Avant-propos

Omniprésence d'Unicode

Depuis une bonne dizaine d'années, Unicode a discrètement métamorphosé le domaine des logiciels et des protocoles de communication. Là, où il y a dix ans à peine, existait une pléthore de codages de caractères différents, souvent incompatibles entre eux, Unicode a apporté simplicité et unicité de codage. Tous les grands systèmes d'exploitation utilisent désormais Unicode, les traitements de texte comme Open Office et Office de Microsoft en font de même. Que vous consultiez Wikipedia, lisiez votre courriel sur Google, recherchiez l'internet avec Yahoo!, vous utilisez Unicode.

Avant l'avènement d'Unicode, les programmes étaient souvent conçus pour ne prendre en charge qu'un jeu de caractères précis utilisé dans un marché particulier. Viser le marché international supposait le développement d'un grand nombre de versions parallèles. Concevoir, tester et entretenir en parallèle ces versions étaient un véritable cauchemar : il fallait souvent corriger le même problème dans chacune de ces versions.

Unicode permet désormais de ne plus développer qu'une seule version de son produit qui permet de prendre en charge plusieurs langues et d'assurer l'échange d'information sur une échelle planétaire.

À qui s'adresse ce livre ?

Les lecteurs de ce livre devraient avoir une bonne connaissance du fonctionnement des ordinateurs. Toutefois, il n'est pas nécessaire qu'ils soient programmeurs bien que de nombreux programmeurs puissent tirer parti de ce livre dans la conception de leurs logiciels.

Ce livre ne se veut pas une traduction du standard Unicode — disponible gratuitement en ligne autant en anglais que, pour une grande partie, en français —

mais plutôt une introduction à Unicode pour certains aspects et un complément au standard Unicode pour d'autres.

Introduction plus concise, parfois, nous l'espérons, plus facile d'abord que le texte du standard Unicode, elle se destine d'abord au public francophone et se concentre sur les grandes écritures qui sauront l'intéresser : l'écriture latine, la grecque, l'arabe, les symboles mathématiques, physiques et musicaux sans toutefois négliger d'introduire les idéogrammes chinois.

Complément aussi, car, si le standard Unicode est principalement destiné aux concepteurs qui voudraient écrire des programmes qui mettent en œuvre des aspects du standard Unicode, par exemple comment afficher correctement l'arabe ou le birman dans un traitement de texte ou un système d'exploitation, le standard Unicode passe sous silence de nombreux aspects pratiques cruciaux pour ceux qui ne veulent qu'utiliser ces logiciels Unicode, comme les navigateurs web ou les bibliothèques Java, et non les concevoir.

Ce livre s'adresse à plusieurs publics parmi lesquels :

- Des informaticiens de métier, des analystes, des gestionnaires en informatique désireux de comprendre et de travailler avec Unicode. Qu'il s'agisse de conversion de données patrimoniales, de création de logiciels internationalisés ou de sites web multilingues ou encore tout simplement de travailler dans un environnement qui utilise Unicode, ce qui est de plus en plus le cas avec des technologies modernes comme XML, C# ou Java.
- Des personnes qui travaillent sur des textes multilingues ou des textes spécialisés : base de données multilingue, textes contenant des symboles mathématiques ou d'autres signes spéciaux, (même les textes juridiques contiennent beaucoup de caractères Unicode). Parmi ces utilisateurs typiques, on retrouvera des typographes, des bibliothécaires et des universitaires dans les facultés de lettres qui travaillent sur des textes multilingues.
- Des enseignants en informatique qui désirent mieux comprendre les différents aspects liés à Unicode et à l'internationalisation et pensent introduire leurs étudiants à ces sujets. Il n'existe à peu près aucun bon livre sur les jeux de caractères et l'internationalisation en français, une des conséquences de ce manque est l'absence de ces sujets dans les programmes universitaires alors que ces techniques sont pourtant de plus en plus demandées. Une bonne compréhension des bases d'Unicode et de l'internationalisation devrait faire partie du bagage de tout bon informaticien.

Structure du livre

Le livre est divisé en quatre parties : Introduction, L'essentiel d'Unicode, Caractères remarquables et Applications et techniques liées à Unicode.

Introduction

Le but de cette partie divisée en deux chapitres est de présenter brièvement les concepts de base, d'introduire la terminologie et de décrire les raisons qui ont poussé à la création et au succès d'Unicode : la pléthore des jeux de caractères.

Le premier chapitre de cette partie constitue également une prise en main d'Unicode : montrer comment l'utilisation peut être facile, comment afficher des caractères Unicode et les saisir. Cette partie est cruciale pour prendre la mesure d'un sujet tel qu'Unicode et l'internationalisation des documents et des logiciels.

Le deuxième chapitre présente les principaux jeux de caractères qui ont précédé Unicode et qui pour certains sont toujours utilisés. Au-delà de l'intérêt historique, ce chapitre permet d'expliquer de nombreux problèmes de conversion toujours très actuels, il peut également servir de référence lors de la conversion de données entre Unicode et, par exemple, les jeux de caractères codés de MS Windows.

L'essentiel d'Unicode

Constituée de deux autres chapitres, la deuxième partie présente l'essentiel d'Unicode. Ce qu'il faut que tout utilisateur averti d'Unicode comprenne d'Unicode. Nous avons voulu résumer et rendre plus accessible le standard Unicode qui peut parfois paraître abstrus et complexe.

Le chapitre 3 présente la structure d'Unicode : comment s'organisent les près de 100 000 caractères qu'Unicode définit désormais. On y explique ensuite les différences entre les différents types de caractères : les caractères de base, les diacritiques. Ce chapitre se penche ensuite sur les principes de conception d'Unicode : quels caractères ont été retenus, lesquels furent rejetés et pourquoi ?

Le chapitre 4 introduit ce qu'on nomme le modèle de caractères d'Unicode et explique la différence entre les différents formats codés d'Unicode : UTF-8, UTF-16 et UTF-32. Il présente ensuite les principales propriétés attachées aux caractères qui facilitent leur traitement ainsi que les formes normalisées d'Unicode qui permettent de comparer des chaînes de caractères Unicode qui pourraient être identiques bien que codées de manières différentes pour des raisons de compatibilité avec les codages patrimoniaux qu'Unicode a dû incorporer. Le chapitre 4 décrit également brièvement un algorithme important d'Unicode : celui qui permet de trier correctement les chaînes de caractères. Enfin, on présente comment un processus doit traiter les caractères Unicode pour qu'on puisse dire qu'il se conforme à Unicode et comment lire les tableaux de caractères Unicode.

Les caractères remarquables

La troisième partie regroupe quatre chapitres qui traitent des principaux types de caractères qui pourraient intéresser les lecteurs de cet ouvrage. Le premier de ces chapitres, le chapitre 5, se penche sur les lettres latines, l'alphabet phonétique et les diacritiques qui intéresseront non seulement les francophones, mais également de nombreux auteurs qui doivent transcrire des langues « exotiques » à l'aide de

l'alphabet latin étendu. Suivent ensuite deux sections — le choix est un peu arbitraire étant donné la taille de cet ouvrage — qui décrivent, l'une l'écriture grecque qui non seulement fait partie de notre culture (encore enseignée comme cours à option dans nos écoles) mais présente des défis intéressants en matière de casse et l'autre approfondit le sujet complexe des diacritiques.

Le chapitre 6 décrit les différents signes de ponctuation utilisés en français et la majorité des langues qui utilisent l'écriture latine. Certains de ces caractères de ponctuation sont également universels : ceux qui régissent la coupure de ligne.

Le chapitre 7 introduit les principaux symboles utilisés dans les notations mathématiques, physiques et la musique.

Quant au chapitre 8, il décrit les principales techniques d'Unicode : les caractères de commande hérités du Latin-1, ainsi que des caractères originaux d'Unicode comme l'Indicateur d'ordre des octets (IOO), le gluon de mots et même de ce qui peut apparaître comme une contradiction les « non-caractères ». Le chapitre 8 se termine par une description des caractères déconseillés et désuets et par les zones qu'Unicode a réservées à un usage privé.

Applications et techniques liées à Unicode

C'est la partie la plus importante du livre : elle comprend cinq chapitres qui traitent de différentes normes et techniques connexes à Unicode.

Le chapitre 9 définit les différentes normes qui permettent d'indiquer la langue (ISO 639), le pays (ISO 3166) et l'écriture (ISO 15924) d'un texte. Ainsi que les standards qui rassemblent et unifient ces normes : le RFC 4646 et le BCP 47.

Le chapitre 10 décrit comment préciser la langue et le codage dans les documents échangés sur Internet. Il aborde également les techniques du côté serveur liées à la négociation automatique de langue, c'est-à-dire comment un serveur Internet peut fournir la page dans la langue préférée du client quand plusieurs versions linguistiques de cette page existent. Enfin, le chapitre se clôt sur une discussion de l'internationalisation des adresses Internet. En d'autres mots, comment utiliser les caractères Unicode dans les noms de domaine tels que <http://écolelibre.com> ou <http://الجـزيرة.net>.

Quant au chapitre 11, il se concentre, à la différence du chapitre précédent qui s'intéressait aux protocoles Internet et aux serveurs, sur les techniques d'internationalisation des pages Internet et des documents XML : comment en préciser la langue, la directionalité quand elle est en arabe ou en hébreu, comment y référer aux caractères Unicode, quand il est préférable d'utiliser du balisage plutôt que des caractères Unicode, comment maîtriser l'affichage d'un texte en arabe ou en hébreu, et enfin comment écrire un formulaire HTML « universel » qui permet la saisie de tous les caractères Unicode.

L'internationalisation des logiciels est abordée au chapitre 12 : il s'agit ici de présenter les principales techniques et outils qui permettent de concevoir des logiciels qui pourront prendre en charge de nombreuses langues et être prêts pour le marché

mondial. L'internationalisation n'implique nullement la traduction des programmes, mais plutôt le développement de ces programmes selon une architecture logicielle qui permettra de traiter de nombreuses langues (trier des noms étrangers, les saisir, les comparer) et, le cas échéant, de traduire facilement l'interface utilisateur de ces programmes.

Le dernier chapitre, le chapitre 13, se penche sur la manière dont Unicode et les polices interagissent. Comment passe-t-on d'une suite de caractères Unicode au résultat affiché ? Comment maîtriser cet affichage (on parle aussi du rendu), plus particulièrement quand il s'agit d'afficher des écritures « exotiques » ou d'utiliser des fonctionnalités plus poussées comme le crénage ou la formation de ligature ?

Comment lire ce livre ?

Bien qu'on puisse lire le livre du premier chapitre au dernier et ainsi jouir d'un panorama complet de ce qu'est Unicode et ses techniques connexes, on peut imaginer différents itinéraires de lecture.

Les chapitres essentiels pour tous les lecteurs sont les chapitres 1 *Concept de base et terminologie*, 3 *Structure d'Unicode* et 4 *Modèle de codage, propriétés et caractères de tri*. Les autres chapitres sont relativement indépendants au prix, parfois, de certaines répétitions pédagogiques et de renvois vers d'autres chapitres où le lecteur trouvera un complément d'information.

Chaque chapitre est suivi d'un bref résumé de plusieurs paragraphes qui permet au lecteur pressé de ne pas lire au complet un chapitre qui lui paraîtrait accessoire ou ennuyeux tout en apprenant l'essentiel. Par ailleurs des petits encarts soulignent les points importants ou les pièges à éviter au sein des chapitres.

Le lecteur pressé peut considérer les chapitres sur les caractères remarquables comme des chapitres de référence auxquels il reviendra à loisir quand il se trouvera confronté aux caractères décrits dans ces chapitres. Nous pensons cependant que le chapitre 8, *Caractères techniques spéciaux*, est fortement recommandé à tous les lecteurs.

Les concepteurs et administrateurs de site web trouveront la lecture des chapitres 9, 10 et 11 instructive. Les programmeurs chargés de l'internationalisation de logiciels devraient lire les chapitres 9 et 12. Ceux qui désirent créer une police pour leur écriture préférée ou veulent comprendre comment fonctionne une police OpenType profiteront de la lecture du chapitre 13.

Conventions adoptées dans ce livre

- **Texte à chasse fixe et à barre latérale** — Le texte à chasse fixe et à barre gauche latérale représente du code dans l'acception la plus large de ce terme. Ainsi, l'extrait suivant représente-t-il du code XML :

```
<?xml version="1.0" encoding="UTF-8"?>
<?eclipse version="3.0"?>
<plugin
```

- **Texte italique à chasse fixe** — Dans le corps du texte explicatif, un passage `italique à chasse fixe` représente une constante qui doit être saisie telle quelle.
- **U+*nnnn*** — Dans le texte courant, U+*n* dénote un caractère où *n* est une suite de 4 à 6 chiffres hexadécimaux. U+0001, U+0012, U+0123, U+1234, U+12345 et U+1AF456 sont des numéros de caractères bien formés. Le préfixe U+ peut être omis dans les tableaux pour des raisons de concision. Un intervalle de numéro de caractères se représente ainsi : U+*xxxx*–U+*yyyy* ou *xxxx*..*yyyy*. L'intervalle U+0900–U+097F comprend 128 valeurs Unicode.
- **0*xyy* et 0*xwwww*** — Les nombres 0x*yy* et 0x*wwww* représentent à l'aide d'une notation commode — sans chiffres en indice — les chiffres hexadécimaux yy_{16} et $wwww_{16}$. Les lettres en italiques *yy* et *wwww* représentent des chiffres hexadécimaux, « 0x » est une constante qui doit s'écrire tel quel. Exemple : 0x2F ou 0x2f (la casse importe peu) représente le nombre $2F_{16}$, c'est-à-dire 47 en notation décimale.
- « **x** » — On entoure de guillemets un caractère qui pourrait être confondu avec la ponctuation ou un mot de la phrase qui le contient : « - », « , », « a » et « . ».
- **Nom en petites capitales** — S'écrivent en petites capitales les noms de caractère officiels ISO/CEI 10646. Le nom officiel du « & » est PERLUÈTE, son numéro de caractère est U+0026. Le numéro de caractère U+0178 représente la LETTRE MAJUSCULE LATINE Y TRÉMA Ÿ.

Suppléments en ligne

On retrouvera sur le site <http://hapax.qc.ca> de nombreux suppléments à cet ouvrage :

- La traduction en français de grandes parties du standard Unicode.
- Les tableaux de tous les caractères Unicode 5.0 en français (on les retrouve aussi sur le site <http://www.unicode.org/fr/charts>).
- Un logiciel qui permet de visualiser les caractères Unicode, de consulter leurs propriétés, de les rechercher par leur nom ou leur type et de les saisir : <http://hapax.qc.ca/BabelMap_fr.html>.
- Des exemples de programmes qui mettent en œuvre certaines techniques abordées dans ce livre.
- Un accès vers des normes ISO gratuites : l'ISO/CEI 10646 qui est le pendant ISO du codage Unicode et l'ISO/CEI 14651 pour le tri multilingue.
- Des convertisseurs de caractères, des polices OpenType, des articles originaux comme un entretien avec le directeur technique du consortium Unicode qui répond à plusieurs critiques souvent formulées au sujet d'Unicode, etc.

Remerciements

J'ai une dette toute particulière envers Aurélien Géron, auteur de plusieurs livres et cofondateur de Wifirst, qui a bien voulu relire le manuscrit final avec une rare minutie, il a suggéré de nombreuses améliorations et relevé de nombreuses coquilles et imprécisions.

Mark Davis, président et cofondateur d'Unicode, a bien voulu relire le manuscrit de cet ouvrage, prodiguer ses conseils sur plusieurs aspects importants et écrire la préface. Qu'il en soit remercié.

Un grand coup de chapeau à Jacques André, ancien directeur de recherche à l'INRIA qui a relu, à la lumière de sa grande connaissance en matières typographiques, de nombreux chapitres et plus particulièrement ceux qui traitent des caractères remarquables et des polices.

Vifs remerciements à François Yergeau, unicodien de longue date et coauteur de quelques standards W3C et IETF, qui a connu ce livre dès sa première mouture, a accompagné sa conception de conseils fréquents et qui a corrigé avec minutie le texte des chapitres 10 et 11.

Je dois également des remerciements à Stéphane Bortzmeyer, ingénieur auprès de l'AFNIC, pour ses corrections avisées et ses conseils relatifs aux chapitres 9, 10 et 11.

J'ai pu bénéficier de l'expertise d'Alain Labonté, expert et délégué du Canada/Québec auprès de l'ISO, auteurs de plusieurs normes ISO, pour tout ce qui a trait au tri multilingue ; il a également relu plusieurs chapitres. Je lui en suis fort reconnaissant.

Je tiens à remercier l'équipe des éditions Dunod, Jean-Luc Blanc et Carole Trochu, pour leur gentillesse, leurs conseils et leurs encouragements répétés.

Un grand merci à Youssef Aït Ouguengay, de l'IRCAM à Rabat, pour sa relecture des exemples arabes de ce livre.

De même, je tiens à exprimer ici ma gratitude envers Scott Horne, traducteur polyglotte et informaticien, qui a révisé les exemples chinois et japonais et ses suggestions en la matière.

Enfin, il me faut affectueusement remercier Tina, pour sa patience pendant l'écriture de ce livre. J'espère enfin pouvoir passer davantage de temps, ce livre terminé, avec elle et nos enfants (Hugues, Thierry, Astrid et Arnaud).

Erreurs, suggestions, commentaires ?

L'auteur remercie à l'avance tous ceux qui prendront le temps de lui signaler toute erreur qui se serait glissée dans le présent ouvrage, ainsi que toute suggestion qui permettrait d'améliorer ce livre.

Adresser les commentaires, les suggestions et les corrections à l'adresse suivante : <patrick@hapax.qc.ca>.