TECHNIQUE DE RÉDUCTION - TRIS INFORMATIQUES À QUATRE CLÉS

Alain LaBonté
informaticien-conseil
Secrétariat du Conseil du trésor
Gouvernement du Québec

Version revue et corrigée de l'original par l'auteur en janvier 1996 et août 1998

Contacter : Alain LaBonté

Table des matières

Avant-propos

Depuis la publication, en août 1988, de la dernière version du document Règles du classement alphabétique en langue française et procédure informatisée pour le tri, de multiples commentaires ont été reçus, aucun ne modifiant la structure de données proposée. En fait, la principale modification effectuée pour la rédaction du projet de norme canadienne touche l'ordre accordé aux ligatures, qui auront un indice de priorité des diacritiques plus petit que ceux des autres caractères accentués au lieu d'avoir un indice plus grand. En effet, l'indice <22> pour les ligatures deviendra l'indice <16> et l'indice <16> précédent, indiquant l'absence d'accent, deviendra l'indice <15> pour marquer la priorité des caractères non accentués sur les ligatures. En français ou en anglais, ceci ne devrait en rien affecter l'ordre de classement des mots existants. Cependant, si un mot comme « cætla » existait, il devrait être classé avant « çà et là » , ce qui n'aurait pas été le cas avec les tables proposées précédemment. Dans le projet de norme, une table a aussi été ajoutée pour donner un classement logique aux caractères spéciaux dans la clé de quatrième ordre, au lieu de simplement les déplacer et de conserver un ordre de classement correspondant à leurs valeurs codées (valeurs qui peuvent différer selon le code utilisé).

Par ailleurs, il est essentiel de remercier M. Pachl, de Prague (Tchécoslovaquie), qui s'est penché sur l'adaptation de la méthode de classement à la langue de son pays, ainsi que M. Johan van Wingen, de Leiden (Pays-Bas), qui a fait le même travail pour différentes autres langues.

M. van Wingen a aiguillonné quelque peu ma curiosité en me lançant prudemment, trop prudemment peut-être, que l'on pourrait simplifier dans certains cas les clés de deuxième et de troisième ordre s'il n'y avait pas d'accent ou de majuscule dans la clé de tri. Après mûre réflexion, non seulement je recommande cette technique, mais je crois qu'elle peut être généralisée en la poussant plus loin. J'ai utilisé à dessein le terme de « réduction » et non de « compression » pour la technique décrite, parce qu'on n'a pas besoin de « décomprimer » pour effectuer les comparaisons conduisant au résultat du tri, une fois que la simplification a eu lieu. Et le bénéfice qui en résulte est une réduction sensible de l'espace requis pour le stockage.

Avantages et inconvénients de la méthode de base

Dans le document préalable Règles du classement alphabétique en langue française et procédure informatisée pour le tri (ISBN 2-550-19046-7), les champs de tri sont décomposés en quatre sous-clés qui, une fois concaténées, peuvent être directement utilisées pour effectuer des comparaisons binaires conventionnelles conduisant à un ordre correct de classement (ordre du dictionnaire) pour la ou les langues visées.

L'avantage de concaténer ces clés et de les stocker telles quelles est évident : beaucoup de systèmes existants peuvent directement s'en accommoder là où un ordre de classement à la fois culturellement correct et mécaniquement prévisible est désiré (programmes de tri, clés permanentes de fichiers indexés, comparaisons dans les langages de programmation, etc.). L'autre avantage majeur est que, comme toute l'information d'origine est conservée, le champ d'origine peut être reconstitué exactement.

Toutefois, l'inconvénient principal de cette méthode est l'espace requis pour le stockage, lorsqu'il existe un besoin de stockage permanent : cet espace sera approximativement de 2 à 3 fois plus grand que l'espace requis par le champ d'origine. La taille des clés peut cependant déjà être réduite en utilisant des champs de longueur variable, ce à quoi la méthode décrite est parfaitement bien adaptée. Une autre technique d'économie d'espace consiste à décomposer le champ de départ uniquement pour la durée de comparaison et à stocker l'original ; cela exige toutefois de modifier tous les programmes existants qui doivent fonctionner selon cet ordre de classement (méthodes d'accès indexé, processus de comparaison dans les systèmes, etc.).

Technique de réduction

Une propriété mathématique des éléments de la structure proposée peut cependant être utilisée de façon très intelligente pour réduire la taille des clés résultantes. Cette propriété, rappelons-le, est la suivante : les indices utilisés pour la confection de la clé de premier ordre sont tous plus grands que les indices utilisés pour la clé de deuxième ordre ; la même logique s'applique entre la clé de deuxième ordre et la clé de troisième ordre ; pour ce qui est de la clé de quatrième ordre, un « zéro logique » agit comme délimiteur au début de celle-ci pour s'assurer qu'elle commence par un indice plus petit que les indices des trois autres ordres de clés.

Cela permet de tirer deux conclusions très simples :

1. Si l'on balaie de **droite à gauche** une clé de deuxième ou de troisième ordre qui vient d'être formée, on peut éliminer les indices correspondant à l'indice le plus petit possible pour une clé de cette ordre jusqu'au premier indice (le dernier de la clé dans le sens habituel de lecture) qui soit plus grand. Plus simplement, de façon particulière, s'il n'y a aucun signe diacritique dans le champ d'origine, on peut omettre complètement la clé de deuxième ordre ; de même, s'il n'y a aucune majuscule dans un mot, on peut omettre complètement la clé de troisième ordre.
2. Pour ce qui est de la quatrième clé, si le champ initial ne contient aucun caractère alphabétique, on sait déjà qu'il n'y aura pas de clés de premier, de deuxième et de troisième ordre (chaînes nulles). Selon la méthode décrite dans le document préalable, on doit coder dans la clé de quatrième ordre la position de chacun des caractères spéciaux. Or, **si les trois premiers ordres de clés sont absents**, il n'y a pas de raison de coder ces positions ; néanmoins, le « zéro logique » doit toujours être préservé comme délimiteur de début.

Le gros avantage de cette technique est qu'elle n'affecte en rien le processus de comparaison. En informatique, généralement, lorsque l'on parle de compression, on doit « décomprimer » avant d'effectuer tout traitement sur ce qui a été « comprimé ». Mais ici, on peut stocker et utiliser pour tout traitement les clés obtenues par la technique de réduction sans aucun problème. Et il sera encore possible de reconstituer le champ d'origine.

Exemple de réduction

Voici la liste donnée en exemple dans le document préalable* (avant réduction), suivie des résultats de l'application de la technique de réduction.

Avant réduction :
LISTE À CLASSER LISTE CONVERTIE POUR LE TRI
(déjà dans l'ordre)

cæcal	*caecal*<16><16><16><22><22><16>* *<08><08><08><08><08><08>*<00>*
caennais	*caennais*<16><16><16><16><16><16><16><16>* *<08><08><08><08><08><08><08><08>*<00>*
C.A.F.	*caf*<16><16><16>*<09><09><09>*<00><01>.<03>.<05>.*
c'est-à-dire	*cestadire*<16><16><16><16><18><16><16><16><16>* *<08><08><08><08><08><08><08><08><08>* *<00><01>'<05>-<07>-*
jésus	*jesus*<16><16><16><17><16>* *<08><08><08><08><08>*<00>*
Jésus	*jesus*<16><16><16><17><16>* *<09><08><08><08><08>*<00>*
pêche	*peche*<16><16><16><19><16>* *<08><08><08><08><08>*<00>*
PÊCHE	*peche*<16><16><16><19><16>* *<09><09><09><09><09>*<00>*
péché	*peche*<17><16><16><17><16>* *<08><08><08><08><08>*<00>*
PÉCHÉ	*peche*<17><16><16><17><16>* *<09><09><09><09><09>*<00>*
pechère	*pechere*<16><16><18><16><16><16><16>* *<08><08><08><08><08><08><08>*<00>*
péchère	*pechere*<16><16><18><16><16><17><16>* *<08><08><08><08><08><08><08>*<00>*
vice-légat	*vicelegat*<16><16><16><17><16><16><16><16><16>* *<08><08><08><08><08><08><08><08><08>* *<00><04>-*
vice versa	*viceversa*<16><16><16><16><16><16><16><16><16><16>* *<08><08><08><08><08><08><08><08><08>* *<00><04> *
vice versa	*viceversa*<16><16><16><16><16><16><16><16><16><16>* *<08><08><08><08><08><08><08><08><08>* *<00><04> <05> *
vice-versa	*viceversa*<16><16><16><16><16><16><16><16><16><16>* *<08><08><08><08><08><08><08><08><08>* *<00><04>-*

Note : L'astérisque (*) n'est pas utilisé pour composer la chaîne de tri mais délimite seulement celle-ci pour montrer la présence d'espaces à l'intérieur du champ à trier et pour séparer les quatre ordres de clés.

Après réduction :

LISTE À CLASSER	LISTE CONVERTIE POUR LE TRI
cæcal	*caecal*<16><16><16><22><22><..>* *<..><..><..><..><..><..>*<00>*
caennais	*caennais*<..><..><..><..><..><..><..><..>* *<..><..><..><..><..><..><..><..>*<00>*
C.A.F.	*caf*<..><..><..>*<09><09><09>*<00><01>.<03>.<05>.*
c'est-à-dire	*cestadire*<16><16><16><16><18><..><..><..><..>* *<..><..><..><..><..><..><..><..><..>* *<00><01>'<05>-<07>-*
jésus	*jesus*<16><16><16><17><..>* *<..><..><..><..><..><..>*<00>*
Jésus	*jesus*<16><16><16><17><..>* *<09><..><..><..><..><..>*<00>*
pêche	*peche*<16><16><16><19><..>* *<..><..><..><..><..><..>*<00>*
PÊCHE	*peche*<16><16><16><19><..>* *<09><09><09><09><09>*<00>*
péché	*peche*<17><16><16><17><..>* *<..><..><..><..><..><..>*<00>*
PÉCHÉ	*peche*<17><16><16><17><..>* *<09><09><09><09><09>*<00>*
pechère	*pechere*<16><16><18><..><..><..><..>* *<..><..><..><..><..><..><..><..><..>*<00>*
péchère	*pechere*<16><16><18><16><16><17><..>* *<..><..><..><..><..><..><..><..><..>*<00>*
vice-légat	*vicelegat*<16><16><16><17><..><..><..><..><..><..>* *<..><..><..><..><..><..><..><..><..><..>* *<00><04>-*
vice versa	*viceversa*<..><..><..><..><..><..><..><..><..><..>* *<..><..><..><..><..><..><..><..><..><..>* <00><04> *
vice versa	*viceversa*<..><..><..><..><..><..><..><..><..><..>* *<..><..><..><..><..><..><..><..><..><..>* *<00><04> <05> *
vice-versa	*viceversa*<..><..><..><..><..><..><..><..><..><..>* *<..><..><..><..><..><..><..><..><..><..>* *<00><04>-*

Note : On a indiqué ici par <..> tout ce qui a été enlevé. Soulignons que 151 indices ont été éliminés sur 326, soit un peu plus de 46 % d'économie. Les champs d'origine consistaient en 116 caractères décomposés maintenant en 175 caractères, soit environ 1,5 fois la taille de départ au lieu de 2,8 fois. Même si en anglais, l'économie serait évidemment plus grande, elle le serait en français aussi, en général, puisque cet exemple concentre des particularités qu'on ne retrouve évidemment pas en si grand nombre dans les mots de la langue française. Pour ce qui est des noms de famille et des prénoms, par exemple au Québec, société à très forte majorité française d'origine, environ une personne sur cinq, tout au plus, a un nom comportant au moins un accent...

Dans la liste de l'exemple précédent, tous les champs d'origine étaient alphabétiques (avec quelques caractères spéciaux). Donnons maintenant un exemple non alphanumérique et voyons l'économie :

Chaîne d'origine :

{[/\]}

Clé résultante d'après la méthode originale :

<00><00>{<01>[<02>/<03>\<04>]<05>}

Clé simplifiée après réduction :

<00>{[/\]}

Dans ce cas, il y a une économie de 50 % (si on néglige le « zéro logique » qui délimite toutes les chaînes ainsi formées). Les chaînes qui ne contiennent pas de lettres ou de chiffres sont toutefois plutôt rares dans les bases de données à caractère nominatif ou dans les banques de textes.

Problèmes potentiels de la technique de réduction

Dans certaines langues ou certaines méthodes de tri, il serait plus difficile de simplifier s'il existait un signe diacritique plus prioritaire (indice le plus petit) que l'absence de signe, ou si, pour régler, par exemple, des problèmes causés par certaines ligatures qui auraient priorité sur les lettres de l'alphabet de base, on leur attribuait un indice plus petit que pour l'absence de signe. Une difficulté analogue existerait s'il y avait une forme de lettre, rare, plus prioritaire que la minuscule.

Ainsi, pour respecter l'usage des dictionnaires de langue allemande et accommoder cette langue dans le projet de norme canadienne, nous avions attribué au « s dur » (ß) allemand un indice de deuxième ordre inférieur à la valeur d'une lettre non accentuée pour qu'un mot comme « groß » soit classé avant « gross ». Mais il serait pratiquement impossible de simplifier la clé de deuxième ordre au Canada si on ne changeait pas la valeur codée attribuée au « s dur ». Or, heureusement pour nous, une autre référence allemande, la norme DIN 5007 révisée, indique que le « s dur » doit être classé après le « double s » : pour cette norme, il faut donc classer « gross » avant « groß ». Alors à cause de cette référence allemande et malgré l'usage du dictionnaire allemand, nous changerons vraisemblablement l'indice du « s dur » dans le projet de norme canadienne pour permettre l'application de cette technique de réduction.

Référence bibliographique de l'édition originale sur papier

Gouvernement du Québec
Dépôt légal -- 3e trimestre 1989
Bibliothèque nationale du Québec
ISBN 2-550-19965-0