).

Le codage ISO/IEC 10646 s'effectue sur quatre octets, qui sont appelés en commençant par le moins significatif, la *cellule* (octet-C), la *ligne* ou le *rang* (octet R), le *plan* (octet-P) et le *groupe* (octet-G) ; voir figure 6.

Cette forme canonique utilise donc un codage à quatre dimensions, qui consiste en 256 groupes tri-dimensionnels; chaque groupe consiste en 256 plans bi-dimensionnels et chaque plan a 256 lignes uni-dimensionnelles, avec chacune 256 cellules. Ces quatre octets sont nécessaires pour coder le jeu complet de tous les caractères du monde avec leurs variantes où les 16 bits (deux octets) du BMP (et d'UNICODE) ne suffisent pas.

5.2. Le Consortium UNICODE

Après quelques années de travaux préparatoires par Apple et Xerox, qui tous deux avaient déjà une longue expérience dans le domaine de la gestion de grands jeux de caractères et de l'internationalisation, le Consortium UNICODE fut créé en 1989. Il comprend

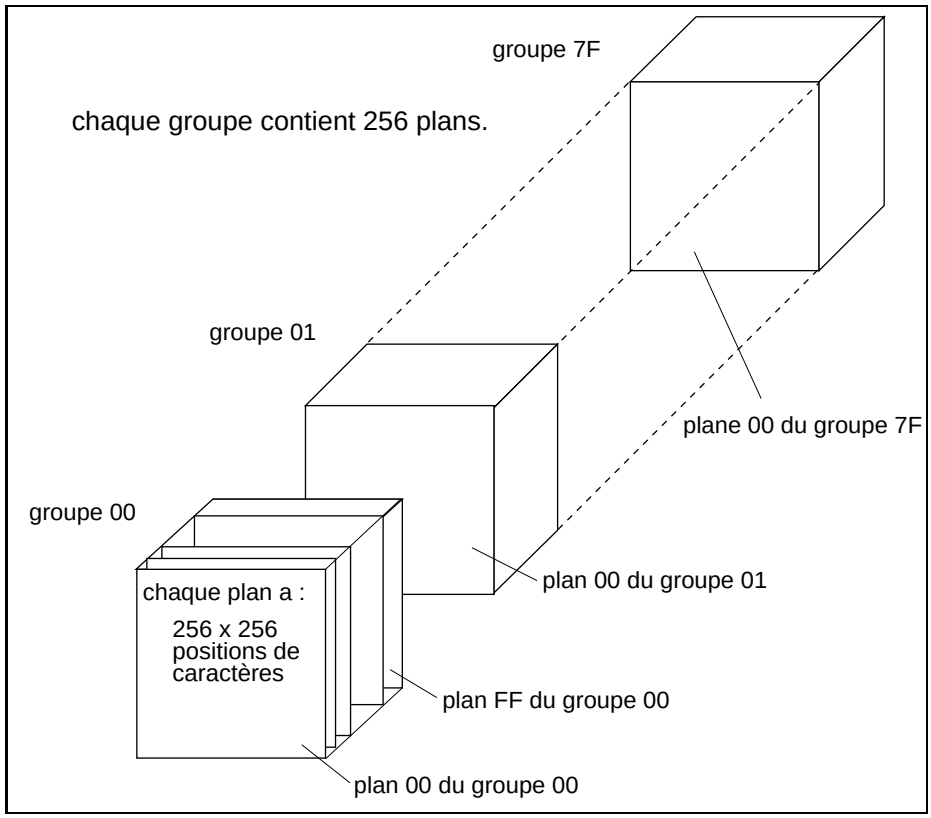


Figure 6 – L'espace de codage UCS.

plusieurs des acteurs les plus importants dans le monde de l'informatique et du logiciel, dont Adobe, Apple, IBM, Microsoft, Sun et Xerox.

Leurs travaux ont montré que la plupart des langues vivantes actuelles et même quelques-unes des langues anciennes peuvent être codées de façon non-ambiguë en utilisant seulement deux octets (16 bits). Ainsi, en accord avec ISO, il fut décidé en 1991 d'inclure ce codage UNICODE à 16 bits comme BMP (groupe 0, plan 0) dans l'espace de codage d'ISO/IEC 10646 (figure 10). On trouvera des informations sur UNICODE et une description des ressources offertes par le Consortium sur l'Internet sur WWW[16] (voir les figures 7 et 8).

Le but d'Unicode [62] est de fournir un codage non-ambigu, fixe sur 16 bits (deux octets), qui n'a besoin ni de séquences de contrôle, ni de méthodes de compactage. Il doit permettre l'échange, le traitement et la visualisation des caractères du monde entier. UNICODE est en quelque sorte une généralisation à double largeur d'ASCII.

[16] La page d'accueil WWW d'Unicode est à l'URL `http://www.stonehand.com/unicode.html`.

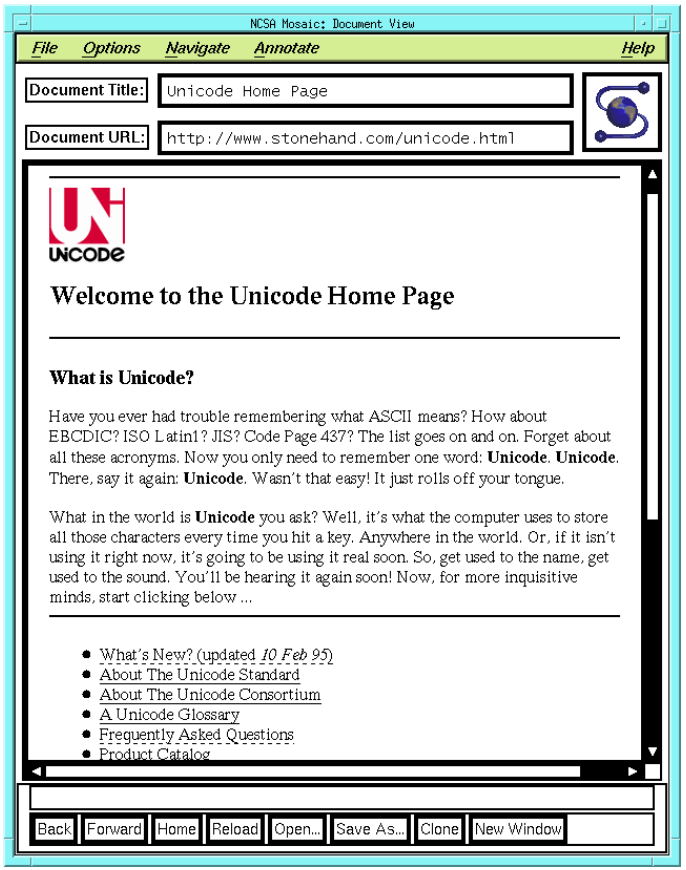


Figure 7 – Page d'accueil WWW de l'organisation UNICODE.

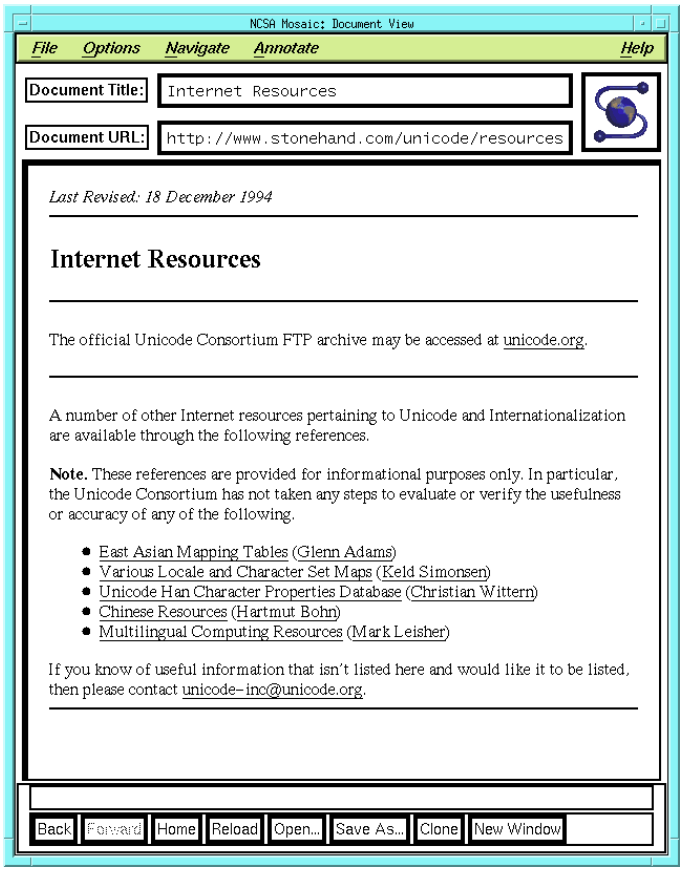


Figure 8 – Ressources Internet offertes par UNICODE.

Table 6 – Alphabets absents de la version actuelle d'UNICODE

Birman	Cree	Khmer (cambodgien)	Maldève (Dihevi)
Mongole	Moso (Naxi)	Pahawh Hmong	Rong (Lepcha)
Cingalais (Sri Lanka)	Tai Lu	Tai Mau	Tibétain
Tifinagh (Berbère)	Yi (Lolo)		

UNICODE permet le codage d'écritures, plutôt que de langues individuelles et il est conçu de telle façon que lorsque plusieurs langues ont en commun des caractères avec une dérivation historique commune, ou des apparences identiques ou très proches, la réunion du répertoire de tous ces caractères de chacune de ces langues est *unifiée* dans un jeu unique identifié comme un système d'écriture unique. Ces symboles unifiés seront alors utilisés pour écrire des langues spécifiques.

Actuellement UNICODE contient 34168 codes distincts pris dans 24 systèmes d'écritures différents (voir figure 10), qui couvrent les principales langues du monde. Il reste néanmoins quelques systèmes d'écriture qui ne sont que partiellement ou pas du tout représentés dans la version actuelle d'Unicode (voir table 6). Leur codage nécessite encore des recherches, mais rien n'empêchera de les inclure à un stade ultérieur.

5.3. Plan UNICODE/BMP

Le BMP (qui coïncide avec le plan UNICODE) est le seul plan qui soit actuellement défini pour ISO/IEC 10646. Sa structure est montrée en figure 9.

On trouvera plus de détails en figure 10. Le premier octet reproduit le codage ISO 8859-1, ce qui garantit une compatibilité avec ASCII et LATIN-1. Les parties supérieures comprennent d'autres *alphabets* et quelques symboles. C'est ce qui correspond à la *zone A*, qui a réservé 19 903 positions pour les systèmes d'écriture alphabétique et syllabiques, dont la partie inférieure contient également une partie phonétique pour le chinois/japonais/coréen/taiwanais avec le hiragana et le katakana, le bopomofo et le hangul.

Suit la *zone I* avec 20 992 positions, contenant des idéographes d'origine chinoise pour les langues chinoise, japonaise et coréenne. Pour limiter le nombre de signes et pour simplifier le traitement des données l'espace de codage pour les idéogrammes a été codé en utilisant un schéma appelé « unification Han ». Ce travail, une collaboration entre des groupes d'experts des pays directement impliqués et des États-Unis, était soumis aux mêmes règles très strictes que celles utilisées pour les normes japonaises [47]. Cette unifacation a permis d'éliminer plus de 99 000 signes communs au répertoire des 120 000 signes que compte l'union des jeux de caractères des normes de ces trois langues combinées. De la sorte le nombre de caractères Han a pu être ramené au chiffre indiqué. Ce principe est utilisé avec succès depuis plus de dix ans par le CCITT et le *East Asean Character Code*.

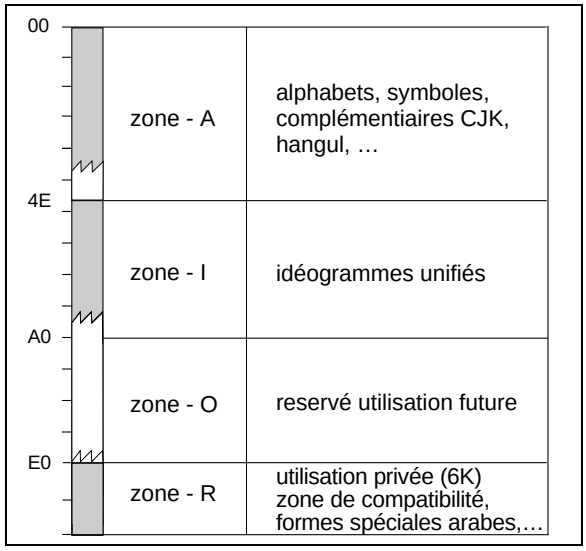


Figure 9 – Structure du BMP de ISO/IEC10646

Finalement on trouve la *zone O* comptant 16 384 positions, qui est réservée pour des extensions (normatives) futures, et la *zone R* avec 8 190 positions. Cette dernière zone, également appelée zone d'*utilisation restreinte*, contient des extensions utilisateurs, une zone de compatibilité et des formes de présentation et d'autres caractères spéciaux.

Le nom de tous les caractères de la partie non-Han est disponible sur l'Internet[17].

On trouvera les glyphes des 1700 premiers caractères dans une planche de [4] et dans trois tables de l'article de Bigelow dans ce *Cahier GUTenberg* [9].

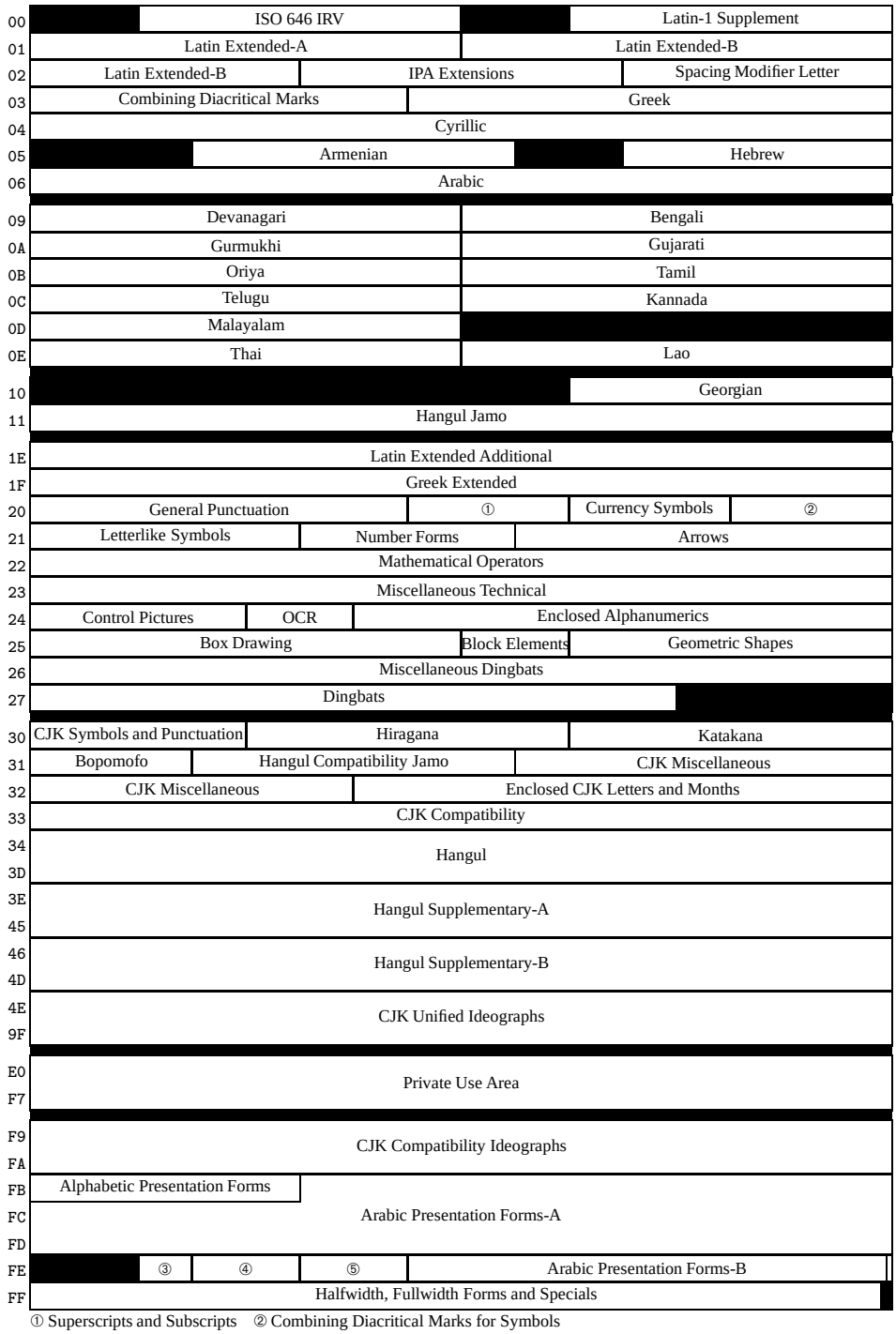
5.4. Comparaison UNICODE et ISO/IEC 10646

5.4.1. Largeur du codage

1. UNICODE est toujours codé sur deux octets (16 bits).
2. ISO/IEC 10646 définit un ensemble universel de codage de caractères sur plusieurs octets (*Universal Multiple-Octet Coded Character Set*), plus précisément sur quatre (UCS-4) ou sur deux octets (UCS-2). Dans ce dernier cas on se réfère au plan multi-lingue de base (*Basic Multilingual Plane* ou BMP). Le codage du BMP, qui est actuellement le seul plan défini, correspond à UNICODE.

Les octets les plus significatifs sont à gauche (ordre *Most Significant Byte* ou MSB, également appelé *Big Endian*). Plusieurs formats de transformation UCS (des UTF, pour

[17] URL `ftp://unicode.org/pub/MappingTables/UnicodeData-1.1.3.txt.Z`.



① Superscripts and Subscripts ② Combining Diacritical Marks for Symbols

③ Combining Half Marks ④ CJK Compatibility Forms ⑤ Small Form Variants

Figure 10 – Structure du plan UNICODE

UCS Transformation Format) ont été définis pour faciliter la transmission de données UCS.

UTF-1 Une transformation à 8-bits qui dans les octets n'utilise pas les caractères de contrôle spécifiés dans la norme ISO/IEC2022 [34] (c. à d. les parties C0, C1, SPACE, DEL, voir aussi la table 10). Ainsi il est possible de transmettre des données par des canaux sensibles à ces valeurs dans les octets.

UTF-7 Une transformation à 7-bits pour utilisation avec MIME (voir ci-dessous 7.4) et d'autres supports de transmission à sept bits.

UTF-8 Une version améliorée de UTF-1, qui transforme un flux de données codées en UNICODE ou ISO/IEC10646 en un flux 8 bits, préservant toute la partie ASCII, et en explosant les autres caractères en 3 octets (s'ils sont codés en 16 bits) ou en 6 octets (s'ils sont codés en 31 bits).

UTF-16/TF16 Une transformation pour obtenir UCS-4 à l'aide de codes UCS-2. Cette transformation préserve deux fois 1024 codes UCS-2, qui sont utilisés comme index pour représenter un million de caractères additionnels, ce qui devrait suffire pour coder tous les caractères chinois rares, ou les alphabets historiques, comme les hiéroglyphes égyptiens, pour lesquels il n'y avait plus de place dans le BMP.

5.4.2. Signes diacritiques

1. UNICODE définit des signes flottants explicitement pour le thaï, l'hébreu, l'arabe et les alphabets indiens. Il y a également un ensemble complet de signes diacritiques non-espaçant et des caractères composites pour les alphabets latins, grecs et cyrilliques, ce qui permet plusieurs représentations pour les lettres composites.

2. ISO/IEC 10646 a deux niveaux d'implantation pour combiner des caractères :

   – niveau 1 : impossibilité de signes de combinaison ;
   – niveau 2 : possibilité d'avoir des signes de combinaison ou des caractères precomposés.

   Des caractères de composition sont disponibles pour contrôler la présentation pour les flux de données bi-directionnelles (comme avec l'hébreu ou l'arabe).

La table 10 montre les symboles et les noms ISO/IEC 10646 pour les caractères de la norme ISO/IEC8859-1 (Latin-1). On les trouve dans le plan 0 d'UNICODE et du BMP d'ISO/IEC 10646.

5.5. Produits disponibles pour UNICODE

— SP, un compilateur SGML de la deuxième génération. Il fonctionne avec des jeux de caractères à seize bits, et en particulier le gestionnaire des entités accepte les codages UTF-8 et UCS-2 (UNICODE).

— *Progress First Software* vend des polices PostScript *Typographic International Series* compatibles avec UNICODE.

— Windows NT de Microsoft fonctionne directement avec UNICODE [25] et inclut une police « Lucida Sans UNICODE » qui contient quelques 1750 symboles (voir l'article de Bigelow dans ce *Cahier GUTenberg* [9]).

— Y & Y Inc. vendent des polices LATIN-2, Cyrillique et UNICODE (voir leur page d'accueil WWW à l'URL `http://www.yandy.com/`).

— X/Open a étudié l'internationalisation des logiciels et en particulier le support des différentes langues [22, 57].

— Le système plan 9 d'AT&T's offre un support de base pour UNICODE.

— Gamma Production commercialise plusieurs logiciels. *Gamma Server for* UNICODE permet une intégration du traitement multi-lingue à l'intérieur de Windows 3.1 ou NT pour toutes les langues UNICODE. *Gamma Unitype* est un utilitaire pour saisir directement plus de 175 langues différentes avec Windows 3.1 ou NT. *Gamma Universe for Windows* est un système multi-lingue UNICODE de traitement de texte avec des polices pour traiter la plupart des langues du monde. Pour toutes ces langues il inclut des vérificateurs d'orthographe et traite les ligatures complexes correctement.

— Le *Système C3* développé par l'Initiative Trans-Européenne (TERENA) offre une conversion entre les différents jeux de caractères[18].

— MASS (*Multilingual Application Support Service*) est un logiciel de développement pour créer des applications multi-lingues. Plusieurs langues et codages pour la saisie sont proposés[19] (voir figure 11).

— Ω, une extension 16-bit de TEX utilise UNICODE comme codage interne (voir [27] et l'article de HARALAMBOUS et PLAICE dans ce *Cahier GUTenberg* [26]).

— Une prochaine version de `Mule`, une extension multi-lingue 16-bits de Gnu `emacs` développée au Japon, ajoutera UNICODE à sa liste de codages possibles pour éditer des textes.

— Dans le monde SGML une initiative pour définir une syntaxe concrète de référence étendue est en cours de préparation. Cette proposition *ERCS (Extended Reference Concrete Syntax)* adresse l'inclusion de balises codées non seulement en ASCII, mais aussi en ISO/IEC10646. Ainsi, par exemple, les utilisateurs des pays de l'Asie du sud-est devraient être capables de saisir, éditer, baliser, sauvegarder et envoyer leurs documents SGML dans leur propre langue en utilisant un système de codage de leur choix[20].

[18] Plus d'information est à l'URL `http://www.nada.kth.se/i18n/c3/`.

[19] Plus d'information est à l'URL `http://www.iss.nus.sg/RND/MLP/Projects/MASS/MASS.html`.

[20] Voir la page d'accueil ERCS à l'URL `http://www.sgmlopen.org/sgml/docs/ercs/ercs-home.html`.

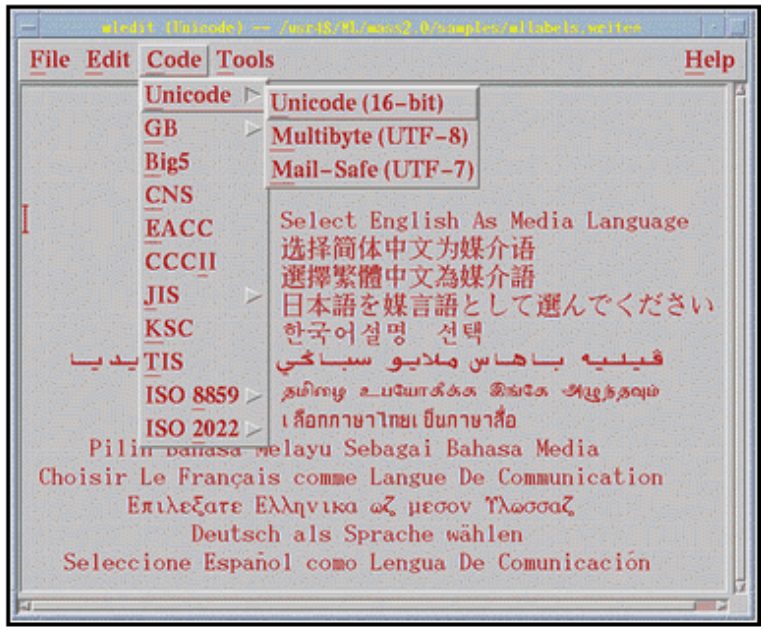


Figure 11 – L'éditeur MLEDIT (système MASS) montrant plusieurs codages.

— Dans le World Wide Web plusieurs initiatives pour visualiser des textes codés en différentes normes ont vu le jour, chacune utilisant une solution *ad hoc*. Récemment un collaborateur de la Commission des Communautés européennes à Bruxelles a proposé un certain nombre d'extension multi-lingues aux clients WWW : UNICODE deviendrait l'alphabet de base du Web et pour « savoir » quelle est la langue dans laquelle un document donné est rédigé, il introduit la notion d'hypertexte multi-lingue aligné (MAH, *Multilingual Aligned Hypertext*) qui permet un accès transparent à la version (langue) désirée en y associant une « étiquette » codée d'après la norme ISO 639, mentionnée au début de cet article[21] (voir la table 11 dans l'annexe).

6. Tables de codage des glyphes

Comme nous l'avons dit, chaque imprimante ou chaque photocomposeuse avait sa propre façon de coder les caractères (et même plus généralement de piloter les sorties, ce qui explique que des auteurs ou des constructeurs aient défini des langages de description de page, comme DVI *DeVice Independant* de TEX puis PostScript d'Adobe). On a donc tout naturellement tenté de normaliser ces codes, qu'ils soient à 6, 7, 16 ou 32 bits. On ne pouvait pas utiliser directement les normes d'échange qui n'ont pas la notion de

[21] Pour plus de détails voir l'URL `http://www.echo.lu/other/norm`.

Table 7 – Exemple d'accès aux fontes très riches de TEX par le seul code ASCII

Code (en ASCII)	glyphe	Code (en ASCII)	glyphe	Code (en ASCII)	glyphe
`\'e`	é	`\'E`	É	`\textsc{\'e}`	É
`\oe`	œ	`\AA`	Å	`\dag`	†
`\alpha`	α	`--`	–	`---`	—

glyphe et ne satisfont donc pas les besoins de la typographie : ni petites capitales, ni ligatures, ni les divers espaces ou tirets nécessaires, etc.

Mais il faut distinguer deux choses bien différentes :

– les caractères réellement présents dans une fonte ;

– la façon dont les formateurs s'en servent.

Par exemple, les ligatures « fi » et « fl » sont présentes dans 99% des polices de caractères commerciales, mais peu de formateurs sont capables, contrairement à TEX, de les imprimer (en fait de les « sélecter ») automatiquement[22].

Le principe est de coder les caractères d'une fonte en utilisant un ou plusieurs codes d'une ou plusieurs normes d'échange. TEX, par exemple, utilise des tables de codage très riches pour ses fontes. Mais on peut y accéder par les seuls 94 caractères du jeu de l'ASCII, un caractère, \, servant à préfixer les codes des autres (exemple table 7). Bien sûr, une saisie directe à 8 bits (comme par exemple avec `emacs` ou avec un système comme celui montré à la figure 11), est aussi possible tout comme divers systèmes de saisie directe dans une langue donnée ont été proposés (par exemple pour l'arabe).

TEX a probablement été le premier système à utiliser la notion de fonte telle qu'elle est répandue aujourd'hui (on verra à ce sujet [29, 41]), c'est-à-dire de base de donnée informatique avec une description de chaque caractère par ses contours, des possibilités de *hints* (adaptation à la grille) et des tables, dites TFM *(TEX Font Metric)*, fournissant les métriques aux utilisateurs (voir [14]). Mais le système adopté par PostScript étant le plus répandu, c'est celui que nous décrivons d'abord.

6.1. Le codage des fontes PostScript

Le codage des fontes PostScript repose sur plusieurs principes [1, 2, 3] :

[22] Le fait que les ligatures « ff », « ffi », etc. se trouvent — le cas échéant — dans des polices complémentaires (par exemple, *Times-Expert* est la police complémentaire de *Times-Roman*, et ainsi de suite) ne fait qu'aggraver la situation : ces pauvres formateurs commerciaux n'ayant pas la notion de *police virtuelle* de TEX, ou la possibilité de travailler en 16 bits, sont incapables de changer automatiquement de police en plein milieu d'un mot, sans intervention manuelle de l'utilisateur.

Table 8 – Dans les fontes traitées par PostScript, le choix d'un vecteur de codage permet d'associer à un code numérique le nom d'une procédure de tracé d'un caractère et par là d'adapter une fonte à son propre codage.

	Adobe standard	Apple Quick Draw	IBM EBCDIC	ISO LATIN-1	Adobe Symbol
	…	…	…	…	…
65	A	A		A	Alpha
66	B	B		B	Beta
67	C	C		C	Gamma
	…	…	…	…	…
193	grave (')	exclamdown (¡)	A	Aacute (Á)	Ifraktur
194	acute (')	logicalnot (¬)	B	Acircumflex (Â)	Rfraktur
	…	…	…	…	…

1. Lors de l'impression, une fonte comprend 256 caractères (et ce dans n'importe quel corps ou n'importe quelle direction).

2. Chaque caractère est en fait une procédure de tracé du glyphe correspondant ; cette procédure a un nom (par exemple `A` pour « A », `semicolon` pour« ; », `eacute` pour « é », etc.

3. Ces 256 procédures peuvent être choisies dans un ensemble beaucoup plus grand : le *Times romain* d'Adobe, par exemple, offre non seulement toutes les lettres accentuées du français, mais aussi celles du polonais, le symbole monétaire florin ou le « ž » (zcaron). Pour celà PostScript utilise un système de codage à deux temps[23] : une table de codage intermédiaire (dite *Encoding Vector*) comprenant 256 entrées permet d'indiquer quel (nom de) caractère correspond à tel code. Par exemple (table 8), il suffit, pour un formateur, d'utiliser le vecteur de codage correspondant à ISOLATIN-1 pour associer au code numérique 193 le nom de la procédure `Aacute` qui dessinera donc le glyphe « Á ». Certaines fontes, comme *Symbol* où se trouvent des caractères tels que α, $\Leftrightarrow$ ou $\int$, ont leur propre jeu de noms de procédures : au code 67 (table 8) on associera par exemple le nom `Gamma` pour dessiner le glyphe Γ.

4. ISOLATIN-1 occupant les 256 positions d'une table de codage, il est alors impossible d'y mettre les autres noms de glyphe (tels que « È » ou « ffi », voire comme 5/8). Le principe est alors d'utiliser plusieurs « fontes », c'est-à-dire de considérer qu'une fonte est formée de plusieurs morceaux, chacun avec son propre système de codage. Par exemple, le *Garamond romain* d'Adobe utilise les « fontes » suivantes :

[23] Pour être complet, signalons qu'un opérateur, `glyphshow` permet d'imprimer un glyphe en ne connaissant que son nom.

— `AGaramond-Regular` pour les caractères d'ISOLATIN-1,

— `AGaramondExp-Regular` pour les petites capitales[24] et quelques ligatures comme (« ffl » ...),

— `AGaramondAlt-Regular` pour des variantes, d'autres ligatures comme « st » ou des lettres finales.

Et de même pour les italiques, les gras, les capitales de titres, etc. La famille *Adobe-Garamond* est donc formée d'une vingtaine de fontes numériques.

Lors de la saisie d'un texte, il faut alors appeler une nouvelle fonte (en Word sur Macintosh par exemple, ceci revient à cliquer sur le nom de la fonte correspondante) chaque fois qu'une nouvelle table de codage doit être utilisée. C'est fastidieux, mais probablement pas plus que de changer de casse !

On trouvera dans ce *Cahier GUTenberg* quelques raisons sur le choix d'une fonte unique ou d'un ensemble de fontes pour offrir un tel jeu de caractères [9].

6.2. Sélection des fontes de (LA)TEX

(LA)TEX utilise, en gros, le même style de fontes que celles de PostScript (et éventuellement celles-là d'ailleurs). Mais, leur emploi est mieux défini et se veut plus général que dans les autres systèmes surtout depuis deux ou trois ans (notion de fonte virtuelle, *New Font Selection Scheme*, etc.). On trouvera dans [14, 21] les principaux textes sur ces mécanismes et y renvoyons le lecteur. En résumé, disons ici que LATEX distingue nettement (même si c'est souvent transparent à l'utilisateur normal) :

1. le contenu physique des fontes (tables de métrique par exemple) ;

2. la façon, pour un utilisateur, d'appeler une fonte ; pour les non-TEXistes, ceci se fait par l'intermédiaire de « style » ou classe de documents soit de façon automatique (mais programmable) – soit de façon explicite ;

3. la façon de faire entrer les fontes physiques d'une famille commerciale (par exemple *Adobe-Garamond*) dans ce système ;

4. la façon de coder les caractères en fonction d'une fonte donnée et, par exemple, de la langue : ce peut être à 7 bits (table 7), à 8 bits (table 9), voire sur 16 ou 32 bits (voir l'article sur Ω dans ce *Cahier GUTenberg* [26]).

5. L'extension `inputenc` de LATEX 2_ε offre la possibilité de spécifier un codage pour le source d'un document (voir section C dans l'annexe).

[24] Ce sont effectivement de vraies petites capitales et non des capitales réduites optiquement à un corps plus petit.

Table 9 – Le codage TEX EC

0	`	´	^	~	¨	˝	°	ˇ	˘	¯	·	¸	¿	‚	‹	›
16	“	”	„	«	»	–	—		ı	ȷ	ff	fi	fl	ffi	ffl	
32	␣	!	"	#	$	%	&	’	(	)	*	+	,	-	.	/
48	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
64	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
80	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
96	‘	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
112	p	q	r	s	t	u	v	w	x	y	z	{	|	}	~	-
128	Ă	Ą	Ć	Č	Ď	Ě	Ę	Ğ	Ĺ	Ľ	Ł	Ń	Ň	Ŋ	Ő	Ŕ
144	Ř	Ś	Š	Ş	Ť	Ţ	Ű	Ů	Ÿ	Ź	Ž	Ż	IJ	İ	đ	§
160	ă	ą	ć	č	ď	ě	ę	ğ	Í	ľ	ł	ń	ň	ŋ	ő	ŕ
176	ř	ś	š	ş	ť	ţ	ű	ů	ÿ	ź	ž	ż	ij	¡	¿	£
192	À	Á	Â	Ã	Ä	Å	Æ	Ç	È	É	Ê	Ë	Ì	Í	Î	Ï
208	Đ	Ñ	Ò	Ó	Ô	Õ	Ö	Œ	Ø	Ù	Ú	Û	Ü	Ý	Þ	SS
224	à	á	â	ã	ä	å	æ	ç	è	é	ê	ë	ì	í	î	ï
240	ð	ñ	ò	ó	ô	õ	ö	œ	ø	ù	ú	û	ü	ý	þ	ß

6.3. Codage « de Cork » TEX EC

Pour permettre aux utilisateurs de TEX de traiter le maximum de langues (utilisant l'alphabet latin) avec une seule police de 256 glyphes, un groupe de travail a développé, lors de la conférence EuroTEX à Cork (Irlande) en septembre 1990, un codage *européen* « EC » [17]. Ce codage est actuellement toujours connu sous le nom de « norme Cork », ou « norme DC », où DC est un précurseur « provisoire » — depuis bientôt cinq ans — de la forme définitive des polices EC.

Le principe a donc été de faire une table de codage de glyphes respectant ISOLATIN-1 mais récupérant les places de cette dernière correspondant à des caractères non-imprimables ou peu utilisés (comme μ).

La table 9 montre que les positions 32–136, 192–222 et 224–254 sont identiques à celles de LATIN-1. Par contre les autres positions sont occupées par des signes diacritiques (positions 00–31) ou des caractères pre-composés avec diacritiques (positions 128–191).

Une police codée d'après la norme EC permet de traiter la plupart des langues européennes utilisant l'alphabet latin, à l'opposé des normes ISO/IEC 8859, qui ont besoin de six codages différents (voir table 4), même si seulement LATIN-1, LATIN-2, LATIN-5 et LATIN-6 sont actuellement utilisés activement.

7. Autres normes

Terminons par quelques informations rapides sur d'autres « normes ».

7.1. Claviers

La normalisation des claviers tend actuellement à prendre deux voies [52].

1. La première est une tentative d'harmonisation des claviers existants (basés sur ceux des premières machines à écrire). Ceci concerne la norme ISO-8884 du 15 septembre 1989 qui permet l'entrée des caractères d'ISO-8859. Une nouvelle norme, ISO/IEC-9995, est pratiquement adoptée et est en fait composée de huit parties (ISO-9995-1 à ISO-995-8 ; parmi celles-ci deux sont particulièrement intéressantes, la partie 6 qui offre des fonctions pour « programmer » les claviers et surtout la partie 7 qui définit les symboles utilisés sur les claviers pour représenter ces fonctions) qui recouvrent les normes existantes précédemment et cherche à harmoniser les claviers diffusés sur le marché (harmoniser et non à résoudre le débat mythique du clavier unique !).

2. Les claviers du futur. Un groupe de travail de l'ISO (SWG K) étudie ce que pourrait être le clavier universel du futur. Au delà des problèmes de forme (clavier linéaire, en V, syllabique, etc.), les problèmes à résoudre relèvent encore de principes (interchangeabilité, modularité, etc.).

En tout cas le multilinguisme ce n'est pas uniquement l'échange des caractères, c'est aussi leur saisie et il est heureux de voir paraître des documents « drafts » comme *Input methods to enter characters of* ISO/IEC-*10646*[25].

7.2. EBCDIC

EBCDIC, *Extended Binary Coded Decimal Interchange Code*, est une norme privée d'IBM des années 1960. Basée au départ sur les codes Hollerith des cartes perforées (code H), une première norme définie par IBM, BDC, *Binary Coded Decimal*, permettait de traiter sur 4 bits les chiffres et les lettres. Selon le poids donné à chaque position binaire, on avait plusieurs codes BCD, par exemple 8421 (le chiffre 9 s'y écrivant alors 1001), 2421 (9=1111), 4121 (9=1100). Il en a aussi existé une version à 6 bits. Avec l'apparition du 360 et des « octets », IBM a défini, vers 1965, cette extension basée sur le même principe qu'ASCII : une plage internationale (c'est-à-dire avec les seuls caractères anglais), et une plage pour des variantes nationales ou … matérielles. Car c'est là sans doute la principale différence entre EBCDIC et ISOLATIN : cette première est à la fois une norme d'échange et une norme de saisie ou de fontes.

[25] URL `ftp://ftp.uni.erlangen.de/pub/doc/ISO/charsets/ucs-input-methods`

En effet, ces codages 8-bits EBCDIC, qui étaient adaptés aux besoins des utilisateurs des gros et moyens systèmes IBM à travers le monde, donnent les correspondances entre les différents claviers d'ordinateurs nationaux, qui sont exprimées à l'aide de « code pages ». Ces différents « code pages » sont une source continue de confusion quand on veut utiliser un texte préparé avec un « code page » donné (par ex. 385 pour le français) pour le visualiser ou l'imprimer sur une machine qui en utilise un autre (par ex. 382 pour l'allemand). Actuellement plusieurs dizaines de code pages EBCDIC sont utilisés à travers le monde[26].

7.3. Et les autres…

Similairement, pour les ordinateurs personnels des douzaines de codages différents sont utilisés sur les PC sur tous les continents ; de plus, treize codages différents existent pour le Mac [7].

7.4. À suivre ?

Ainsi que nous le disions dans notre introduction (page 3) nous nous sommes concentrés ici sur les normes au sens « Institut de normalisation ». Au delà de cet aspect normatif, il reste tout un côté « utilisation » que nous n'avons pas abordé : ce sera l'objet d'un prochain *Cahier GUTenberg* pour lequel nous faisons dès à présent un appel à soumission d'articles. En particulier nous recherchons des articles sur :

— Mime[27].

— les problèmes de codage à l'entrée (`inputenc`) ;

— la représentation interne (Ω par exemple utilise UNICODE) et les liens avec les césures ;

— le codages des polices (au niveau TEX–fontenc, puis le vecteur de codage PostScript–`encodingvector` ...) ;

— les moteurs TEX version 3 et leur relation à ces problèmes (ML-TEX, TEX–XET, Ω) ;

— etc.

Au congrés GUTenberg GUT95, qui s'est tenu les 1 et 2 juin à la Grande Motte il fut decidé de créer un groupe de reflexion sur les problèmes de codages. Toute personne interessée aux travaux de ce groupe ou desirant d'y participer est priée de contacter la redaction ou le bureau de l'Association GUTenberg.

[26] Voir par exemple `ftp://dkuug.dk/i18n/charmaps`, `ftp://unicode.org/pub/MappingTables/WindowsMaps`

[27] Voir `ftp://ftp.vlsivie.tuwien.ac.at/pub/8bit/FAQ-ISO-8859-1` ; voir aussi `comp.mail.mime` : *Multipurpose Internet Mail Extensions*

8. Conclusion

Nous l'avons vu, la plus grande anarchie a longtemps régné dans les codages de caractères, de glyphes, etc. soit parce que ces normes étaient inadaptées aux diverses langues, soit que les constructeurs n'en faisaient qu'à leur tête ! UNICODE/ISO-10646 pourrait n'être à son tour qu'une n+1^{e} norme. Nous pensons plutôt que ce sera « la » norme du futur proche ! En effet, cette norme a pour elle deux atouts majeurs :

1. elle est issue d'un groupe de vendeurs de matériels qui sont les premiers à se plaindre de la multiplicité des divers codages ;
2. c'est une norme ISO, cet organisme ayant aussi développé en parallèle d'autres normes qui lui sont liées, comme celles sur les claviers.

Tout est donc fait pour qu'elle s'impose. Les recherches faites actuellement autour d'elle (inclusion d'UNICODE dans WWW, dessin de fontes aussi grosses, etc.) sont pour nous la preuve d'un grand engouement pour cette norme et le garant de sa future réussite. Rendez-vous au prochain millénaire ?

Bibliographie

[1] Adobe Systems Incorporated, *Adobe Type 1 Font Format*, version 1.1, Addison-Wesley Publishing Company, Reading, MA (USA), 1990.

[2] Adobe Systems Incorporated, *PostScript Language Reference Manual*, Addison-Wesley Publishing Company, Reading, MA (USA), 2nd edition, 1991.

[3] Jacques ANDRÉ et Justin BUR, « Métrique de fontes PostScript », *Cahiers GUTenberg*, n° 8, mars 1991, 29–50.

[4] Jacques ANDRÉ, « Unicode – une casse de 38 000 signes », *Caractères*, n° 373, 1994, p. 32–36.

[5] J. ANDRÉ et A. WILD, Ligatures, typographie et informatique, *Rapports de recherche*, Inria, n° 2429, décembre 1994.

[6] Jacques ANIS (éditeur), « Écritures », *Linx*, numéro spécial 31, Nanterre, 1994.

[7] Apple Computer Inc., *Guide to Macintosh Software Localization,* Addison Wesley, 1992.

[8] Jürgen BETTELS et F. Avery Bishop, Unicode: A Universal Character Code, *Digital technical Journal*, 5(3):21–31, Summer 1993
(disponible à l'URL `http://www.digital.com/info/DTJ/i18n-toc.html`).

[9] Chuck BIGELOW and Kris HOLMES, « The design of a UNICODE font », *EPODD, Electronic Publishing, Origination, Dissemination and Design*, vol. 6(3), september 1993 (actes de RIDT'94 *Raster Imaging and Digital Typography*), 289–305. Traduit en français « Création d'une police UNICODE », *Cahiers GUTenberg*, n° 20 (ce cahier), avril 1995,

[10] Gérard BLANCHARD, « Nœuds & esperluettes – actualités et pérennité d'un signe », *Communication et langages*, n° 92, 1992, p. 85–101.

[11] Florian CAJORI, *A History of Mathematical Notations, Volume II: Notations Mainly in Higher Mathematics*, Open Court Press, 1929 (reprinted in 1952).

[12] Chinese National Standard, CNS 11643-1986, Taipeh, 1986.

[13] Marcel COHEN, *La grande invention de l'écriture et de son évolution,* Imprimerie nationale, Paris, 1958.

[14] Alain COUSQUER et Éric PICHERAL, « Polices, TEX et Cie. », *Cahiers GUTenberg*, n° 9, juillet 1992, 3–31.

[15] David DIRINGER, *The Alphabet. A Key to the History of Mankind (Third Edition),* Hutchinson of London, 1968.

[16] John DREYFUS et François RICHAUDEAU, *La chose imprimée*, Éditions Retz, Paris, 1977.

[17] Michael J. FERGUSON, « Fontes latines européennes et TEX 3.0 », *Cahiers GUTenberg* n° 7, novembre 1990, p. 29–31.

[18] James G. FÉVRIER, *Histoire de l'écriture,* Payot, Paris, 1984.

[19] R. S. GILIAREVKI et V. S. GRIVNIN, *Determining a language by its alphabet (en russe),* Vostotshnaya literatura, Moskva, 1960.

[20] Michel GOOSSENS et Erik VAN HERWIJNEN. « Introduction à SGML, DSSSL et SPDL. » *Cahiers GUTenberg*, n° 12, décembre 1991, 37–56.

[21] Michel GOOSSENS, Frank MITTELBACH et Alexander SAMARIN, *The LaTeX Companion,* Addison-Wesley, 1994.

[22] Timothy G. GREENWOOD, International Cultural Differences In Software, *Digital technical Journal*, 5(3):32–43, Summer 1993
(disponible à l'URL `http://www.digital.com/info/DTJ/i18n-toc.html`).

[23] Maurice GREVISSE, *Le bon usage,* Duculot, 1986.

[24] Barbara F. GRIMES, *Languages of the World, 11th edition,* Summer Institute of Linguistics, May 1988, Dallas, Texas.

[25] William S. HALL, Internationalization in Windows, Part I: Programming with UNICODE, *Microsoft Systems Journal*, 58–71, juin 1994.

[26] Yannis HARALAMBOUS et John PLAICE, « Ω, une extension de TEX incluant UNICODE et des filtres de type Lex ». Ce *Cahier GUTenberg*, n° 20, avril 1995, pages 55–79.

[27] Yannis HARALAMBOUS et John PLAICE, « Ω + Virtual METAFONT = UNICODE + Typography ». Présenté à la Journée Ω à Genève le 16 mars 1995. À paraître dans *TUGboat*.

[28] F. HAYES, Consortium forms Universal Character Code Standard, *UnixWorld*, 8:109–110, May 1991.

[29] Roger HERSCH (ed.), *Visual and Technical Aspects of Types,* Cambridge University Press, 1993.

[30] Imprimerie nationale, *Les caractères de l'Imprimerie nationale,* Imprimerie nationale Editions, Paris, 1990.

[31] International Organization for Standardization, *Code pour la représentation des noms de langue,* ISO 639:1988 (E/F), ISO Genève, 1988.

[32] International Organization for Standardization, *Committee Draft: Terminology–Code for the representation of names of languages, Part 2: Alpha-3 code,* ISO CD 639-2, ISO Geneva, 1993.

[33] International Organization for Standardization, *Information technology–ISO 7-bit coded character set for information interchange,* ISO/IEC 646:1991, ISO Geneva, 1991.

[34] International Organization for Standardization, *Information technology–Character code structure and extension techniques,* ISO/IEC 2022:1994, ISO Geneva, 1994.

[35] International Organization for Standardization, *Information technology–Control functions for coded character sets,* ISO/IEC 6429:1992, ISO Geneva, 1992.

[36] International Organization for Standardization, *Information technology–8-bit single-byte coded graphic character set–Parts 1 to 10,* ISO/IEC 8859-1:1987 to ISO/IEC8859-10:1992, Geneva, 1987–92.

[37] International Organization for Standardization, *Information technology–Universal Multiple-Octet Coded Character Set (UCS)–Part1: Architecture and Basic Multilingual Plane*, ISO/IEC 10646-1:1993, Geneva, 1993.

[38] Japanese Standards Association, *Jouhou koukan you kanji fugoukei (Code of the Japanese Graphic Character Set for Information Interchange),* JIS X 208-1990, Tokyo, 1990.

[39] Japanese Standards Association, *Jouhou koukan you kanji fugou-hojo kanji (Code of the supplementary Japanese Graphic Character Set for Information Interchange),* JIS X 212-1990, Tokyo, 1990.

[40] JISHU Biaozhun Chubanshe (Technical Standards Publishing), *Code of Chinese Graphic Character Set for Information Interchange* GB 2312-1980, Beijing, 1980.

[41] Peter KAROW, *Schrifttechnologie – Methoden un Werkzeuge*, Springer-Verlag, 1993 ; et (en anglais) *Font Technology*, URW Verlag, Hambourg, 1994.

[42] Korean Industrial Standards Association, *Jeongho gyohwanyong buho (Hangul mit Hanja) (Code for Information Interchange (Hangul mit Hanja)),* KS C 5601-1987, Seoul, 1987.

[43] Korean Industrial Standards Association, *Jeongho gyohwanyong buho hwakjang saten (Code of the supplementary Korean graphic character set for Information Interchange),* KS C 5657-1991, Seoul, 1991.

[44] M.Y. KSAR, Untying Tongues, *ISO Bulletin*, 24(6):2–8, June 1993.

[45] Ahmed LAKHDAR-GHAZAL, *Le système Lakhdar-Ghazal de composition arabe standard, Cahier GUTenberg* « TEX et l'écriture arabe », *à paraître.*

[46] Tracy LAQUEY, *Sésame pour l'Internet – Initiation au réseau planétaire,* Addison-Wesley France, Paris, 1994.

[47] Ken LUNDE, *Understanding Japanese Information Processing,* O'Reilly & Associates, Inc. 1993.

[48] Michel MALHERBE, *Les langages de l'humanité,* Seghers, Paris 1983.

[49] Alan MARSHALL, *Ruptures et continuités dans un changement de système technique – le remplacement du plomb par la lumière dans la composition typographique*, thèse, Grenoble, 18 décembre 1991. Parue comme *Publication interne Irisa*, n° 638, mars 1992.

[50] Bernard MARTI et co-auteurs, *Télématique – techniques, normes, services*, Dunod, 1990.

[51] Roland MEYNET, *L'écriture arabe en question*, Publications du Centre Culturel Universitaire, Dar El-Machreq Éditeurs, Beyrouth, 1971.

[52] Yves NEUVILLE, « Normalisation prospective des claviers et multi-linguisme », *Actes du colloque Lexipraxi 90*, AILF, Paris, novembre 1990.

[53] Ch. PETZOLD, Move Over, ASCII! UNICODE Is Here, *PC Magazine*, 12(18):374–376, October 1993.

[54] Ch. PETZOLD, Unicode, wide characters, and C, *PC Magazine*, 12(19):369–376, November 1993.

[55] Ch. PETZOLD, Viewing a UNICODE TrueType font under Windows NT, *PC Magazine*, 12(20):379–390, November 1993.

[56] Ch. PETZOLD, Typing UNICODE characters from the keyboard, *PC Magazine*, 12(21):426–444, December 1993.

[57] Wendy RANNENBERG and Jürgen BETTELS, The X/Open Internationalization Model, *Digital technical Journal*, 5(3):32–43, Summer 1993
(disponible à l'URL `http://www.digital.com/info/DTJ/i18n-toc.html`).

[58] K.M. SHELDON, ASCII goes global, *Byte*, 16:108–116, July 1991.

[59] R. SOUTHALL, « Towards the present day font », A. Marshall (ed.) *Actes du colloque « La Lumitype-Photon », Lyon, octobre 1994*, à paraître.

[60] Ian TSCHICHOLD, *Formen Wandlungen der &-zeichen*, D. Stempel AG, Francfort. Traduction française de René Grasset à paraître à l'École Estienne.

[61] B. L. ULLMAN, *Ancient writing and its influence*, Cooper Square Publishers, Inc., New York, 1963.

[62] The Unicode Consortium, *The* UNICODE *Standard: Worldwide Character Encoding*, Version 1.0, Volumes 1 and 2. Addison Wesley, 1991/92.

Annexes

A. Les noms des caractères de la norme ISOLATIN-1

Table 10: Noms normatifs (anglais/français) ISO/IEC10646 des caractères de la norme ISO/IEC8859-1.

Les caractères de contrôle C0– nommés d'après leurs acronymes ISO 646[28]		
(les noms entre parenthèses sont ceux de la norme POSIX)		
<NUL> or <NU>	000	NULL NUL
<SOH> or <SH>	001	START OF HEADING DÉBUT D'EN-TÊTE
<STX> or <SX>	002	START OF TEXT DÉBUT DE TEXTE
<ETX> or <EX>	003	END OF TEXT FIN DE TEXTE
<EOT> or <ET>	004	END OF TRANSMISSION FIN DE TRANSMISSION
<ENQ> or <EQ>	005	ENQUIRY DEMANDE
<ACK> or <AK>	006	ACKNOWLEDGE ACCUSÉ DE RÉCEPTION
<BEL> or <BL> (<alert>)	007	BELL SONNERIE
<BS> (<backspace>)	008	BACKSPACE ESPACE ARRIÈRE
<HT> (<tab>)	009	HORIZONTAL TABULATION TABULATION HORIZONTALE
<LF> (<newline>)	010	LINE FEED INTERLIGNE
<VT> (<vertical-tab>)	011	VERTICAL TABULATION TABULATION VERTICALE
<FF> (<form-feed>)	012	FORM FEED PRÉSENTATION DE FEUILLE
<CR> (<carriage-return>)	013	CARRIAGE RETURN RETOUR DE CHARIOT
<SO>	014	SHIFT OUT HORS-CODE (CODE SPÉCIAL)
<SI>	015	SHIFT IN EN-CODE (CODE NORMAL)
<DLE> or <DL>	016	DATALINK ESCAPE ÉCHAPPEMENT À LA TRANSMISSION
<DC1> or <D1>	017	DEVICE CONTROL ONE CONTRÔLE DE PÉRIPHÉRIQUE UN
<DC2> or <D2>	018	DEVICE CONTROL TWO CONTRÔLE DE PÉRIPHÉRIQUE DEUX
<DC3> or <D3>	019	DEVICE CONTROL THREE CONTRÔLE DE PÉRIPHÉRIQUE TROIS
<DC4> or <D4>	020	DEVICE CONTROL FOUR CONTRÔLE DE PÉRIPHÉRIQUE QUATRE
<NAK> or <NK>	021	NEGATIVE ACKNOWLEDGE ACCUSÉ DE RÉCEPTION NÉGATIF
<SYN> or <SY>	022	SYNCHRONOUS IDLE SYNCHRONISATION

[28] Les termes français sont ceux donnés dans le *Dictionnaire de l'anglais de l'informatique* de Jacques HILDEBERT, Presses Pocket, 1992, car il n'existe pas de traduction française officielle ni dans ISO646, ni, pour ces caractères dans ISO8859(F).

Table 10: Caractères ISO/IEC8859-1 (suite)

<ETB> or <EB>	023		END OF TRANSMISSION BLOCK FIN DE BLOC DE TRANSMISSION
<CAN> or <CN>	024		CANCEL ANNULATION
<EM>	025		END OF MEDIUM FIN DE SUPPORT
<SUB> or <SB>	026		SUBSTITUTE SUBSTITUTION
<ESC> or <EC>	027		ESCAPE ÉCHAPPEMENT
<IS4> or <FS>	028		FILE SEPARATOR SÉPARATEUR DE FICHIERS
<IS3> or <GS>	029		GROUP SEPARATOR SÉPARATEUR DE GROUPES DE DONNÉES
<IS2> or <RS>	030		RECORD SEPARATOR SÉPARATEUR D'ENREGISTREMENTS
<IS1> or <US>	031		UNIT SEPARATOR SÉPARATEUR DE SOUS-ARTICLES
Caractères imprimables[29]			
<space>	032		SPACE ESPACE
<exclamation-mark>	033	!	EXCLAMATION MARK POINT D'EXCLAMATION
<quotation-mark>	034	"	QUOTATION MARK GUILLEMET
<number-sign>	035	#	NUMBER SIGN SYMBOLE NUMÉRO
<dollar-sign>	036	$	DOLLAR SIGN SYMBOLE DOLLAR
<percent>	037	%	PERCENT SIGN SYMBOLE POURCENT
<ampersand>	038	&	AMPERSAND PERLUÈTE
<apostrophe>	039	'	APOSTROPHE APOSTROPHE
<left-parenthesis>	040	(	LEFT PARENTHESIS PARENTHÈSE GAUCHE
<right-parenthesis>	041	)	RIGHT PARANTHESIS PARENTHÈSE DROITE
<asterisk>	042	*	ASTERISK ASTÉRISQUE
<plus-sign>	043	+	PLUS SIGN SIGNE PLUS
<comma>	044	,	COMMA VIRGULE
<hyphen>	045	–	HYPHEN-MINUS TIRET, SIGNE MOINS
<period>	046	.	FULL STOP POINT
<slash>	047	/	SOLIDUS BARRE OBLIQUE
<zero>	048	0	DIGIT ZERO CHIFFRE ZÉRO
<one>	049	1	DIGIT ONE CHIFFRE UN
<two>	050	2	DIGIT TWO CHIFFRE DEUX
<three>	051	3	DIGIT THREE CHIFFRE TROIS

[29] Traduction française de ISO8859-1(F) du 15 février 1987.

Table 10: Caractères ISO/IEC8859-1 (suite)

<four>	052	4	DIGIT FOUR CHIFFRE QUATRE
<five>	053	5	DIGIT FIVE CHIFFRE CINQ
<six>	054	6	DIGIT SIX CHIFFRE SIX
<seven>	055	7	DIGIT SEVEN CHIFFRE SEPT
<eight>	056	8	DIGIT EIGHT CHIFFRE HUIT
<nine>	057	9	DIGIT NINE CHIFFRE NEUF
<colon>	058	:	COLON DEUX POINTS
<semicolon>	059	;	SEMICOLON POINT VIRGULE
<less-than>	060	<	LESS-THAN SIGN SIGNE INFÉRIEUR ‘
<equals-sign>	061	=	EQUALS SIGN SIGNE ÉGAL À
<greater-than>	062	>	GREATER-THAN SIGN SIGNE SUPÉRIEUR À
<question-mark>	063	?	QUESTION MARK POINT D’INTERROGATION
<commercial-at>	064	@	COMMERCIAL AT À COMMERCIAL
<A>	065	A	LATIN CAPITAL LETTER A LETTRE MAJUSCULE A
<B>	066	B	LATIN CAPITAL LETTER B LETTRE MAJUSCULE B
<C>	067	C	LATIN CAPITAL LETTER C LETTRE MAJUSCULE C
<D>	068	D	LATIN CAPITAL LETTER D LETTRE MAJUSCULE D
<E>	069	E	LATIN CAPITAL LETTER E LETTRE MAJUSCULE E
<F>	070	F	LATIN CAPITAL LETTER F LETTRE MAJUSCULE F
<G>	071	G	LATIN CAPITAL LETTER G LETTRE MAJUSCULE G
<H>	072	H	LATIN CAPITAL LETTER H LETTRE MAJUSCULE H
<I>	073	I	LATIN CAPITAL LETTER I LETTRE MAJUSCULE I
<J>	074	J	LATIN CAPITAL LETTER J LETTRE MAJUSCULE J
<K>	075	K	LATIN CAPITAL LETTER K LETTRE MAJUSCULE K
<L>	076	L	LATIN CAPITAL LETTER L LETTRE MAJUSCULE L
<M>	077	M	LATIN CAPITAL LETTER M LETTRE MAJUSCULE M
<N>	078	N	LATIN CAPITAL LETTER N LETTRE MAJUSCULE N
<O>	079	O	LATIN CAPITAL LETTER O LETTRE MAJUSCULE O
<P>	080	P	LATIN CAPITAL LETTER P LETTRE MAJUSCULE P
<Q>	081	Q	LATIN CAPITAL LETTER Q LETTRE MAJUSCULE Q
<R>	082	R	LATIN CAPITAL LETTER R LETTRE MAJUSCULE R

Table 10: Caractères ISO/IEC8859-1 (suite)

<S>	083	S	LATIN CAPITAL LETTER S LETTRE MAJUSCULE S
<T>	084	T	LATIN CAPITAL LETTER T LETTRE MAJUSCULE T
<U>	085	U	LATIN CAPITAL LETTER U LETTRE MAJUSCULE U
<V>	086	V	LATIN CAPITAL LETTER V LETTRE MAJUSCULE V
<W>	087	W	LATIN CAPITAL LETTER W LETTRE MAJUSCULE W
<X>	088	X	LATIN CAPITAL LETTER X LETTRE MAJUSCULE X
<Y>	089	Y	LATIN CAPITAL LETTER Y LETTRE MAJUSCULE Y
<Z>	090	Z	LATIN CAPITAL LETTER Z LETTRE MAJUSCULE Z
<left-bracket>	091	[	LEFT SQUARE BRACKET CROCHET GAUCHE
<backslash>	092	\	REVERSE SOLIDUS BARRE OBLIQUE INVERSÉE
<right-square-bracket>	093	]	RIGHT SQUARE BRACKET CROCHET DROIT
<circumflex>	094	^	CIRCUMFLEX ACCENT ACCENT CIRCONFLEXE
<underscore>	095	_	LOW LINE TRAIT BAS
<grave-accent>	096	`	GRAVE ACCENT ACCENT GRAVE
<a>	097	a	LATIN SMALL LETTER A LETTRE MINUSCULE a
<b>	098	b	LATIN SMALL LETTER B LETTRE MINUSCULE b
<c>	099	c	LATIN SMALL LETTER C LETTRE MINUSCULE c
<d>	100	d	LATIN SMALL LETTER D LETTRE MINUSCULE d
<e>	101	e	LATIN SMALL LETTER E LETTRE MINUSCULE e
<f>	102	f	LATIN SMALL LETTER F LETTRE MINUSCULE f
<g>	103	g	LATIN SMALL LETTER G LETTRE MINUSCULE g
<h>	104	h	LATIN SMALL LETTER H LETTRE MINUSCULE h
<i>	105	i	LATIN SMALL LETTER I LETTRE MINUSCULE i
<j>	106	j	LATIN SMALL LETTER J LETTRE MINUSCULE j
<k>	107	k	LATIN SMALL LETTER K LETTRE MINUSCULE k
<l>	108	l	LATIN SMALL LETTER L LETTRE MINUSCULE l
<m>	109	m	LATIN SMALL LETTER M LETTRE MINUSCULE m
<n>	110	n	LATIN SMALL LETTER N LETTRE MINUSCULE n
<o>	111	o	LATIN SMALL LETTER O LETTRE MINUSCULE o
<p>	112	p	LATIN SMALL LETTER P LETTRE MINUSCULE p
<q>	113	q	LATIN SMALL LETTER Q LETTRE MINUSCULE q

Table 10: Caractères ISO/IEC8859-1 (suite)

<r>	114	r	LATIN SMALL LETTER R LETTRE MINUSCULE r
<s>	115	s	LATIN SMALL LETTER S LETTRE MINUSCULE s
<t>	116	t	LATIN SMALL LETTER T LETTRE MINUSCULE t
<u>	117	u	LATIN SMALL LETTER U LETTRE MINUSCULE u
<v>	118	v	LATIN SMALL LETTER V LETTRE MINUSCULE v
<w>	119	w	LATIN SMALL LETTER W LETTRE MINUSCULE w
<x>	120	x	LATIN SMALL LETTER X LETTRE MINUSCULE x
<y>	121	y	LATIN SMALL LETTER Y LETTRE MINUSCULE y
<z>	122	z	LATIN SMALL LETTER Z LETTRE MINUSCULE z
<left-brace>	123	{	LEFT CURLY BRACKET ACCOLADE GAUCHE
<vertical-line>	124	|	VERTICAL LINE BARRE VERTICALE
<right-brace>	125	}	RIGHT CURLY BRACKET ACCOLADE DROITE
<tilde>	126	~	TILDE TILDE
<delete> or <DEL>	127		DELETE OBLITÉRATION
Caractères de contrôle C1			
Les codes 128–131 et 152–154 sont d'ISO/IEC10646, les autres d'ISO/IEC6429[30].			
<PAD>	128		PADDING CHARACTER CARACTÈRE DE GARNISSAGE
<HOP>	129		HIGH OCTET PRESET OCTET SUPÉRIEUR PRÉDÉFINI
<BHP>	130		BREAK PERMITTED HERE INTERRUPTION PERMISE ICI
<NBH>	131		NO BREAK HERE INTERRUPTION INTERDITE ICI
<IND>	132		INDEX INDEX
<NEL>	133		NEXT LINE LIGNE SUIVANTE
<SSA>	134		START OF SELECTED AREA DÉBUT DE ZONE SELECTIONNÉE
<ESA>	135		END OF SELECTED AREA FIN DE ZONE SELECTIONNÉE
<HTS>	136		HORIZONTAL TABULATION SET METTRE TABULATION HORIZONTALE
<HTJ>	137		CHARACTER TABULATION WITH JUSTIFICATION CARACTÈRE DE TABULATION AVEC JUSTIFICATION
<VTS>	138		VERTICAL TABULATION SET METTRE TABULATION VERTICALE
<PLD>	139		PARTIAL LINE DOWN DÉCALAGE VERS LE BAS D'UNE FRACTION DE LIGNE
<PLU>	140		PARTIAL LINE UP DÉCALAGE VERS LE HAUT D'UNE FRACTION DE LIGNE
<RI>	141		REVERSE INDEX INDEX INVERSÉ
<SS2>	142		SINGLE-SHIFT TWO

[30] Traduction française par les auteurs basée sur la description dans ISO/IEC6429

Table 10: Caractères ISO/IEC8859-1 (suite)

			DÉCALAGE SIMPLE DEUX (ISO 2022)
<SS3>	143		SINGLE-SHIFT THREE
			DÉCALAGE SIMPLE TROIS (ISO 2022)
<DCS>	144		DEVICE CONTROL STRING
			DÉBUT DE CHAÎNE DE CONTRÔLE DE PÉRIFÉRIQUE
<PU1>	145		PRIVATE USE ONE
			FONCTION PRIVÉE UN
<PU2>	146		PRIVATE USE TWO
			FONCTION PRIVÉE DEUX
<STS>	147		SET TRANSMIT STATE
			INITIALISER L'ÉTAT DE TRANSMISSION
<CCH>	148		CANCEL CHARACTER
			ANNULER CARACTÈRE
<MW>	149		MESSAGE WAITING
			MESSAGE EN ATTENTE
<SPA>	150		START OF GUARDED PROTECTED AREA
			DÉBUT DE ZONE PROTÉGÉE
<EPA>	151		END OF GUARDED PROTECTED AREA
			FIN DE ZONE PROTÉGÉE
<SOS>	152		START OF STRING
			DÉBUT DE CHAÎNE
<SGCI>	153		SINGLE GRAPHIC CHARACTER INTRODUCER
			DÉBUT DE CARACTÈRE GRAPHIQUE SIMPLE
<SCI>	154		SINGLE CHARACTER INTRODUCER
			DÉBUT DE CARACTÈRE SIMPLE
<CSI>	155		CONTROL SEQUENCE INTRODUCER
			DÉBUT DE SÉQUENCE DE CONTRÔLE
<ST>	156		STRING TERMINATOR
			FIN DE CHAÎNE
<OSC>	157		OPERATING SYSTEM COMMAND
			DÉBUT DE COMMANDE DE SYSTÈME D'EXPLOITATION
<PM>	158		PRIVACY MESSAGE
			DÉBUT DE MESSAGE CONFIDENTIEL
<APC>	159		APPLICATION PROGRAM CONTROL
			SÉQUENCE DE CONTRÔLE DE PROGRAMME D'APPLICATION
Nom des caractères basé sur ISO/IEC8859-1[31]			
Les lettres accentuées ont le nom du caractère de base suivi du type d'accent			
<no-break-space>	160		NO-BREAK SPACE
			ESPACE SANS COUPURE
<inverted-exclamation>	161	¡	INVERTED EXCLAMATION MARK
			POINT D'EXCLAMATION INVERSÉ
<cent-sign>	162	¢	CENT SIGN
			SYMBOLE CENTIME
<pound-sign>	163	£	POUND SIGN
			SYMBOLE LIVRE
<currency-sign>	164	¤	CURRENCY SIGN
			SYMBOLE MONITAIRE
<yen-sign>	165	¥	YEN SIGN
			SYMBOLE YEN
<broken-bar>	166	¦	BROKEN BAR
			BARRE VERTICALE INTERROMPUE
<paragraph-sign>	167	§	PARAGRAPH SIGN
			SIGNE PARAGRAPHE
<diaeresis>	168	¨	DIAERESIS
			TRÉMA
<copyright-sign>	169	©	COPYRIGHT SIGN
			SYMBOLE COPYRIGHT
<feminine-ordinal-a>	170	ª	FEMININE ORDINAL INDICATOR
			INDICATEUR ORDINAL FÉMININ

[31] Termes français de ISO/IEC8859-1(F) du 15 février 1987.

Table 10: Caractères ISO/IEC8859-1 (suite)

<left-angle-quotation>	171	«	LEFT-POINTING DOUBLE ANGLE QUOTATION MARK GUILLEMET ANGULAIRE GAUCHE
<not-sign>	172	¬	NOT SIGN SYMBOLE NÉGATION
<soft-hyphen>	173		SOFT HYPHEN TIRET TEMPORAIRE
<registered-mark>	174	®	REGISTERED SIGN SYMBOLE MARQUE DÉPOSÉE
<macron>	175	¯	MACRON MACRON
<degree-sign>	176	°	DEGREE SIGN ROND SUPÉRIEUR, SIGNE DEGRÉ
<plus-minus>	177	±	PLUS-MINUS SIGN SIGNE PLUS OU MOINS
<superscript-2>	178	²	SUPERSCRIPT TWO EXPOSANT DEUX
<superscript-3>	179	³	SUPERSCRIPT THREE EXPOSANT TROIS
<acute-accent>	180	´	ACUTE ACCENT ACCENT AIGU
<micro-sign>	181	µ	MICRO SIGN SYMBOLE MICRO
<pilcrow-sign>	182	¶	PILCROW SIGN SYMBOLE PARAGRAPHE
<middle-dot>	183	·	MIDDLE DOT POINT CENTRAL
<cedilla>	184	¸	CEDILLA CÉDILLE
<superscript-1>	185	¹	SUPERSCRIPT ONE EXPOSANT UN
<masculine-ordinal-o>	186	º	MASCULINE ORDINAL INDICATOR INDICATEUR ORDINAL MASCULIN
<right-angle-quotation>	187	»	RIGHT-POINTING DOUBLE ANGLE QUOTATION MARK GUILLEMET ANGULAIRE DROIT
<one-quarter>	188	¼	VULGAR FRACTION ONE QUARTER FRACTION ORDINAIRE UN QUART
<one-half>	189	½	VULGAR FRACTION ONE HALF FRACTION ORDINAIRE UN DEMI
<three-quarters>	190	¾	VULGAR FRACTION THREE QUARTERS FRACTION ORDINAIRE TROIS QUARTS
<inverted-question>	191	¿	INVERTED QUESTION MARK POINT D'INTERROGATION INVERSÉ
<A-grave>	192	À	LATIN CAPITAL LETTER A WITH GRAVE LETTRE MAJUSCULE A AVEC ACCENT GRAVE
<A-acute>	193	Á	LATIN CAPITAL LETTER A WITH ACUTE LETTRE MAJUSCULE A AVEC ACCENT AIGU
<A-circumflex>	194	Â	LATIN CAPITAL LETTER A WITH CIRCUMFLEX LETTRE MAJUSCULE A AVEC ACCENT CIRCONFLEXE
<A-tilde>	195	Ã	LATIN CAPITAL LETTER A WITH TILDE LETTRE MAJUSCULE A AVEC TILDE
<A-diaeresis>	196	Ä	LATIN CAPITAL LETTER A WITH DIAERESIS LETTRE MAJUSCULE A AVEC TRÉMA
<A-ring>	197	Å	LATIN CAPITAL LETTER A WITH RING ABOVE LETTRE MAJUSCULE A AVEC ACCENT ROND SUPÉRIEUR
<AE>	198	Æ	LATIN CAPITAL LIGATURE AE DIPHTONGUE MAJUSCULE AE
<C-cedilla>	199	Ç	LATIN CAPITAL LETTER C WITH CEDILLA LETTRE MAJUSCULE C AVEC CÉDILLE
<E-grave>	200	È	LATIN CAPITAL LETTER E WITH GRAVE LETTRE MAJUSCULE E AVEC ACCENT GRAVE
<E-acute>	201	É	LATIN CAPITAL LETTER E WITH ACUTE LETTRE MAJUSCULE E AVEC ACCENT AIGU

Table 10: Caractères ISO/IEC8859-1 (suite)

<E-circumflex>	202	Ê	LATIN CAPITAL LETTER E WITH CIRCUMFLEX LETTRE MAJUSCULE E AVEC ACCENT CIRCONFLEXE
<E-diaeresis>	203	Ë	LATIN CAPITAL LETTER E WITH DIAERESIS LETTRE MAJUSCULE E AVEC TRÉMA
<I-grave>	204	Ì	LATIN CAPITAL LETTER I WITH GRAVE LETTRE MAJUSCULE I AVEC ACCENT GRAVE
<I-acute>	205	Í	LATIN CAPITAL LETTER I WITH ACUTE LETTRE MAJUSCULE I AVEC ACCENT AIGU
<I-circumflex>	206	Î	LATIN CAPITAL LETTER I WITH CIRCUMFLEX LETTRE MAJUSCULE I AVEC ACCENT CIRCONFLEXE
<I-diaeresis>	207	Ï	LATIN CAPITAL LETTER I WITH DIAERESIS LETTRE MAJUSCULE I AVEC TRÉMA
<Eth>	208	Ð	LATIN CAPITAL LETTER ETH (ICELANDIC) LETTRE MAJUSCULE ISLANDAISE ETH
<N-tilde>	209	Ñ	LATIN CAPITAL LETTER N WITH TILDE LETTRE MAJUSCULE N AVEC TILDE
<O-grave>	210	Ò	LATIN CAPITAL LETTER O WITH GRAVE LETTRE MAJUSCULE O AVEC ACCENT GRAVE
<O-acute>	211	Ó	LATIN CAPITAL LETTER O WITH ACUTE LETTRE MAJUSCULE O AVEC ACCENT AIGU
<O-circumflex>	212	Ô	LATIN CAPITAL LETTER O WITH CIRCUMFLEX LETTRE MAJUSCULE O AVEC ACCENT CIRCONFLEXE
<O-tilde>	213	Õ	LATIN CAPITAL LETTER O WITH TILDE LETTRE MAJUSCULE O AVEC TILDE
<O-diaeresis>	214	Ö	LATIN CAPITAL LETTER O WITH DIAERESIS LETTRE MAJUSCULE O AVEC TRÉMA
<multiplication-sign>	215	×	MULTIPLICATION SIGN SIGNE MULTIPLICATION
<O-slash>	216	Ø	LATIN CAPITAL LETTER O WITH STROKE LETTRE MAJUSCULE O AVEC BARRE OBLIQUE
<U-grave>	217	Ù	LATIN CAPITAL LETTER U WITH GRAVE LETTRE MAJUSCULE U AVEC ACCENT GRAVE
<U-acute>	218	Ú	LATIN CAPITAL LETTER U WITH ACUTE LETTRE MAJUSCULE U AVEC ACCENT AIGU
<U-circumflex>	219	Û	LATIN CAPITAL LETTER U WITH CIRCUMFLEX LETTRE MAJUSCULE U AVEC ACCENT CIRCONFLEXE
<U-diaeresis>	220	Ü	LATIN CAPITAL LETTER U WITH DIAERESIS LETTRE MAJUSCULE U AVEC TRÉMA
<Y-acute>	221	Ý	LATIN CAPITAL LETTER Y WITH ACUTE LETTRE MAJUSCULE Y AVEC ACCENT AIGU
<Thorn>	222	Þ	LATIN CAPITAL LETTER THORN (ICELANDIC) THORN MAJUSCULE ISLANDAIS
<sharp-s>	223	ß	LATIN SMALL LETTER SHARP S (GERMAN) LETTRE MINUSCULE ALLEMANDE DOUBLE s
<a-grave>	224	à	LATIN SMALL LETTER A WITH GRAVE LETTRE MINUSCULE a AVEC ACCENT GRAVE
<a-acute>	225	á	LATIN SMALL LETTER A WITH ACUTE LETTRE MINUSCULE a AVEC ACCENT AIGU
<a-circumflex>	226	â	LATIN SMALL LETTER A WITH CIRCUMFLEX LETTRE MINUSCULE a AVEC ACCENT CIRCONFLEXE
<a-tilde>	227	ã	LATIN SMALL LETTER A WITH TILDE LETTRE MINUSCULE a AVEC TILDE
<a-diaeresis>	228	ä	LATIN SMALL LETTER A WITH DIAERESIS LETTRE MINUSCULE a AVEC TRÉMA
<a-ring>	229	å	LATIN SMALL LETTER A WITH RING ABOVE LETTRE MINUSCULE a AVEC ROND SUPÉRIEUR
<ae>	230	æ	LATIN SMALL LIGATURE AE DIPHTONGUE MINUSCULE ae
<c-cedilla>	231	ç	LATIN SMALL LETTER C WITH CEDILLA LETTRE MINUSCULE c AVEC CÉDILLE
<e-grave>	232	è	LATIN SMALL LETTER E WITH GRAVE LETTRE MINUSCULE e AVEC ACCENT GRAVE

Table 10: Caractères ISO/IEC8859-1 (suite)

<e-acute>	233	é	LATIN SMALL LETTER E WITH ACUTE LETTRE MINUSCULE e AVEC ACCENT AIGU
<e-circumflex>	234	ê	LATIN SMALL LETTER E WITH CIRCUMFLEX LETTRE MINUSCULE e AVEC ACCENT CIRCONFLEXE
<e-diaeresis>	235	ë	LATIN SMALL LETTER E WITH DIAERESIS LETTRE MINUSCULE e AVEC TRÉMA
<i-grave>	236	ì	LATIN SMALL LETTER I WITH GRAVE LETTRE MINUSCULE i AVEC ACCENT GRAVE
<i-acute>	237	í	LATIN SMALL LETTER I WITH ACUTE LETTRE MINUSCULE i AVEC ACCENT AIGU
<i-circumflex>	238	î	LATIN SMALL LETTER I WITH CIRCUMFLEX LETTRE MINUSCULE i AVEC ACCENT CIRCONFLEXE
<i-diaeresis>	239	ï	LATIN SMALL LETTER I WITH DIAERESIS LETTRE MINUSCULE i AVEC TRÉMA
<eth>	240	ð	LATIN SMALL LETTER ETH (ICELANDIC) LETTRE MINUSCULE ISLANDAISE ETH
<n-tilde>	241	ñ	LATIN SMALL LETTER N WITH TILDE LETTRE MINUSCULE n AVEC TILDE
<o-grave>	242	ò	LATIN SMALL LETTER O WITH GRAVE LETTRE MINUSCULE o AVEC ACCENT GRAVE
<o-acute>	243	ó	LATIN SMALL LETTER O WITH ACUTE LETTRE MINUSCULE o AVEC ACCENT AIGU
<o-circumflex>	244	ô	LATIN SMALL LETTER O WITH CIRCUMFLEX LETTRE MINUSCULE o AVEC ACCENT CIRCONFLEXE
<o-tilde>	245	õ	LATIN SMALL LETTER O WITH TILDE LETTRE MINUSCULE o AVEC TILDE
<o-diaeresis>	246	ö	LATIN SMALL LETTER O WITH DIAERESIS LETTRE MINUSCULE o AVEC TRÉMA
<division-sign>	247	÷	DIVISION SIGN SIGNE DIVISION
<o-slash>	248	ø	LATIN SMALL LETTER O WITH STROKE LETTRE MINUSCULE o AVEC BARRE OBLIQUE
<u-grave>	249	ù	LATIN SMALL LETTER U WITH GRAVE LETTRE MINUSCULE u AVEC ACCENT GRAVE
<u-acute>	250	ú	LATIN SMALL LETTER U WITH ACUTE LETTRE MINUSCULE u AVEC ACCENT AIGU
<u-circumflex>	251	û	LATIN SMALL LETTER U WITH CIRCUMFLEX LETTRE MINUSCULE u AVEC ACCENT CIRCONFLEXE
<u-diaeresis>	252	ü	LATIN SMALL LETTER U WITH DIAERESIS LETTRE MINUSCULE u AVEC TRÉMA
<y-acute>	253	ý	LATIN SMALL LETTER Y WITH ACUTE LETTRE MINUSCULE y AVEC ACCENT AIGU
<thorn>	254	þ	LATIN SMALL LETTER THORN (ICELANDIC) THORN MINUSCULE ISLANDAIS
<y-diaresis>	255	ÿ	LATIN SMALL LETTER Y WITH DIAERESIS LETTRE MINUSCULE y AVEC TRÉMA

B. La norme ISO-639 et son extension CD 639-2

La norme ISO-639 [31] propose des codes à deux lettres pour représenter les langues les plus importantes. Il correspond aux premières et quatrièmes colonnes de la table 11. Récemment une extension à trois lettres du code a été proposée [32]. Il apparaît dans les colonnes deux et cinq de la même table, où les troisièmes et sixièmes colonnes donnent le nom normatif français des langues en question.

Table 11: Les codes pour les langues ISO-639/ISO-CD 639-2

aa	aar	afar	ab	abk	abkhaze
	ace	aceh		ach	acoli
	ada	adangme		afa	afro-asiatiques, autres langues
	afh	afrihili	af	afr	afrikaans
	aka	akan		akk	akkadian
sq	alb/sqi	albanais		ale	aléoute
	alg	algonquines, langues	am	amh	amharique
	ang	anglo-saxon (ca 450-1100)		apa	apache
ar	ara	arabe		arc	araméen
	arm/hye	arménien		arn	araucan
	arp	arapaho		art	artificielles, autres langues
	arw	arawak	as	asm	assamais
	ath	athapascanes, langues		ava	avar
	ave	aveste		awa	awadhi
ay	aym	aymara	az	aze	azéri
	bad	banda		bai	bamiléké, langues
ba	bak	bachkir		bal	baloutchi
	bam	bambara		ban	balinais
	baq/eus	basque		bas	basa
	bat	baltiques, autres langues		bej	bedja
be	bel	biélorusse		bem	bemba
bn	ben	bengali		ber	berbères, langues
	bho	bhojpuri	bh	bih	bihari
	bik	bikol		bin	bini
bi	bis	bichlamar		bla	blackfoot
bo	bod/tib	tibétain		bra	braj
br	bre	breton		bua	bouriate
	bug	bugi	bg	bul	bulgare
my	bur/mya	birman		cad	caddo
	cai	indiennes d'Amérique centrale, autres langues		car	caribe
ca	cat	catalan		cau	caucasiennes, autres langues
	ceb	cebuano		cel	celtiques, autres langues
	ces/cze	tchèque		cha	chamorro
	chb	chibcha		che	tchetchène
	chg	djaghataï	zh	chi/zho	chinois
	chm	mari		chn	chinook, jargon
	cho	choctaw		chr	cherokee
	chu	slavon		chv	tchouvache
	chy	cheyenne		cop	copte
	cor	cornique	co	cos	corse
	cpe	créoles et pidgins anglais, autres		cpf	créoles et pidgins français, autres

Table 11: Les codes ISO 639/CD 639-2 (*suite*)

	cpp	créoles et pidgins portugais, autres		cre	cree
	crp	créoles et pidgins divers		cus	couchitiques, autres langues
cy	cym/wel	gallois	cs	cze/ces	tchèque
	dak	dakota	da	dan	danois
	del	delaware	de	deu/ger	allemand
	din	dinka		div	maldivien
	doi	dogri		dra	dravidiennes, autres langues
	dua	douala		dum	néerlandais moyen (ca 1050-1350)
nl	dut/nld	néerlandais		dyu	dioula
dz	dzo	dzongkha		efi	efik
	egy	égyptien		eka	ekajuk
el	ell/gre	grec moderne (1453–)		elx	élamite
en	eng	anglais	eo	enm	anglais moyen (1100–1500)
	epo	espéranto		esk	inuit
es	esl/spa	espagnol	et	est	estonien
eu	eus/baq	basque		ewe	éwé
	ewo	ewondo		fan	fang
fo	fao	féroïen	fa	fas/per	persan
	fat	fanti	fj	fij	fidjien
fi	fin	finois		fiu	finno-ougriennes, autres langues
	fon	fon	fr	fra/fre	français
fr	fre/fra	français		frm	français moyen (ca 1400–1600)
	fro	français vieux (842–ca 1400)	fy	fry	frison
	ful	peul		gaa	ga
	gae/gdh	gaélique d'Écosse	ga	gai/iri	irlandais
	gay	gayo	gd	gdh/gae	gaélique d'Écosse
	gem	germaniques, autres langues	ka	geo/kat	géorgien
	ger/deu	allemand		gez	guèze
	gil	kiribati	gl	glg	galicien
	gmh	allemand, moyen haut (ca 1050–1500)		goh	allemand, vieux haut (ca 750–1050)
	gon	gond		got	gothique
	grb	grebo		grc	grec ancien (jusqu'à 1453)
	gre/ell	grec moderne (1453–)	gn	grn	guarani
gu	guj	goudjarati		hai	haida
ha	hau	haoussa		haw	hawaïen
iw	heb	hébreu		her	herero
	hil	hiligaynon		him	himachali
hi	hin	hindi		hmo	hiri motu
hu	hun	hongrois		hup	hupa
hy	hye/arm	arménien		iba	iban
	ibo	igbo		ice/isl	islandais

Table 11: Les codes ISO 639/CD 639-2 (*suite*)

	ijo	ijo
ie	ile	interlingue
ia	ina	interlingua (International Auxilary Language Association)
in	ind	indonésien
ik	ipk	inupiak
	iri/gai	irlandais
is	isl/ice	islandais
jw	jav/jaw	javanais
ja	jpn	japanais
	jrb	judéo-arabe
	kab	kabyle
kl	kal	groenlandais
kn	kan	kannada
ks	kas	kashmiri
	kau	kanouri
kk	kaz	kazakh
	khi	khoisan, autres langues
	kho	khotanais
rw	kin	rwanda
	kok	konkani
ko	kor	coréen
	kro	kru
	kua	kuanyama
ku	kur	kurde
	kut	kutenai
	lah	lahnda
lo	lao	lao
lv	lav	letton
ln	lin	lingala
	lol	mongo
	ltz	luxembourgeois
	lug	ganda
	lun	lunda
mk	mac/mke	macédonien
	mag	magahi
	mai	maithili
ml	mal	malayalam
	mao/mri	maori
mr	mar	marathe
	max	manx
	men	mendé
	mic	micmac
	iku	inuktitut
	ilo	ilocano
	inc	indo-aryennes, autres langues
	ine	indo-européennes, autres langues
	ira	iraniennes, autres langues
	iro	iroquoises, langues
it	ita	italien
jw	jaw/jav	javanais
	jpr	judéo-persan
	kaa	karakalpak
	kac	kachin
	kam	kamba
	kar	karen
	kat/geo	géorgien
	kaw	kawi
	kha	khasi
km	khm	cambodgien
	kik	kikuyu
ky	kir	kirghiz
	kon	kongo
	kpe	kpellé
	kru	kurukh
	kum	koumyk
	kus	kusaie
	lad	judéo-espagnol
	lam	lamba
la	lat	latin
	lez	lezghien
lt	lit	lituanien
	loz	lozi
	lub	luba
	lui	luiseno
	luo	luo (Kenya and Tanzanie)
	mad	madourais
	mah	marshall
	mak	makassar
	man	mandingue
	map	malayo-polynésiennes, autres langues
	mas	masai
ms	may/msa	malais
	mga	irlandais moyen (900–1200)
	min	minangkabau

Table 11: Les codes ISO 639/CD 639-2 (*suite*)

	mis	diverses, langues
	mkh	môn-khmer, autres langues
mt	mlt	maltais
	mno	manobo
mo	mol	moldave
	mos	mossi
	msa/may	malais
	mun	mounda, langues
	mwr	marwari
	myn	maya, langues
	nai	indiennes d'Amérique du Nord, autres langues
	nav	navaho
	nde	ndébéle du Nord
	ndo	ndonga
	new	newari
	niu	niué
	non	norrois, vieux
	nso	sotho du Nord
	nya	nyanja
	nyn	nyankolé
	nzi	nzema
	oji	ojibwa
om	orm	galla
	oss	ossète
	oto	otomangue, langues
	pag	pangasinan
	pam	pampangan
	pap	papiamento
	peo	perse, vieux (ca 600-400 av. J.-C.)
	phn	phénicien
pl	pol	polonais
pt	por	portugais
	pro	provençal (jusqu'à 1500)
qu	que	quechua
	rar	rarotonga
rm	roh	rhéto-roman
ro	ron/rum	roumain
rn	run	rundi
	sad	sandawe
	sah	iakoute
	mke/mac	macédonien
mg	mlg	malgache
	mni	manipuri
	moh	mohawk
mn	mon	mongole
mi	mri/mao	maori
	mul	multilingue
	mus	muskogee
my	mya/bur	birman
	nah	nahautl
na	nau	nauru
	nbl	ndébéle du Sud
nl	nld/dut	néerlandais
ne	nep	népalais
	nic	nigéro-congolaises, autres langues
	nno	norvégien (nynorsk)
no	nor	norvégien
	nub	nubiennes, langues
	nym	nyamwezi
	nyo	nyoro
oc	oci	occitan (après 1500)
or	ori	oriya
	osa	osage
	ota	turc ottoman (1500–1928)
	paa	papoues-australiennes, autres langues
	pal	pahlavi
pa	pan	Pendjabi
	pau	palau
	per/fas	persan
	pli	pali
	pon	ponape
	pra	prâkit
ps	pus	pachto
	raj	rajasthani
	roa	romanes, autres langues
	rom	tsigane
ro	rum/ron	roumain
ru	rus	russe
sg	sag	sango
	sai	indiennes d'Amérique du Sud, autres langues

Table 11: Les codes ISO 639/CD 639-2 (*suite*)

	sal	salish, langues		sam	samaritain
sa	san	sanskrit		sco	écossais
sh	scr	serbo-croate		sel	selkoup
	sem	sémitiques, autres langues		sga	irlandais ancien (jusqu'à 900)
	shn	chan		sid	sidamo
si	sin	cinghalais		sio	sioux, langues
	sit	sino-tibétaines, autres langues		sla	slaves, autres langues
sk	slk/slo	slovaque	sk	slo/slk	slovaque
sl	slv	slovène		smi	lapon
sm	smo	samoan	sn	sna	shona
sd	snd	sindhi		sog	sogdien
so	som	somali		son	songhai
	sot	sotho du Sud		spa/esl	espagnol
sq	sqi/alb	albanais		sra	sarde
	srr	sérère		ssa	nilo-sahariennes, autres langues
	ssw	swazi		suk	sukuma
su	sun	soundanais		sus	sousou
	sux	sumérien	sv	sve/swe	suédois
sw	swa	souahili	sv	swe/sve	suédois
	syr	syriaque		tah	tahitien
ta	tam	tamoul	tt	tat	tatare
te	tel	telougou		tem	temne
	ter	tereno	tg	tgk	tadjik
tl	tgl	tagalog	th	tha	thaï
	tib/bod	tibétain		tig	tigré
ti	tir	tigrigna		tiv	tiv
	tli	tlingit		tmh	tamacheq
to	tog	tonga (Nyasa)	to	ton	tonga (îles Tonga)
	tru	truk		tsi	tsimshian
tn	tsn	tswana	ts	tso	tsonga
tk	tuk	turkmène		tum	tumbuka
tr	tur	turc		tut	altaïques, autres langues
tw	twi	twi		tyv	touva
	uga	ougaritique		uig	ouïgour
uk	ukr	ukrainien		umb	umbundu
	und	indéterminé	ur	urd	ourdou
uz	uzb	ouzbek		vai	vaï
	ven	venda	vi	vie	vietnamien
vo	vol	volapük		vot	vote
	wak	wakashennes, langues		wal	walamo
	war	waray		was	washo
	wel/cym	gallois		wen	wenda
wo	wol	oualof	xh	xho	xhosa

Table 11: Les codes ISO 639/CD 639-2 (*suite*)

	yao	yao		yap	yapois
ji	yid	yiddish	yo	yor	yorouba
	zap	zapotèque		zen	zenaga
	zha	zhuang	zh	zho/chi	chinois
zu	zul	zoulou		zun	zuni

C. L'extension `inputenc`

L'extension `inputenc` de LaTeX2ε permet à l'utilisateur de spécifier le codage de saisie (par exemple ASCII, ISOLATIN-1 ou Macintosh). On dira

```
\usepackage[codage]{inputenc}
```

À l'intérieur d'un document le codage peut être spécifié avec la commande :

```
\inputencoding{codage}
```

Par défaut la distribution LaTeX2ε vient avec les codages :

ascii codage ASCII (codes 0 à 127) ;

latin1 codage ISOLATIN-1

En plus il existe des fichiers provenant du monde PC (par exemple pour les compatibles PC commercialisés en France où l'on utilise le "pagecode 850" il y a un fichier `pc850`) et du monde Macintosh (avec un fichier qui s'appelle `stdmac`).

À chaque codage correspond un fichier de type `.def`, par exemple `latin1.def`. Dans un tel fichier on associe à chaque caractère à l'entrée une commande LaTeX en utilisant la syntaxe :

```
\DeclareInputText{car}{texte}
\DeclareInputMath{car}{math}
```

Ces commandes spécifient que le caractère « *car* » correspond à la commande LaTeX « *texte* » et « *math* » en mode texte et mathématique, respectivement. Par exemple dans le fichier `latin1.def` on trouvera les définitions suivantes :

```
\DeclareInputText{"D6}{\AE}   % Æ en mode texte
\DeclareInputMath{"B5}{\mu}   % μ en mode mathématique
```

Le fichier de type `.def` peut également contenir des déclarations utilisant les commandes `\providecommand` et `\ProvideTextCommandDefault`, par exemple,

```
\ProvideTextCommandDefault{\textonequarter}{$\frac14$}
\DeclareInputText{"BC}{\textonequarter}
```

Une autre extension pourra donc changer le comportement du caractère `"BC` en redéfinissant la commande `\textonequarter`.

Restrictions

— La version actuelle de l'extension `inputenc.sty` n'autorise pas qu'un caractère *car* soit associé en même temps à une déclaration texte et mathématique.

— Le fichier avec les définitions de type `.def` ne doit pas contenir des déclarations qui dépendent du codage de sortie (et des polices).

— Un fichier `.def` ne définit que les codes supérieur à 127 (c. à d. le fichier `ascii.def` est vide).

— Le fichier `.def` ne doit pas contenir des définitions de polices ou un nombre trop important de nouvelles commandes.

— De la même façon on doit faire attention de ne pas (re)définir des noms (commandes) qui existent déjà dans d'autres extensions.

— En « inventant » des noms pour les nouveaux glyphes le développeur aura avantage à étudier le fichier `inputenc.dtx` qui contient quelques directives.