

---

# Unicode : tentations et limites

## L'avis d'un typographe

**Olivier Randier**

*orandier@fr.inter.net*

---

*RÉSUMÉ. L'examen du standard Unicode du point de vue du typographe, compositeur et dessinateur de caractères, révèle, à travers lacunes et erreurs de divers types, certaines tentations mercantiles et ethnocentriques inquiétantes. L'intégration de cet élément essentiel d'un futur système de composition typographique universel nécessite aujourd'hui de délimiter plus clairement attributions et limites du codage et de réfléchir à ses interconnexions avec les autres éléments (moteur de rendu typographique, fontes de caractères, interface de saisie...).*

*ABSTRACT. Examining Unicode standard as a typographer, typesetter and type designer, reveals, through various types of gaps and mistakes, some worrying mercantilist and ethnocentrist temptations. Integrating this essential item to a future universal typesetting system requires now to delimitate more accurately duties and limits of the encoding and to think about its interconnections with other items (typographic engine, fonts, data acquisition interface...).*

*MOTS-CLÉS : Unicode, typographie, composition multilingue, composition scientifique, ethnocentrisme, mercantilisme, codage, format universel, interface de saisie.*

*KEY WORDS: Unicode, typography, multilingual typesetting, scientific typesetting, ethnocentrism, mercantilism, encoding, universal text format, data acquisition interface*

---

Depuis l'irruption de l'informatique, la composition typographique se débat quotidiennement avec des problèmes de codage diversement compatibles entre eux, avec l'insuffisance de ceux-ci dès qu'il s'agit de traiter systèmes d'écriture ou types de composition un peu complexes (composition scientifique, multilingue...). C'est dire si Unicode est attendu aujourd'hui par les professionnels de ce secteur comme un enjeu technologique essentiel. D'autre part, la démocratisation de ces techniques, à travers la publication assistée par ordinateur, rend ce codage « universel » d'autant plus crucial qu'on peut difficilement attendre de profanes l'effort d'apprendre des méthodes de transcodage complexes et fastidieuses.

Pour autant, on ne saurait se passer d'une réflexion critique vis-à-vis d'un standard dont l'impact — technologique, mais, surtout, culturel — est encore difficile à évaluer, mais certainement considérable.

Typographe, je suis intéressé au développement d'Unicode à double titre : en tant que compositeur, j'en serai utilisateur et j'ai des besoins à satisfaire ; en tant que dessinateur de caractères, je dois faire correspondre des glyphes réels aux caractères virtuels d'Unicode. En cela, le typographe participe — devrait participer — à la définition du standard dans sa réalité visible, pour répondre aux besoins de l'utilisateur (c'est-à-dire, au final, du lecteur).

Lorsque j'ai commencé à envisager le projet d'une fonte de labeur<sup>1</sup> couvrant toute la partie occidentale d'Unicode (latin, grec, cyrillique), ainsi que les caractères scientifiques et techniques, l'examen du standard dans le détail m'a été nécessaire. L'occasion m'en a été donnée par Patrick Andries qui a proposé aux participants de la liste *Typographie*<sup>2</sup> de donner leur avis sur la traduction du standard, pour le livre qu'il écrit sur le sujet<sup>3</sup>. J'ai ainsi pu mesurer le chemin accompli — considérable — mais aussi, à travers un grand nombre d'erreurs manifestes dans le standard et sa terminologie, les dangereuses tentations et les limites d'Unicode.

## 1. Tentations d'Unicode

Unicode, il est peut-être utile de le rappeler, même s'il a une relation directe avec la norme ISO-CEI 10646 (JUC, jeu universel de caractères), n'est pas une norme internationale, mais un standard établi par un consortium à vocation commerciale dominé par des entreprises américaines. À ce titre, il n'est pas exempt d'arrière-pensées mercantiles et ethnocentriques.

---

1. S'il existe aujourd'hui des centaines de milliers de fontes numériques, une infime partie de celles-ci peuvent réellement être utilisées pour composer du labeur, c'est-à-dire du texte à lire, livres ou presse, par opposition à la titrairie, texte à voir, destinée à la composition d'affiches, de titres, etc.

2. <https://www.irisa.fr/wws/info/typographie>.

3. <http://iquebec.ifrance.com/hapax/>.

### 1.1. *Ethnocentrisme*

Il se manifeste en premier lieu dans la prise en compte inégale des langues ou systèmes d'écriture minoritaires. Plus précisément, on peut dire qu'il vaut mieux être minoritaire aux États-Unis ou en Europe qu'en Afrique, par exemple. Le cherokee ou l'inuit, mais aussi des écritures mortes qui n'intéressent plus aujourd'hui qu'une infime minorité de chercheurs, comme le gotique<sup>4</sup> ou l'étonnant déséret<sup>5</sup>, sont correctement gérés par Unicode, alors que le tfinagh, l'écriture des berbères (plus de 30 millions de locuteurs dans une dizaine de pays d'Afrique), en est toujours totalement absent. C'est que le traitement des écritures sans enjeu économique dépend surtout de l'initiative de bénévoles ou de subventions, un comble de la part d'un consortium commercial qui comprend notamment la plus grosse fortune mondiale...

Plus subtilement, Unicode perpétue des archaïsmes typographiques qui sont la manifestation de l'hégémonie culturelle de certaines nations. Ainsi, le traitement des guillemets, basé sur la notion, certes traditionnelle, mais artificielle, de guillemets « anglais », « français », et « allemands » (voir encadré ci-dessous).

Les typographes ont pris l'habitude de répartir — par commodité de langage, il faut bien le dire — les guillemets en trois paires, nommées guillemets « anglais » (“citation”), guillemets « allemands » („citation“), guillemets « français » (« citation »).

Or on voit que ces guillemets ne forment en fait pas trois paires, car le guillemet « allemand » fermant est un guillemet « anglais » ouvrant qui appartient déjà à la paire précédente. Son codage (201F) constitue donc un doublon.

Résultat : le concepteur d'Andale Mono WT Chart, fonte de démonstration d'Unicode, utilisée dans les tableaux de Patrick Andries, s'est fourvoyé en tentant d'interpréter « guillemet-virgule supérieur culbuté ». Il a fait un guillemet-apostrophe réfléchi, ce qu'on ne saurait obtenir en culbutant<sup>6</sup> un guillemet-virgule !

En réalité, chaque langue ou presque a des usages spécifiques des quelques signes suivants :

◀◀	00AB Guillemet gauche
▶▶	00BB Guillemet droit
““	201C Guillemet-apostrophe double culbuté
””	201D Guillemet-apostrophe double
„„	201E Guillemet-virgule double inférieur
””	201F Guillemet-virgule double supérieur culbuté <sup>7</sup>

Le codage superflu semble surtout manifester une volonté d'établir trois paires fixes aisément manipulables, au mépris d'usages différents de ces guillemets dans d'autres langues européennes.

4. Le gotique, à ne pas confondre avec la gothique, était l'écriture des Goths. Bien qu'il ne comporte que très peu de signes distincts du grec et du latin de l'époque, il fait l'objet d'un codage spécifique intégral dans Unicode. ▷

L'organisation d'Unicode en blocs séparés semble aussi propager l'idée de langues dominantes simplement élargies pour tenir compte de leurs vassales (c'est particulièrement évident dans l'organisation des blocs du latin, en cercles concentriques, du plus occidental – le « latin de base » qui ne peut servir qu'à l'anglais – au plus exotique – le « latin étendu B », où l'on jette en vrac les lettres spécifiques des langues africaines ou asiatiques).

La séparation nette entre latin, grec et cyrillique peut paraître raisonnable, mais c'est négliger les interactions qui ont existé et existent encore entre les langues et la perméabilité des barrières entre langues et même entre alphabets différents. On peut se demander quelle nécessité il y a à découper en morceaux le grand ensemble des écritures alphabétiques européennes. Sur le plan pratique, on aurait pu penser qu'il serait plus simple pour tout le monde de regrouper toutes les variantes d'une même lettre, par exemple. Plus encore, il est absurde, même si ça a parfois des raisons historiques, de devoir chercher dans un autre bloc la majuscule d'une lettre (et inversement), comme dans le cas de notre « Ÿ », dont la minuscule, 00FF, se trouve dans le bloc « supplément latin-1 », alors que sa majuscule, 0178, se trouve dans le bloc « latin étendu A ».

## 1.2. *Mercantilisme*

La manifestation la plus évidente de cette orientation pernicieuse d'Unicode est l'intégration de caractères propriétaires, voire de fontes entières. L'exemple le plus flagrant est celui du bloc 2700, entièrement constitué des glyphes d'ITC Zapf Dingbats. Cette fonte ne constitue qu'un assemblage, certes utile (encore qu'on se demande si on a vraiment l'usage de 19 (!) variantes d'étoiles ou de soi-disant astérisques, d'origines diverses — on y trouve l'étoile de David, qui devrait être codée avec les autres symboles religieux), mais arbitraire, de symboles divers, qu'un fondateur prestigieux (Adobe) fournit gratuitement avec un grand nombre de ces produits, notamment les licences PostScript pour les imprimantes. Le plus étonnant est que cette intégration n'a pas été faite à la demande d'Adobe, mais de l'éditeur d'un traitement de textes aujourd'hui disparu (WordPerfect) ! Peut-on baser un standard aussi important pour la culture du monde sur des considérations mercantiles aussi délétères ?

---

5. Le déséret est un alphabet phonémique synthétique inventé en 1850 à l'usage des mormons pour écrire l'anglais. Son usage très limité (4 livres !) cessera d'être encouragé par l'Église de Jésus-Christ des saints des derniers jours en 1869...

6. Dans la terminologie Unicode, « culbuté » indique une rotation de 180°, « réfléchi » une symétrie sur l'axe vertical, « renversé » une symétrie sur l'axe horizontal. Les signes culbutés sont beaucoup plus fréquents, car ce sont les seuls qu'il était facile d'obtenir au plomb (en mettant le caractère à l'envers).

7. Appeler virgule supérieure ce qui est évidemment une apostrophe aboutit à une impasse terminologique et oblige à nommer la virgule elle-même inférieure, ce qui donne de sérieux doutes sur sa position (sur la ligne de base, en dessous ?). Il y a toutefois un progrès par rapport à la version anglaise précédente, qui nommait « virgule supérieure » le guillemet-apostrophe lui-même, et le guillemet-virgule « guillemet en double-neuf inférieur », ce qui aboutissait au surréaliste « guillemet en double-neuf supérieur culbuté »...

Beaucoup des signes de ces blocs n'ont ainsi pas de valeur sémantique spécifique et dérogent en ce sens à la logique « caractère » affirmée dans le reste du standard : « Unicode traite les caractères du casseau différemment de tous les autres caractères. [...] les caractères Zapf Dingbats correspondent à un ensemble de glyphes provenant d'une police de caractères spécifique et leur valeur sémantique est leur forme. »

Les conséquences de cette entorse ne se sont d'ailleurs pas fait attendre. Zapf Dingbats contient quelques variantes de deux coches, ✓ et ✕, constituant à elles seules un système logique conventionnel : quand on vérifie une liste, on peut souhaiter indiquer que la vérification est faite et que l'objet est correct (✓) ou pas (✕). Or on vient d'intégrer dans le bloc « ponctuation » le signe 2052 « Signe moins commercial » (?) — avec la forme proposée suivante : % (incohérente avec sa terminologie) —, qui nous a posé des problèmes de traduction. Sont données comme variantes de ce signe V, x et % ! Il semble qu'on ait mélangé plusieurs équivalences graphiques de cette convention, compréhensibles dans des systèmes typographiques pauvres, comme la dactylographie. Or, c'est la convention la plus sémantiquement contradictoire qui est à l'origine de la terminologie du signe proposé. Et tout ceci alors que la convention est déjà représentée. Mais inutilisable, si l'on considère que « [...] leur valeur sémantique est leur forme »...

Système typographique normal	Équivalences graphiques en systèmes typographiques pauvres (dactylographie, courrier électronique...)			Sémantique
✓	V	√	- (!)	correct (positif)
✕	X	% ou %	+ (!)	incorrect (négatif)

Mais cette logique mercantile peut avoir des conséquences bien plus graves. Bien entendu, l'ethnocentrisme en est une des plus inquiétantes. D'une façon générale, il est à craindre que soient sciemment négligés des langues et des codes graphiques spécialisés pas assez rentables. Il est d'ores et déjà flagrant que l'on néglige les besoins du compositeur au profit de l'utilisateur bureautique, marché beaucoup plus attractif. Comment expliquer, sinon, l'absence de consultation des spécialistes du signe que sont les typographes, évidente par le grand nombre d'erreurs<sup>8</sup> que j'ai pu repérer à la première lecture ? et que soient toujours absents du standard nombre de signes (petites capitales, lettres supérieures, chiffres bas de casse, ligatures, etc.) dont la profession réclame une gestion facilitée à cor et à cri depuis des années ? Et s'il est aussi flagrant que les typographes n'ont pas été consultés dans l'élaboration du standard, on peut se demander ce qu'il en est des autres disciplines impliquées dans la gestion de l'écriture...

Le cas de l'alphabet phonétique international (A.P.I.) est, à ce titre, édifiant. Cet alphabet, composé d'emprunts aux alphabets latins et grecs, comporte aussi un bon nombre de signes culbutés. Il était beaucoup plus facile et moins coûteux, pour le compositeur au plomb, de retourner des plombs dans le composteur pour disposer des

8. On pourra constater, en les comparant, que Patrick Andries a grandement amélioré la traduction du standard par rapport à l'original, en intégrant, soit dans la traduction, soit dans les annotations, beaucoup de corrections suggérées par les personnes qu'il a consultées, notamment des participants de *Typographie*.

nouveaux signes que de les faire graver (figure 3). Même si cette pratique s'est perpétuée dans le jargon des linguistes, il n'était pas souhaitable, pour des raisons didactiques, de la prolonger dans un standard destiné à l'informatique. A-t-on vraiment interrogé les linguistes sur la signification réelle de ces signes ? On peut légitimement se le demander.

Un autre aspect du mercantilisme qui sous-tend la démarche du consortium concerne les motivations même d'Unicode. Au-delà de la résolution des problèmes de transcodages, l'obsession, parfois en dépit du bon sens, de préserver le « texte sous-jacent » reflète aussi l'ambition avouée de faire du texte un matériau dont on gère les flux et les traitements (traduction, mise en page, etc.) comme ceux de n'importe quel autre produit, avec en point de mire l'automatisation de tous ces traitements. Même si nous, professionnels, ne nous sentons pas réellement menacés par ce point de vue, irréaliste, cette volonté de faire d'un artisanat une industrie ne peut que nous inquiéter, non pas tant pour notre devenir social que pour la qualité de notre travail.

Il paraît légitime, dès lors, de se demander qui contrôlera le travail du consortium, qui sanctionnera d'éventuels abus de position dominante (rappelons que Microsoft en fait partie...), qui s'assurera d'un niveau de qualité suffisant et d'un suivi correct de l'utilisateur final ? La réponse est, bien sûr, éminemment politique.

## 2. Limites d'Unicode

Toutes ces objections nous renvoient aux limites, à tous les sens du terme, d'Unicode, limites de ce que l'on peut gérer au niveau d'un codage, limites que l'on doit lui poser, limites plus ou moins justifiées dans sa définition actuelle.

### 2.1. Caractère ou glyphe ?

Dans les nombreuses discussions que nous avons eues à ce sujet, l'une des critiques qui fut exprimée le plus souvent sur la liste *Typographie* est celle de la logique caractère, strictement sémantique, d'Unicode, opposée à notre réalité quotidienne d'une logique de glyphe. Même si nous comprenons parfaitement les nécessités liées au choix de cette logique au départ, certains effets pervers sont pour nous très dommageables.

En effet, un standard essentiel comme Unicode aura forcément valeur prescriptive. Tout ce qui en sera absent risque d'être pratiquement condamné à disparaître. Inversement, la présence injustifiée d'un signe inciterait à un usage fautif. Ainsi, le symbole du franc (20A3 F), invention ubuesque récente du ministère des Finances, n'a jamais été utilisé dans la réalité (et présente, dans la fonte des tableaux de P. Andries, la forme F — issue des spécifications de Microsoft —, elle-même inexacte !).

Le codage superflu des lettres accentuées du grec moderne, déjà présentes dans le bloc « grec étendu », a incité le concepteur d'Andale à donner à l'accent aigu du grec la forme d'une *quote* (') au lieu de sa forme normale (´), suivant en cela, il est vrai, une réforme contestée — ça devait l'arranger pour l'insérer au sein du tréma grec (¨).



raison des formes différentes, et, dans le cas contraire, l'identité de forme de signes sémantiquement distincts dans le standard n'est pas pour faciliter le travail de l'opérateur chargé de saisir un manuscrit. Sans compter les problèmes terminologiques que posent la distinction entre caractères sémantiquement différents mais homoglyphes. Ainsi, les doublons et triplets partagés entre les blocs cyrilliques, grecs et latins donnent lieu à des dénominations parfois surréalistes, comme « 0263 Lettre minuscule latine gamma », ou « 0460 Lettre majuscule cyrillique oméga » (*sic*).

De même, le bloc 2150 « formes numériques » code les chiffres romains de 1 à 12 (pourquoi jusqu'à 12, pour les cadrans d'horloge ?), ce qui laisse rêveur puisque un enfant de 12 ans sait que tous les chiffres romains peuvent être composés avec les lettres I, V, X, L, C, D et M (7 lettres). Il paraît que c'est pour la compatibilité avec une norme extrême-orientale...

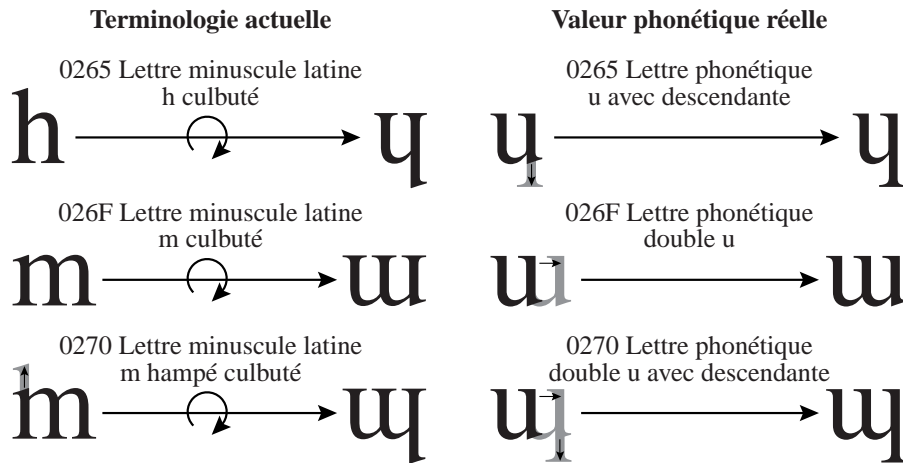
Inversement, le standard déroge souvent à sa logique et peut oublier de distinguer des signes dont l'identité formelle n'est pas avérée. Ainsi, on remarquera l'absence des esprits du grec. C'est que l'on n'a pas jugé utile de les distinguer des diacritiques « 0313 Diacritique virgule en chef », et « 0314 Diacritique virgule réfléchie en chef ». Or, s'il est vrai que l'esprit doux est identique au signe marquant l'élision en grec, la *coronis*, proche, dans sa forme, de notre apostrophe, il est erroné de croire que la *coronis* est une apostrophe. En effet, si l'esprit ou la *coronis*, dans des fontes classiques, prend souvent la forme d'un point plus ou moins prolongé d'un petit trait en arc de cercle, comme notre apostrophe, quand, dans des fontes linéales modernes, l'apostrophe latine peut adopter la forme d'un point carré prolongé d'un trait droit (Helvetica), voire celle d'un accent aigu (Futura, Optima), l'esprit ou *coronis* doit, lui, toujours conserver sa forme fondamentale, qui est une demi-lune. L'esprit et la *coronis* ne sont donc pas strictement homoglyphes de l'apostrophe latine. Identifier ces diacritiques aboutirait, pour des fontes modernes, à ne pas disposer de diacritiques utilisables pour le grec.

latin		grec
“C’est l’été”	Bodoni italique	ἐπ’ ἐμὲ λέγων, Leipsias
“C’est l’été”	Helvetica	ἐπ’ ἐμὲ λέγων, Belles Lettres
“C’est l’été”	Futura	ἐπ’ ἐμὲ λέγων, Greek Sans 486
“C’est l’été”	Optima italique	ἐπ’ ἐμὲ λέγων, Porson

**Figure 2.** Formes distinctives de l'esprit doux et de la coronis, et de l'apostrophe

La terminologie même peut aussi être source d'interprétations fautives. Le cas des signes culbutés de l'A.P.I. montre comment une nomenclature approximative induit des formes inappropriées — encore aujourd'hui, ce sont ces formes qui prévalent dans les fontes A.P.I. existantes, en contradiction avec la sémantique de ces signes (figure 3).





**Figure 3.** Formes et significations réelles de certaines lettres culbutées de l'A.P.I.

Ici, on voit que la terminologie provoque une interprétation sémantiquement incorrecte des signes. Les formes culbutées traditionnelles, qui sont pourtant de bonne approximations, ne rendent pas totalement justice à ces voyelles ou semi-voyelles dérivées du « u » (le « h culbuté » note le son « u » courant du français).

On constate aussi que le standard s'y perd parfois entre les terminologies de traditions typographiques différentes. Le bloc 2000 « ponctuation générale » en est une parfaite illustration. On y trouve, par exemple, quatre tirets, 2012 « Tiret numérique », 2013 « Tiret demi-cadratin », 2014 « Tiret cadratin<sup>9</sup> », 2015 « Barre horizontale ». En réalité, sémantiquement parlant, il n'existe qu'un seul tiret. Simplement, les anglosaxons ont pris l'habitude de lui donner une largeur d'un demi-cadratin, alors qu'en Europe on préfère un cadratin. Il s'agit donc de deux glyphes du même signe. En fait, nous sommes contents qu'il y ait les deux, pour la composition bilingue, et nous avons pris l'habitude, puisqu'il est déjà disponible dans les fontes européennes, d'utiliser ce tiret demi-cadratin pour indiquer des intervalles (par exemple dans des dates), rôle maintenant attribué au tiret numérique. Mais que vient faire là cette « barre horizontale » ? Patrick Andries précise que, selon le standard, elle « indique un changement d'interlocuteur dans les dialogues ». En fait, il est probable qu'on a voulu, pour réintroduire artificiellement la logique sémantique, distinguer les deux usages conventionnels du tiret : introduire les articles d'une énumération ou les répliques d'un dialogue, d'une part, encadrer les incises, d'autre part. Mais nous avons toujours utilisé le même tiret pour ces deux usages, demi-cadratin outre-atlantique, cadratin ici, car la largeur du tiret n'est pas perçue par le lecteur. Nous avons donc deux variantes de glyphes du tiret (qui se justifient) et une variante sémantique (dont on sait que faire).

9. Le cadratin, unité fondamentale en typographie, est un carré de dimension égale à la force de corps, c'est-à-dire qu'un cadratin de corps 12 fait 12 points de côté.

Pour les espaces, c'est encore pire. D'abord nous avons les espaces cadratin et demi-cadratin. Pas de problème. Ensuite on trouve un cadrat et un demi-cadrat, ce qui ne manque pas de sel, ceux-ci étant utilisés autrefois pour remplir les lignes creuses d'une composition au plomb (pour éviter qu'elles ne tombent de la forme), fonction difficilement transposable en composition informatique ! Ensuite, on trouve un tiers, un quart et un sixième de cadratin, une espace numérique, une espace-ponctuation, une espace fine et une ultra-fine. Ici, il faut comprendre que les anglo-saxons et certains européens ont suivi des logiques différentes : en France, par exemple, les chiffres faisait traditionnellement un demi-cadratin, et les séries de trois étaient espacés d'une fine, également sous-multiple du cadratin, comme suit : « 3 000 000,00 F » ; aux États-Unis, par contre, les chiffres avaient une chasse commune arbitraire, et les séries de trois étaient espacés d'une virgule (dont la chasse est aussi arbitraire) : « \$3,000,000.00 ». Pour rattraper des alignements dans un tableau de chiffres, au plomb, les américains avaient donc besoin d'espaces relatives (espace-chiffre, espace-ponctuation), là où nous utilisons des espaces absolues (sous-multiples du cadratin). Mais leur valeur sémantique est plus ou moins la même. Par ailleurs, la chasse de l'espace-mot et de la fine ont variés selon les usages et les matériels, la première faisant généralement un quart ou un tiers de cadratin, et la deuxième la moitié ou le tiers de la précédente. Quand à l'ultra-fine, c'est simplement la plus fine des espaces disponibles (1/2 point, généralement), utilisée pour éviter que certains éléments soient trop rapprochés (par exemple, un appel de note du mot qui le précède). Bref, on voit que sont mélangées ici différentes logiques, ce qui ne nous dérange pas, d'ailleurs, mais risque de dérouter l'utilisateur non averti (et les programmeurs des logiciels de composition, ce qui est plus grave). Il aurait sans doute été plus rationnel de coder plus de sous-multiples du cadratin (on n'en a jamais assez) et de laisser à la charge du logiciel de régler la valeur des espaces modales en fonction des besoins généraux ou ponctuels de l'utilisateur.

## 2.2. Codage ou format ?

La question des limites devient cruciale quand on commence à aborder le problème des variations de casse, particulièrement dans le domaine scientifique. Ainsi, le bloc 2070 comprend tous les chiffres arabes en exposant et en indice. D'un point de vue typographique, ça paraît à première vue une bonne idée, puisque la graisse de ces chiffres doit être corrigée en fonction de leur taille relative au corps courant. Mais on remarque ensuite que le bloc comprend également parenthèses, signes plus et moins, et la lettre *n*...

Il devient très vite évident, si l'on y réfléchit un peu, que l'on est là devant le tonneau des Danaïdes : n'importe quel signe pouvant avoir une forme exposant et indice, voire exposant ou indice de *n*-ième niveau, les possibilités sont virtuellement infinies. Il est donc impossible de gérer totalement cet attribut typographique au niveau du codage lui-même. Partant de là, il eut peut-être mieux valu ne pas commencer une intégration qui ne pourra être que partielle, pour éviter une différence de rendu entre les signes prévus par le standard et ceux qui devront, par la force des choses, être intégrés par des procédures de niveau supérieur.

Inversement, le problème des petites capitales est, lui, clairement délimité (il ne concerne que les lettres des écritures bicamérales européennes) et aurait pu être géré efficacement par le codage, si le standard ne campait pas là sur une position dogmatique.

<i>Fuligo septica</i> (L <small>INNÉ</small> ), repéré [...]	Petites capitales obtenues par réduction
<i>Fuligo septica</i> (L <small>INNÉ</small> ), repéré [...]	Petites capitales obtenues par réduction et graisse supérieure
<i>Fuligo septica</i> (L <small>INNÉ</small> ), repéré [...]	Petites capitales du jeu expert

**Figure 4.** Les petites capitales, de valeur orthotypographique réelle, ne peuvent pas, quoiqu'en laisse penser de nombreux logiciels, être gérées correctement par réduction. La différence de graisse et de chasse avec des capitales réduites peut être compensée, mais pas le dessin des accents. Il faut donc faire appel à des fontes spécifiques, les jeux expert. Problème partiellement résolu par le format OpenType, mais liant leur usage à des procédures de niveau supérieur, qui pourraient être évitées par leur codage dans Unicode.

Il reste donc une réflexion à faire sur ce type de rendus et leur gestion. La réponse la plus évidente est qu'Unicode doit être complété par un format universel de texte enrichi, comme ambitionne de l'être XML, par exemple. Mais c'est aussi aux typographes et aux fondeurs qu'il incombe de proposer des solutions à certains problèmes que le codage n'a pas à gérer. Par exemple, les rendus par réduction des indices et exposants peuvent être grandement améliorés si la progression des graisses de fontes de labeur dédiées à la composition scientifique est calculée pour permettre la compensation de taille par l'usage d'une graisse supérieure (figure 5, a, b et c).

### Garamond Adobe

$n^x = 10^{-3}$ , dit M <sup>lle</sup> VIEILLEFILLE...	Exposants par réduction du corps
$n^x = 10^{-3}$ , dit M <sup>lle</sup> VIEILLEFILLE...	Exposants par compléments experts
$n^x = 10^{-3}$ , dit M <sup>lle</sup> VIEILLEFILLE...	Exposants par réduction du corps et augmentation de graisse

### Baskerville Monotype

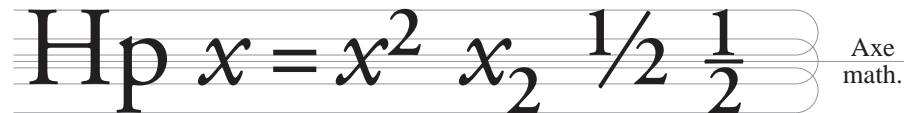
$n^x = 10^{-3}$ , dit M <sup>lle</sup> VIEILLEFILLE...	Exposants par réduction du corps
$n^x = 10^{-3}$ , dit M <sup>lle</sup> VIEILLEFILLE...	Exposants par compléments experts
$n^x = 10^{-3}$ , dit M <sup>lle</sup> VIEILLEFILLE...	Exposants par réduction du corps et augmentation de graisse

**Figure 5a.** Comparaison de différentes méthodes de rendu des exposants

L'un des objectifs de mon travail est justement de démontrer qu'il est possible d'améliorer le rendu des lettres réduites (exposants, indices, petites capitales) par

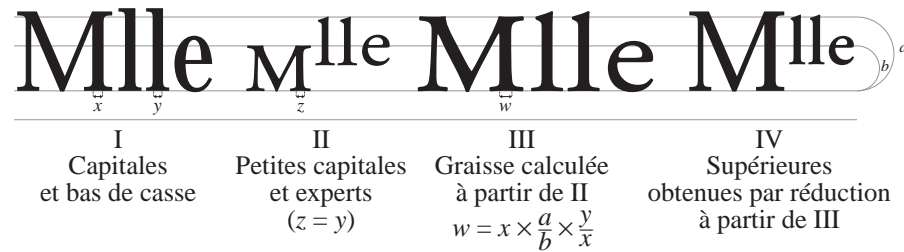
l'emploi d'une graisse supérieure, si elle est appropriée (c'est le cas, figure 5a, en Garamond, pas en Baskerville), et de déterminer les règles que cela implique.

On peut ainsi déterminer une relation naturelle d'égalité de hauteur entre indices, exposants et bas de casse, ce qui rend possible le calcul d'une graisse commune idéale pour le rendu de ces signes.



**Figure 5b.** Relation naturelle entre bas de casse, exposants et indices

Et en déduire des règles pour la progression des graisses, jusqu'ici arbitraire.



**Figure 5c.** Calcul de graisse pour le rendu des exposants et indices

De même, une meilleure gestion des petites capitales est possible, si l'attribut est géré au plus bas niveau, comme l'est l'italique à l'heure actuelle. Il est donc souhaitable de poser des limites au standard, en relation avec l'établissement parallèle d'un format adapté et avec une nouvelle approche du travail du fondeur de caractères.

### 2.3. Avec quoi compose-t-on ?

Le tapage autour d'Unicode ne doit pas nous faire oublier qu'un codage ne signifie pas grand-chose sans les fontes pour l'exprimer et qu'un caractère sans glyphe n'est que du bruit. À l'heure actuelle, il existe très peu de fontes couvrant une partie substantielle d'Unicode, et pas de vraie fonte de labeur, à part peut-être le Sylfaen de John Hudson. On peut d'ailleurs douter qu'il en existe un jour une couvrant tout Unicode. Si le travail que j'ai entrepris vise notamment à fournir un modèle de travail pour la partie occidentale d'Unicode, il va de soi que l'intégralité de celui-ci ne sera jamais couverte que par des assemblages de fontes, à l'instar des métafontes de T<sub>E</sub>X. Or, si le nouveau format de fonte, OpenType, apporte des éléments de solution, il n'existe pas encore d'outil pour permettre à l'utilisateur de gérer *lui-même* efficacement ces assemblages (et nous permettre de continuer à utiliser nos chères – très chères –

typothèques). On a le sentiment que la logique qui prévaut est toujours la conception des fontes sous Windows : tout ce qui n'est pas du latin, c'est des symboles... L'utilitaire Adobe Type Reunion a constitué, en son temps, un progrès important : combiné à l'utilisation de balises de styles, il permettait de regrouper les différentes graisses d'une même fonte en un ensemble fonctionnel, accessible par un sous-menu ou des raccourcis clavier. De même, ce dont nous avons besoin, aujourd'hui, c'est de pouvoir associer facilement des séries de glyphes à des séries de caractères, et cela, aussi bien horizontalement (affectation d'une série de glyphes à leurs caractères correspondants, figure 1 ❶) que verticalement (correspondance d'une série de variantes de glyphes à un seul caractère, figure 2 ❷ et ❸). Ceci permettrait d'aller bien au-delà de la simple (re)conversion de fontes comme cela est proposé par les *glyphnames conversions* d'Adobe, qui, de toute façon, ne peuvent gérer que les codages déjà précédemment standardisés. Par exemple, il n'existe pas, hors d'Unicode, de codage standard pour les écritures latino-africaines, il est donc impossible de s'appuyer sur les *glyphnames conversions* pour convertir les fontes actuellement utilisées pour celles-ci.

On peut très bien imaginer un système d'affectation bidirectionnel qui permettrait en outre, à l'aide de conventions et de balises préétablies dans un format universel, de

Affectation horizontale ❶

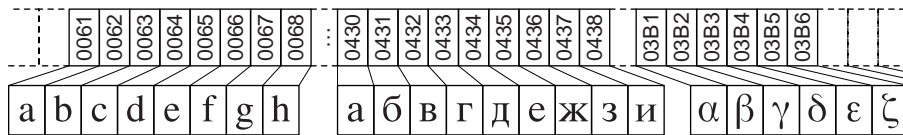
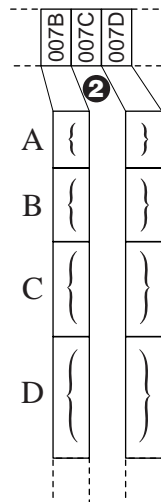


Figure 6a. Affectation primaire de glyphes à des caractères Unicode

Affectation verticale ❷



Affectation horizontale et verticale ❸

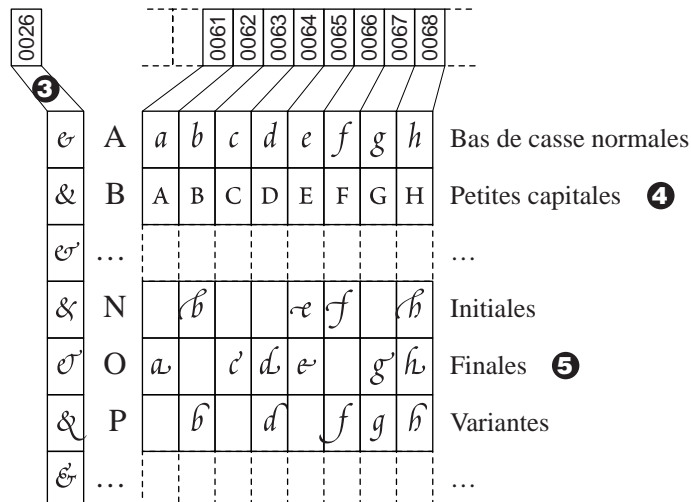


Figure 6b. Affectation secondaire de glyphes à des caractères Unicode

gérer efficacement des attributs mal pris en charge aujourd'hui, comme les petites capitales ④, par exemple, ou les différentes variantes usuelles d'une scripte ⑤. On voit comment un ensemble typographique complexe, comme le Poetica ③, ④ et ⑤, pourrait y gagner en simplicité d'emploi.

#### 2.4. *Comment saisit-on ?*

Même si Microsoft a prouvé, hélas, qu'il est possible de fidéliser des utilisateurs malgré une ergonomie de saisie désastreuse, Unicode risque de rester lettre morte – si j'ose dire – si on en néglige cet aspect pratique. S'il est encore possible de mémoriser quelques codes à trois chiffres, on ne peut demander à un opérateur de retenir des codes à quatre chiffres – et en hexadécimal, qui plus est – simplement pour insérer un alpha ou une case à cocher dans un texte. L'interface de saisie est un élément essentiel de l'accès du public à Unicode. Le pire, dans ce domaine, serait de cantonner Unicode à une sorte de table des caractères géante, qu'il faudrait appeler chaque fois qu'on sort de Latin-1.

Les solutions existent déjà, qu'on les nomme « support multilingue » sous Windows, ou « WorldScript » sous MacOS. Rappelons que ces systèmes permettent, entre autres, de reconfigurer le clavier en fonction de la langue choisie. Encore faut-il documenter correctement ces ressources et en permettre la personnalisation aisée. Or, l'édition des claviers reste étonnante sous Windows et, si elle était relativement aisée sous MacOS, les nouveaux formats de ressources clavier Unicode (uchr) ne sont toujours pas documentés et ne disposent pas de l'éditeur de ressource des anciens (kchr). Car la reconfiguration de claviers nécessite deux niveaux de localisation : il faut faire correspondre les signes et ressources d'une langue source avec la configuration clavier spécifique d'une autre langue cible (qui, en dehors des États-Unis, n'est pas celle d'origine du système). La plupart du temps, on ne dispose que des correspondances de la langue source avec le qwerty américain. Saisir dans une langue étrangère n'est déjà pas facile, mais si en plus elle est translittérée pour un clavier différent du nôtre, cela devient vite très dissuasif... C'est pour les langues minoritaires, comme les langues africaines et asiatiques utilisant l'alphabet latin complété de lettres spécifiques, que le problème risque d'être le plus crucial. En l'absence des ressources claviers spécifiques, qui ont peu de chances d'être développées, pour des raisons mercantiles, il est à craindre que leur expression soit fortement handicapée par la nécessité de naviguer, à la saisie, entre des blocs différents. Si rien n'est fait, Unicode pourrait contribuer à accentuer leur isolement, alors qu'il devrait, en fait, constituer pour elles une opportunité unique de développement et d'ouverture (voir l'article d'Andrei Popescu-Belis).

Enfin, un facteur qui ne doit pas être négligé est celui du balisage des langues. Lorsque l'on fait de la composition multilingue, il peut être crucial de spécifier dans quelle langue est composé tel ou tel passage d'un texte, par exemple lors de l'utilisation de vérificateurs orthographiques. Ce balisage, effectué *a priori* ou *a posteriori*, devrait faire partie du format de document. Il permettrait ainsi l'emploi des dictionnaires appropriés, des formatages corrects des nombres, des ordres de tri, etc., ainsi qu'une bascule facilitée dans la configuration clavier utile (translittérée ou non). Combiné avec

l'efficacité d'un utilitaire comme PopChar<sup>10</sup>, cela permettrait aussi de supprimer totalement de la saisie la frappe de codes abscons, qui rebute l'utilisateur et fait perdre inutilement des millions d'heures de travail.

D'une façon générale, on a aujourd'hui la sensation à la fois excitante et frustrante que le système de composition universel que nous attendons depuis si longtemps est à portée de main. Il ne reste plus qu'à assembler les morceaux encore éparpillés de ce (très gros) puzzle. Et à trouver le Gutenberg qui en fera un outil efficace...

En fin de compte, il est clair qu'Unicode en sera l'un des éléments essentiels et qu'il est, malgré des réserves importantes, dans l'ensemble bien conçu. Il ne lui manque guère que la touche d'humanité qu'il ne pourra acquérir qu'avec l'usage. Il est maintenant urgent que les utilisateurs – et notamment les professionnels du livre – en prennent possession, le confrontent à la réalité, et réclament les amendements nécessaires. En espérant qu'ils puissent s'exprimer et être écoutés, dorénavant.

---

10. Je cite ce petit utilitaire à titre d'exemple d'ergonomie : d'un clic, il affiche tous les glyphes de la fonte courante (sous Unicode, ce pourrait être le bloc linguistique correspondant à la configuration clavier sélectionnée), on glisse sur le caractère choisi, qui est inséré quand on relâche la souris. On peut même en profiter pour apprendre le raccourci-clavier, qui est affiché. C'est devenu, sous MacOS, un complément quasi indispensable à la saisie. Comparé aux outils fournis à l'heure actuelle pour gérer Unicode, il donne une idée du travail qui reste à accomplir.