



Date : 21/07/2006

Comment la localisation gêne les portails internationaux : jeux de caractères et accès international

Pierre Clavel

Bibliothèque nationale suisse

Berne, Suisse

Meeting:	77 UNIMARC
Simultaneous Interpretation:	Yes
<p>1. WORLD LIBRARY AND INFORMATION CONGRESS: 72ND IFLA GENERAL CONFERENCE AND COUNCIL 20-24 August 2006, Seoul, Korea http://www.ifla.org/IV/ifla72/index.htm</p>	

Résumé:

L'accès à des ressources électroniques disponibles dans un autre pays que le sien peut poser des difficultés techniques. Cet article montre quels obstacles perturbent l'accès au niveau des jeux de caractères. La plupart des logiciels et certains composants matériels, tels les claviers, sont localisés, c'est-à-dire déclinés en versions adaptées à une langue et à son usage dans une région déterminée. Les usagers sont maintenant habitués à cela et le considèrent normal. Bien qu'Unicode facilite grandement l'interopérabilité, les portails internationaux rencontrent des difficultés dans ce domaine à cause de différences significatives entre langues et/ou pays dans le traitement des données, qui continuent à gêner la recherche, l'affichage et le tri de données.

Cette étude a été menée dans le cadre du projet TEL-ME-MOR, avec le soutien de la Commission européenne.

1. Introduction

Plusieurs facteurs ont contribué à l'accroissement du nombre de portails, plateformes accédant simultanément à plusieurs bases de données. La répartition géographique et linguistique de leurs partenaires a également augmenté. Un facteur de cette croissance, à côté des performances améliorées des réseaux et ordinateurs, est certainement la diminution du nombre de jeux de caractères employés, exigeant ainsi moins de conversions pour permettre l'interopérabilité. Il reste cependant un facteur mineur et parfois ignoré, le plus important étant la plus grande disponibilité de documents en ligne permettant un accès de partout dans le monde. On pourrait se demander si les différences de langue ne seraient pas l'obstacle annulant les possibilités offertes par les progrès évoqués ci-dessus. Bien que la barrière des langues subsiste, son impact est atténué par le fait la plupart des bibliothèques ont des documents dans d'autres langues que la langue locale et que de nombreux documents en ligne, visuels et sonores, ne sont pas liés à une langue. Ils doivent néanmoins être décrits et indexés, ce qui fait ressurgir les mêmes questions de jeux de caractères. Bien que cet article se concentre sur les portails de bibliothèques, The European Library

(www.theeuropeanlibrary.org) en particulier, les questions discutées ici concernent également d'autres types de portails et de moteurs de recherche.

Le but de The European Library est d'offrir l'accès à plusieurs ressources par la combinaison d'un index centralisé et de recherches distribuées. Ces ressources sont des documents numériques dans divers formats et des notices bibliographiques provenant actuellement de 16 bibliothèques nationales européennes. Le but final est d'intégrer toutes les bibliothèques nationales des 46 pays membres du Conseil de l'Europe (page d'entrée en français à l'adresse www.coe.int/DefaultFR.asp, disponible en 32 autres langues). Ceci n'impliquera pas moins de 5 systèmes d'écriture différents: arménien, cyrillique, géorgien, grec et latin. Cet article n'aborde les questions de jeux de caractères que dans le cas de l'alphabet latin.

Le but du projet TEL-ME-MOR (The European Library: Modular Extensions for Mediating Online Resources, www.telmemor.net) est d'aider les bibliothèques nationales des Nouveaux Etats Membres de l'Union européenne à devenir membres à part entière de The European Library. L'une en était déjà membre, (Slovénie), trois l'avaient rejointe au moment d'écrire ces lignes (Estonie, Lettonie, Slovaquie) et les six autres devraient y entrer d'ici au début de 2007. Six autres partenaires, dont la Bibliothèque nationale suisse, contribuent à des tâches spécifiques du projet pour lesquelles ils disposent de compétences techniques ou administratives particulières.

2. Notions sur les jeux de caractères et les problèmes qu'ils peuvent poser

Les ordinateurs mémorisent et traitent l'information par une succession de *bits*, éléments de mémoire pouvant n'avoir que deux valeurs, 0 et 1. Pour agrandir le nombre de valeurs possibles, les bits sont combinés en groupes où le nombre de valeurs possibles est égal à 2 à la puissance du nombre de bits, puisque chaque bit supplémentaire double le nombre de valeurs possibles. Exemples:

2 bits: 00, 01, 10, 11 = 4 valeurs = 2^2

3 bits: 000, 001, 010, 011, 100, 101, 110, 111 = 8 valeurs = 2^3

etc.

Un groupe de bits peut ainsi représenter un caractère. La succession des valeurs possibles associées aux caractères en fait un jeu de caractères.

Chaque constructeur des premiers ordinateurs avait défini son propre jeu de caractères, mais le besoin d'une normalisation s'est fait sentir à la fin des années 1950 pour permettre l'échange de données.

Le premier jeu de caractères normalisé, ASCII (American Standard Code for Information Interchange), avait 7 bits, donc 128 valeurs possibles, et permettait de stocker correctement des textes en anglais. À ce stade, les langues employant un autre système d'écriture ou l'alphabet latin avec des accents sur certaines lettres n'avaient simplement aucun moyen d'être traitées correctement. ASCII fut rendu international en 1967 comme norme ISO 646, avec la définition de 10 codes de caractères réservés pour des variantes nationales.

La transmission de données devint progressivement suffisamment fiable pour que l'on pût employer le huitième bit pour des codes supplémentaires plutôt que comme bit de parité, ouvrant la voie à l'usage d'ordinateurs dans n'importe quelle langue disposant d'un système d'écriture alphabétique.

On développa un grand nombre de nouvelles normes à 8 bits pour enrichir la gamme des caractères disponibles et répondre aux besoins de divers langues et groupes d'utilisateurs. Environ 180 normes incluent l'ASCII et sont compatibles entre elles pour la première moitié des jeux de caractères mais pas pour la seconde, comme le montre l'exemple ci-dessous.

Une personne travaillant sur un PC configuré pour l'Europe de l'Ouest écrit la phrase suivante dans un simple fichier texte

'Le théâtre sera réparé après le mois d'août, là où même le plâtre paraît neuf.'

Si ce fichier est ensuite transmis et ouvert sur un Macintosh ou un très vieux PC également configuré pour l'Europe de l'Ouest mais employant MS-DOS, ou encore sur un PC configuré pour l'Europe de l'Est, cette phrase se lira, respectivement:

'Le thÈ,tre sera rÈparÈ aprÈs le mois d'ao°t, l‡ o~ mlme le pl,tre paraÓt neuf.'

'Le thŒtre sera rŒparŒ aprŒs le mois d'ao√t, l‡ o· mΩme le plŒtre paraet neuf.'

'Le théâtre sera réparé aprĉs le mois d'août, lġ où mĉme le plâtre paraġt neuf.'

Quelqu'un connaissant la langue employée a peut-être de bonnes chances de rétablir les lettres correctes en les devinant. Cependant, en plus de pouvoir considérer qu'un tel affichage n'est pas satisfaisant, on voit que chercher dans un catalogue avec un jeu de caractères incorrect ne permettra pas d'obtenir de bons résultats.

3. Première étape vers une solution

Les logiciels permettent maintenant de charger et d'employer plusieurs jeux de caractères à 8 bits simultanément mais cela exige de signaler quelque part le jeu de caractères employé pour tout segment de texte, à moins qu'il ne s'agisse du jeu de caractères par défaut. Le jeu de caractères par défaut peut cependant être différent d'un ordinateur à l'autre, ce qui perturbe leur interopérabilité car annoncer son jeu de caractères par défaut n'est pas présent dans tous les protocoles de communication. C'est pourquoi l'industrie et les organes de normalisation recherchent une solution amenant à un jeu de caractères unique et complet. Après des premiers travaux effectués séparément (et même conflictuellement) Unicode (www.unicode.org, l'initiative de l'industrie) et ISO 10646 (l'initiative des organes de normalisation) se mirent d'accord sur des normes où au moins les codes des caractères sont identiques. Elles diffèrent seulement au niveau d'informations supplémentaires concernant les caractères, Unicode leur attribuant plus de propriétés qu'ISO.

Unicode est un jeu de caractères à 16 et 32 bits destiné à contenir dans un jeu unique tous les caractères nécessaires. 16 bits permettent de définir 65'536 caractères, bien assez pour incorporer tous les alphabets et les syllabaires. Ce nombre a été étendu à 1'114'112 en définissant une section virtuelle à 32 bits destinée aux idéogrammes. Comme les jeux de caractères à 8 bits contenant ASCII, Unicode inclut Latin-1 (ISO 8859-1), le rendant compatible avec le jeu à 8 bits de l'Europe de l'Ouest sans autre conversion que l'insertion d'octets vides.

Unicode et ISO 10646 décrivent des caractères, pas des glyphes. L'exemple suivant le montre mieux qu'une explication:

Caractère	Code	Nom
В	U+0412	Lettre majuscule cyrillique ve
Β	U+0392	Lettre majuscule grecque beta
B	U+0042	Lettre majuscule latine b

En cyrillique, grec et latin, les lettres majuscules ve, beta et, respectivement, b ont exactement la même apparence. Elles emploient le même *glyphe*. Elles restent cependant des *caractères* différents, avec des codes différents, puisqu'elles sont employées dans des systèmes d'écriture différents, ce qui se remarque lorsqu'on voit leurs minuscules: в, β et b respectivement.

Dans les jeux de caractères à 8 bits, la relation entre le code d'un caractère et la valeur numérique effectivement traitée et mémorisée par les ordinateurs est simple : c'est la même dans (presque) tous les cas parce que les octets restent une unité de base des microprocesseurs. Unicode fait par contre une distinction entre le code d'un caractère en tant qu'abstraction et comment ce dernier est exprimé en bits, ce qu'on appelle son *modèle de codage* ou le mécanisme de sérialisation des caractères. Le choix d'un modèle de codage dépend de ce que l'on doit faire avec les caractères, par exemple les garder en mémoire ou leur faire subir un certain traitement, et les contraintes spécifiques à ces actions. Il y a 3 principaux modèles de codage, nommés UTF-32, UTF-16 et UTF-8.

Dans UTF-32, chaque caractère est encodé en 32 bits, ou 4 octets. Son emploi n'est réellement pertinent qu'avec des idéogrammes. Dans UTF-16, chaque caractère est encodé en 16 bits, ou 2 octets, sauf les caractères dont le code est supérieur à 65'535 (hexadécimal FFFF), où 4 octets sont nécessaires. Cet encodage est préféré dans des applications où la position d'un caractère au sein d'une chaîne doit pouvoir être prédite indépendamment des caractères qui le précèdent. Dans UTF-8, chaque caractère est encodé 1 à 4 octets selon un algorithme spécial. Les caractères ASCII emploient 1 octet, tous les autres caractères européens (y compris les systèmes d'écriture non latins) et du Proche-Orient 2 octets et les systèmes d'écriture asiatiques ainsi que certains caractères spéciaux comme les symboles mathématiques 3 ou 4 octets. Cet encodage est préféré lorsque l'espace mémoire ou la vitesse de transmission importe et est de ce fait répandu sur la toile.

La norme Unicode définit également certaines caractéristiques techniques des caractères. Chaque caractère a :

- Un *nom* normalisé.
- Une *catégorie générale*, telle que lettre, nombre, signe de ponctuation etc.
- Une *classe combinatoire canonique* : des signes diacritiques ou similaires peuvent être ajoutés à des lettres dans de nombreuses langues pour marquer une intonation ou modifier le son de la lettre. Ces signes sont codés individuellement dans Unicode et ce paramètre définit notamment où ils sont placés par rapport à leurs lettres porteuses.

- Une *classe bidi*, définissant le comportement directionnel du caractère par rapport à ses voisins dans un texte, de gauche à droite, de droite à gauche, ainsi que diverses options lorsque les deux directions cohabitent dans un texte.
- Des références aux codes d'autres caractères, comme les composants d'une lettre avec diacritique, la minuscule d'une majuscule et vice-versa, ainsi que la valeur numérique des caractères représentant des nombres.

Unicode fonctionne aujourd'hui sur toutes les plateformes et dans de nombreuses applications, y compris les bases de données et les imprimantes. Comme Unicode inclut les caractères de tous les autres jeux de caractères existants (excepté certains d'Extrême-Orient), une conversion sans perte d'information est possible de ces jeux vers Unicode, mais bien sûr pas en sens inverse.

4. Les jeux de caractères posent-ils encore des problèmes ?

Oui, même dans un monde où tout serait en Unicode, il reste des problèmes à prendre en compte dans les trois principales étapes de l'interaction d'un usager avec un portail ou un moteur de recherche :

1. Tenir compte des différentes manières d'indexer des caractères spéciaux d'une base de données à l'autre.
2. Saisir des termes de recherche comportant des caractères absents du jeu de caractères par défaut installé sur l'ordinateur de l'utilisateur.
3. Traiter les enregistrements renvoyés par le portail selon différentes attentes des usagers.

Variations dans l'indexation d'un même caractère

Presque toutes les langues employant le système d'écriture latin utilisent des signes diacritiques et/ou des caractères spéciaux, à l'exception de l'anglais. Les premiers systèmes pour bibliothèques, conçus par et pour des bibliothèques anglophones, appliquaient les règles d'indexation de l'anglais, où les signes diacritiques sont ignorés, tout comme la casse, et les caractères spéciaux convertis en la lettre ordinaire la plus proche : « ô », « ó » ou « ø » pouvant apparaître dans un nom ou un titre étranger étaient donc toutes réduites à « o » dans les clés d'index. Toutefois, cette pratique ne correspond souvent pas à l'usage en vigueur là où ces signes diacritiques et caractères spéciaux sont effectivement employés. De nombreuses langues ont des caractères spéciaux, y compris des combinaisons avec signes diacritiques, qui ont la valeur d'une lettre individuelle. Par exemple, « ø » n'est pas indexé comme un « o » en danois et en norvégien, « â » n'est pas équivalent à « a » en suédois, ni « Ł » à « L » en polonais. Plus grave, le même caractère peut demander différents traitements selon la langue dans laquelle il est employé : « ö » est employé en estonien, en allemand et en hongrois (parmi d'autres) avec des règles d'indexation différentes dans ces trois langues (respectivement conservé comme caractère individuel, converti en « oe » et en « o »). Les règles de tri varient également fortement, plusieurs caractères spéciaux étant placés après « z ».

L'évolution des techniques permet progressivement aux programmeurs de logiciels de tenir compte de ces variations culturelles, donnant naissance à une nouvelle discipline, la localisation. Aujourd'hui, les systèmes d'exploitation et les systèmes pour bibliothèques sont localisés, pour le plus grand confort des usagers. On remarquera cependant qu'ils sont localisés selon la langue du pays dans lequel ils se trouvent, sans tenir compte de la langue des documents indexés. En d'autres termes, le même nom ou titre contenant un « ö » sera indexé différemment dans une base de données estonienne, allemande ou hongroise, indépendamment de sa propre langue ou de son origine.

La table ci-après montre quels sont les caractères spéciaux ou lettres avec signe diacritique qui ne sont *pas* converties en la lettre ordinaire la plus proche dans les bases de données des bibliothèques nationales des Nouveaux Etats Membres. Il est extrêmement probable que cela correspond également au profil local¹ de ces pays et/ou de ces langues.

¹ Un profil local est un ensemble de paramètres influant sur le comportement d'un ordinateur de telle sorte qu'il corresponde aux usages locaux dans un certain nombre de domaines : langue par défaut et zone géographique, desquels seront tirés le jeu de caractères par défaut, l'ordre alphabétique, la disposition du clavier, mais également la ponctuation (tant autour du texte que dans les nombres), le symbole de la monnaie, et les formats habituels des dates, heures, prix etc.

Bibliothèque nationale de	Caractères spéciaux
République tchèque	Č Ř Š Ž dans la plupart des bases de données Dans certaines bases de données, Á Ď É Ě Í Ň Ó Ť Ú ù Ý sont également maintenus tels quels
Estonie	Š Ž Ő Ä Ö Ü
Hongrie, Lettonie, Lituanie	Aucun, les signes diacritiques sont ignorés dans l'indexation
Pologne	Ą Ć Ę Ł Ń Ó Ś Ź Ż
Slovaquie	Č Ř Š Ž
Slovénie	Č Š Ž

Saisie et distribution des requêtes

Les chapitres précédents de cet article ont montré que la diversité des normes, tant dans les jeux de caractères que les règles d'indexation, est susceptible de compromettre l'interopérabilité. Lorsque les ressources électroniques n'offraient que du texte, la langue était le principal obstacle et rendait ces problèmes peu visibles : si l'on sait une autre langue que la sienne suffisamment bien pour en lire des textes, on connaît probablement tous ses caractères et comment on les traite. Maintenant que des images et de la musique sont plus largement disponibles, la langue n'est plus tant un problème pour l'utilisation du document mais le subsiste lors de la recherche ; la localisation est un nouveau challenge.

Tout d'abord, les méta-données des ressources ne sont pas forcément encodées dans le même jeu de caractères, rendant des conversions nécessaires. Plusieurs cas doivent être considérés :

1. Si tous les caractères d'un terme recherché encodé en Unicode ont un équivalent dans les jeux de caractères des bases de données interrogées, une conversion sans perte d'information est possible. C'est le cas de quelques caractères spéciaux présents dans plusieurs des variantes à 8 bits. Par exemple, tous les jeux de caractères ISO à 8 bits destinés à l'Europe ont les mêmes codes pour « ä », « ö » et « ü ». Ces caractères peuvent donc être envoyés tels quels dans les requêtes distribuées, laissant aux systèmes interrogés le soin d'ignorer le signe diacritique ou non. Les systèmes sont en effet tolérants à « trop » de détail et acceptent les diacritiques dans les requêtes même si ceux-ci sont ignorés dans l'indexation.
2. Si certains caractères d'un terme recherché encodé en Unicode n'ont pas d'équivalent dans certains des jeux de caractères des bases de données interrogées, une conversion sans perte d'information est impossible. La perte est probablement sans importance cependant : le caractère manquant serait de toute façon ignoré dans la base de données interrogée, qui supprimerait son signe diacritique ou le convertirait en sa lettre ordinaire la plus proche pour l'indexation et la recherche. Dans ce cas, le portail pourrait effectuer la conversion en amont, avant d'envoyer la requête distribuée dans un jeu de caractères où le caractère d'origine est absent.
3. La perte provoquée par la conversion pourrait même être bénéfique en étendant la recherche. Par exemple, si un usager saisit « Łodz » comme terme de recherche, celui-ci sera envoyé tel quel aux bases de données ayant Unicode ou le jeu de caractères de l'Europe de l'Est. « Ł » étant absent des autres jeux de caractères à 8 bits, les bibliothèques employant l'un de ces jeux auront été forcées de réduire ce caractère en « L » et encoder « Lodz ». Un terme recherché réduit de cette façon permettra donc quand même d'accéder aux notices pertinentes.

Deuxièmement, les usagers sont rarement conscients des variations de profils locaux en dehors des langues qu'ils parlent. Reprenant l'exemple précédent, si un usager ne parlant pas polonais cherche une image de Łodz et saisit « Lodz » comme terme recherché, les bases de données polonaises ne renverront que les notices où le nom de cette ville a d'emblée été écrit ainsi (par exemple parce qu'elles décrivent des documents n'étant pas d'origine polonaise) ou liées à une notice d'autorité où cette forme a été ajoutée comme renvoi. L'utilisateur va manquer plusieurs notices, dans la mesure où ces bases de données contiendront probablement plus de documents pertinents, au moins dans le cas de The European Library. Le problème inverse peut en effet se rencontrer dans des bases de données communautaires telles que flickr, www.flickr.com, où « Łodz » est référencée 50 fois moins

que « Lodz ». Il s'agit d'une base de données de photographies numériques, à laquelle des touristes non-Polonais ont manifestement plus contribué que les autochtones.

Troisièmement, même si l'utilisateur sait qu'il vaudrait mieux saisir « Łodz » plutôt que « Lodz », il reste un problème de saisie : les claviers sont localisés d'un point de vue national, régional et/ou linguistique pour saisir les caractères spéciaux de la (ou des) langue(s) nationale(s) plus quelques autres choisis presque arbitrairement. Bien qu'Unicode soit maintenant le jeu de caractères de base des systèmes d'exploitation actuels, les pilotes de clavier sont toujours restreints aux touches et combinaisons de touches nécessaires à la (ou aux) langue(s) nationale(s) pour laquelle (ou lesquelles) le clavier a été conçu. La saisie par d'autres moyens, lorsqu'elle est possible, est mal pratique.

Affichage des résultats

Comme on l'a vu précédemment, les bases de données interrogées sont susceptibles de renvoyer des notices dans différents jeux de caractères. Mélanger plusieurs jeux de caractères dans une même page web n'est pas facile parce que le jeu de caractères est déclaré dans la balise <head> et n'est pas censé changer dans le corps du texte. Bien qu'il y ait des moyens de le faire néanmoins, il n'est en fait pas nécessaire de faire cohabiter plusieurs jeux de caractères dans l'affichage des résultats : tout ce qui n'a pas été renvoyé en Unicode devrait être converti dans ce jeu.

Les notices renvoyées peuvent être affichées telles qu'elles arrivent ou réarrangées selon divers critères. Dès qu'intervient quelque part un ordre alphabétique, comme c'est le cas avec certains systèmes qui trient les résultats avant de les envoyer, la diversité des profils locaux entraîne un risque que les usagers « loupent » des ressources parce qu'ils s'attendent à les voir ailleurs dans l'ensemble des résultats. C'est particulièrement net parce que les profils locaux d'ordres alphabétiques ne sont pas si différents les uns des autres que l'application d'un profil inconnu fasse paraître l'ensemble comme non trié.

On peut comparer par exemple les ordres alphabétiques des profils locaux anglais, estonien et slovaque :

Anglais : 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

Estonien : a b c d e f g h i j k l m n o p q r s š z ž t u v w õ ä ö ü x y

Slovaque : a b c č d e f g h ch i j k l m n o p q r ř s š t u v w x y z ž

Les séquences de l'estonien et du slovaque montrent quatre cas d'adaptations locales de l'ordre alphabétique ordinaire :

- Déplacement de certaines lettres : t, u, v, w, x et y viennent après z en estonien.
- Individualisation d'une lettre avec signe diacritique après sa lettre porteuse : č et ř en slovaque, š et ž dans les deux profils locaux.
- Individualisation d'une lettre avec signe diacritique à la fin de l'alphabet : õ, ä, ö et ü en estonien.
- Traitement d'une paire de lettres comme si elles étaient un seul caractère : ch en slovaque.

Selon la combinaison des profils locaux de l'utilisateur et de la base de données interrogée, jusqu'à 15% des notices pourraient paraître mal placées du point de vue de l'utilisateur, pas forcément assez pour que ça lui saute aux yeux mais bien assez pour lui occasionner des difficultés à trouver une ressource uniquement parce que son nom ou son titre contient un caractère spécial. Les effets de ces différences ne devraient pas être sous-estimés : l'ordre alphabétique est une chose qu'on apprend très tôt, en même temps que les lettres elles-mêmes.

5. Solutions et conclusion

Les détails des problèmes décrits ici ont été établis par un questionnaire envoyé à toutes les bibliothèques nationales des Nouveaux Etats Membres et les résultats de ce dernier consignés dans un rapport. Ils montrèrent tout d'abord qu'Unicode était déjà bien répandu. Au fur et à mesure que son usage se propage et que les bases de données employant un autre jeu de caractères disparaissent, plusieurs problèmes d'interopérabilité dans les jeux de caractères se résoudre d'eux-mêmes.

Les problèmes de profils locaux subsisteront cependant dans les deux domaines mentionnés : les divergences dans les règles d'indexation et l'ordre alphabétique.

Trois solutions permettraient d'atténuer les problèmes entraînés par les divergences dans les règles d'indexation.

L'une est de fournir aux utilisateurs les moyens de saisir leurs requêtes aussi précisément que possible. D'abord pour les rendre attentifs au fait que certains caractères spéciaux sont susceptibles d'être importants, leur usage devrait être ajouté aux descriptions des bases de données que l'on peut interroger. Une simple liste des caractères spéciaux indexés individuellement comme dans le tableau

de la page 3 donnerait déjà un on indice. Ensuite, pour faciliter la saisie, un clavier virtuel peut être ajouté à la page où l'on saisit sa recherche, de manière que les usagers puissent saisir les caractères absents de leur clavier physique.

Une autre aide pourrait venir de notices d'autorité enrichies. Si les plus importants noms et termes contenant un caractère spécial indexé individuellement dans son propre profil local mais pas dans les autres ont des renvois à partir d'une forme non localisée, tous les usagers auront de meilleures chances d'atteindre les notices pertinentes, c'est-à-dire même les usagers ignorant les particularités des profils locaux.

Une troisième solution au problème des règles d'indexation différentes pourrait être un doublement des index dans les systèmes pour bibliothèques, qui auraient alors une série d'index localisés et une série d'index non localisés, c'est-à-dire où tous les signes diacritiques sont ignorés et tous les caractères spéciaux convertis en la lettre ordinaire la plus proche. Un port Z39.50 séparé et destiné à l'accès international et aux portails pointerait alors vers les index non localisés. On remarquera que cette solution échange le « silence » contre le « bruit », puisque certains noms ou mots différenciés dans certains profils locaux seraient indexés de la même manière. Pour cette raison, l'accès aux index non localisés devrait rester une option, permettant aux usagers de choisir l'accès le plus approprié à leurs connaissances linguistiques et aux caractéristiques de leur recherche.

Comme ceci n'est techniquement pas réalisable actuellement, The European Library appliquera la première solution, documentation des caractères spéciaux et clavier virtuel. La faiblesse de cette solution est que les usagers lisent rarement les instructions et que cela leur demande un niveau de connaissances qu'ils n'ont pas nécessairement. Heureusement, plusieurs partenaires appliquent la seconde solution et complètent une partie de leurs notices d'autorité par des formes non localisées.

La résolution des divergences des profils locaux sur l'ordre alphabétique est possible en faisant faire le tri par le portail, sur lequel il n'est pas difficile d'installer des modules de tri pour tous les profils locaux européens. On peut alors laisser l'utilisateur choisir le profil local qu'il désire voir appliquer à ses résultats ou, plus simplement, le déduire de la langue d'interface sélectionnée. Il serait nécessaire de retrier les ensembles de notices envoyés par certains systèmes, afin d'appliquer le profil local attendu par l'utilisateur.

La localisation est une bonne chose : elle rend les ordinateurs plus conviviaux et tient compte de la diversité culturelle. Il est cependant important de ne pas compromettre la coopération internationale dans l'accès aux ressources numériques. Passer au système uniforme des débuts à une série de profils locaux adaptés aux diverses communautés d'utilisateurs a en effet entraîné quelques nouveaux problèmes. Les portails internationaux ont un défi à relever dans ce domaine et, heureusement, quelques moyens pour le faire. Ceux-ci comprennent aussi bien des réglages techniques que la sensibilisation des usagers de cette diversité culturelle, en toute concordance avec les buts de l'Europe.